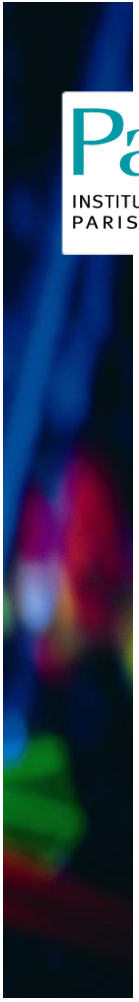
# Deep learning for semantic description of visual human traits

Grigory Antipov



ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

TELECOM ParisTech

2017-ENST-0071

EDITE - ED 130

**Doctorat ParisTech**

# T H È S E

**pour obtenir le grade de docteur délivré par**

## TÉLÉCOM ParisTech

**Spécialité « SIGNAL et IMAGES »**

*présentée et soutenue publiquement par*

**Grigory ANTIPOV**

le 15 décembre 2017

# Apprentissage Profond pour la Description Sémantique des Traits Visuels Humains

Directeur de thèse : **Jean-Luc DUGELAY**
Co-encadrement de la thèse : **Moez BACCOUCHE**

**Jury**

| | |
|---|---|
| **Mme Bernadette DORIZZI**, PRU, Télécom SudParis | Présidente |
| **Mme Jenny BENOIS-PINEAU**, PRU, Université de Bordeaux | Rapporteur |
| **M. Christian WOLF**, MC/HDR, INSA de Lyon | Rapporteur |
| **M. Patrick PEREZ**, Chercheur/HDR, Technicolor Rennes | Examinateur |
| **M. Moez BACCOUCHE**, Chercheur/Docteur, Orange Labs Rennes | Encadrant |
| **M. Jean-Luc DUGELAY**, PRU, Eurecom Sophia Antipolis | Directeur de Thèse |
| **M. Sid-Ahmed BERRANI**, Directeur de l'Innovation/HDR, Algérie Télécom | Invité |

**TÉLÉCOM ParisTech**
école de l'Institut Télécom - membre de ParisTech

T H È S E

# Deep Learning for Semantic Description of Visual Human Traits

GRIGORY ANTIPOV

Joint PhD thesis between Orange Labs and Eurecom


# Remerciements

Ces 3 années de thèse de doctorat ont été pour moi non seulement un défi scientifique mais aussi une immersion dans la culture et la langue française. C'est pour cela qu'en dépit des fautes potentielles, j'ai décidé d'écrire ces lignes de remerciement en français.

En premier lieu, j'aimerais remercier les trois excellents encadrants, Professeur Jean-Luc Dugelay, Docteur Sid-Ahmed Berrani et Docteur Moez Baccouche, qui ont soigneusement guidé mon travail.

Jean-Luc a dirigé cette thèse avec toute son expérience et son expertise dans ce domaine. Je lui suis particulièrement reconnaissant pour la possibilité de passer la première année dans son équipe de recherche à Eurecom où j'ai appris beaucoup de choses.

A son tour, Sid-Ahmed m'a offert cette unique opportunité de faire une thèse CIFRE au sein d'Orange Labs. Ses précieux conseils, son écoute et ses encouragements m'ont énormément aidé durant la première et plus dure étape de la thèse.

Moez a repris l'encadrement pour Orange après le départ de Sid-Ahmed. Son investissement dans cette thèse est inestimable. Pas une seule journée n'est passée sans que nous nous ayons échangé en direct ou par téléphone, sans compter sa grande implication dans toutes les expérimentations et la rédaction des articles. Ses connaissances, sa disponibilité et sa volonté d'aider ont été absolument indispensables pour moi et je l'en remercie vivement.

Je tiens ensuite à remercier les membres du jury de thèse qui ont accepté d'évaluer et de discuter ce travail. Je remercie tout d'abord Mme Bernadette Dorizzi, Professeur à Télécom SudParis, d'avoir présidé le jury de soutenance. Je remercie également Mme Jenny Benois-Pineau, Professeur à l'Université de Bordeaux, et M. Christian Wolf, Maître de Conférences à INSA de Lyon, d'avoir rapporté mon manuscrit et d'avoir contribué à son amélioration. Je tiens enfin à exprimer ma gratitude à M. Patrick Perez, Chercheur à Technicolor Rennes, pour son évaluation de ce travail et pour ses questions enrichissantes.

J'ai eu de la chance d'effecteur ces travaux en faisant partie de deux équipes très accueillantes. Je remercie d'abord tous mes collègues de l'équipe MAS d'Orange Labs, et tout particulièrement Valentin, Clément, Sonia, Stéphane, Adria, Philippe, Nicolas, Claudia, Olivier, Benoît, Patrice et Emmanuel pour m'avoir fait partager leurs connaissances et pour tous les agréables moments passés ensemble autour d'une tasse de café. Je veux également remercier mes collègues-doctorants de l'équipe « Imaging Security » d'Eurecom, Chiara, Natacha, Valeria, Ester et Neslihan, avec qui nous avons eu de nombreuses discussions à la fois très plaisantes et enrichissantes.

Par ailleurs, je n'aurais jamais terminé cette thèse sans le soutien sans réserve de mes parents, de mon frère Dima et de ma belle-famille. Je les remercie tous de tout mon cœur.

Enfin, mes remerciements les plus profonds vont à Pauline qui est ma source d'inspiration et de confiance en moi.

# Abstract

The recent progress in artificial neural networks (rebranded as "deep learning") has significantly boosted the state-of-the-art in numerous domains of computer vision offering an opportunity to approach the problems which were hardly solvable with conventional machine learning. Thus, in the frame of this PhD study, we explore how deep learning techniques can help in the analysis of one the most basic and essential semantic traits revealed by a human face, namely, gender and age. In particular, two complementary problem settings are considered: (1) gender/age prediction from given face images, and (2) synthesis and editing of human faces with the required gender/age attributes.

Convolutional Neural Network (CNN) has currently become a standard model for image-based object recognition in general, and therefore, is a natural choice for addressing the first of these two problems. However, our preliminary studies have shown that the effectiveness of CNNs for a particular task strongly depends on the problem itself and on the strategy which is used for training.

Therefore, in this thesis, we conduct a comprehensive study which results in an empirical formulation of a set of principles for optimal design and training of gender recognition and age estimation CNNs. For example, we demonstrate that learning a CNN to directly recognize gender is less effective than learning the same neural network to firstly recognize a person identity, and then adapting it for gender prediction. We also show that age estimation CNN benefits from a specific representation of age labels which is known as Label Distribution Age Encoding (LDAE). All in all, the conclusions of the performed study allow us to design the state-of-the-art CNNs for gender/age prediction according to the three most popular benchmarks, and to win an international competition on apparent age estimation. When evaluated on a very challenging internal dataset, our best models reach 98.7% of classification accuracy and an average error of 4.26 years for gender recognition and age estimation, respectively.

In order to address the problem of synthesis and editing of human faces, we design and train GA-cGAN, the first Generative Adversarial Network (GAN) which can generate synthetic faces of high visual fidelity within required gender and age categories. Despite GANs are widely praised as one of the best models for image synthesis, applying them for face editing remains an open problem because of the poor preservation of the original face identity by the existing approaches. In this thesis, we propose a novel method which allows employing GA-cGAN for gender swapping and aging/rejuvenation without losing the original identity in synthetic faces. The key idea of our approach is the usage of a separately trained face recognition CNN which helps to minimize the person identity difference between the original and the edited faces. In order to show the practical interest of the designed face editing method, we apply it for age normalization in a cross-age face verification scenario. In average, our method allows improving the accuracy of an off-the-shelf face verification software by about 8 points.

# Résumé

Les récents progrès dans le domaine des réseaux de neurones artificiels (plus connu actuellement sous le nom d'"apprentissage profond") ont permis d'améliorer l'état de l'art dans plusieurs domaines de la vision par ordinateur en offrant une possibilité de s'attaquer à des problèmes qui étaient difficilement traitables par les méthodes d'apprentissage automatique conventionnelles. Ainsi, dans le cadre de cette thèse, nous étudions la question suivante : comment des techniques d'apprentissage profond peuvent-elles aider dans l'analyse des traits sémantiques révélés par le visage humain, à savoir : le genre et l'âge. En particulier, deux problèmes complémentaires sont considérés : (1) la prédiction du genre et de l'âge à partir d'images de visages, et (2) la synthèse et l'édition du genre et de l'âge dans des images de visages.

Par ailleurs, les réseaux de neurones convolutifs (CNNs) sont devenus les modèles standards pour la reconnaissance d'objets visuels et, ainsi, représentent un choix naturel pour traiter la première de ces deux problématiques. Néanmoins, nos études préliminaires ont démontré que l'efficacité des CNNs sur une tâche particulière dépend aussi bien du problème en question que de la stratégie d'apprentissage.

Par conséquent, dans cette thèse, nous effectuons une étude détaillée qui permet d'établir une liste de principes pour la conception et l'apprentissage des CNNs pour la classification du genre et l'estimation de l'âge. Par exemple, nous démontrons que le pré-apprentissage pour la reconnaissance faciale est essentiel pour les CNNs de la classification du genre, et que l'encodage de l'âge distribué (LDAE) est avantageux pour les CNNs de l'estimation de l'âge. Dans l'ensemble, les conclusions de cette étude nous permettent de concevoir les CNNs les plus performantes de l'état de l'art pour la prédiction du genre et de l'âge. De plus, ces modèles nous ont permis de remporter une compétition internationale sur l'estimation de l'âge apparent. Ainsi, nos meilleurs CNNs obtiennent une précision moyenne de 98.7% pour la classification du genre et une erreur moyenne de 4.26 ans pour l'estimation de l'âge, quand ils sont évalués sur un corpus interne particulièrement difficile.

Finalement, afin d'adresser le problème de la synthèse et de l'édition d'images de visages, nous proposons un modèle nommé GA-cGAN : le premier réseau de neurones génératif adversaire (GAN) qui peut produire des visages synthétiques réalistes avec le genre et l'âge souhaités. L'application des GANs à l'édition des images de visages reste un problème ouvert dû au fait que les approches existantes ne préservent pas suffisamment bien l'identité de la personne. Ainsi, nous proposons une nouvelle méthode permettant d'employer GA-cGAN pour le changement du genre et le vieillissement / rajeunissement tout en préservant l'identité dans les images synthétiques. L'idée clé de notre approche consiste en l'utilisation d'un CNN entraîné pour la reconnaissance faciale afin de minimiser la différence entre les identités dans les visages original et édité. Nous appliquons la méthode proposée à la normalisation de l'âge dans le cadre de la vérification faciale avec des écarts d'âges importants. En moyenne, cela permet d'améliorer la précision d'un logiciel de vérification faciale sur étagère d'environ 8 points.

# Contents

# List of Figures

# List of Tables

# List of Acronyms

| | |
|---|---|
| **0/1-CAE** | 0/1-Classification Age Encoding |
| **AAM** | Active Appearance Models |
| **AAE** | Adversarial AutoEncoder |
| **AAEC** | Apparent Age Estimation Competition |
| **AI** | Artificial Intelligence |
| **ALI** | Adversarially Learned Inference |
| **AUC** | Area Under (ROC) Curve |
| **AGES** | AGing pattErn Subspace |
| **ANN** | Artificial Neural Network |
| **BIF** | Biologically Inspired Features |
| **BM** | Boltzmann Machine |
| **CA** | Classification Accuracy |
| **CDL** | Coupled Dictionary Learning |
| **CNN** | Convolutional Neural Network |
| **cGAN** | conditional Generative Adversarial Network |
| **CV** | Cross Validation |
| **cVAE** | conditional Variational AutoEncoder |
| **DBN** | Deep Belief Network |
| **DCGAN** | Deep Convolutional Generative Adversarial Network |
| **ELU** | Exponential Linear Unit |
| **EM** | Expectation Maximization |
| **FAR** | False Acceptance Rate |
| **FRR** | False Rejection Rate |
| **FS** | Fully-Synthetic |
| **GA-cGAN** | Gender/Age-conditioned Generative Adversarial Network |
| **GAN** | Generative Adversarial Network |
| **GPU** | Graphical Processor Unit |

**GSN** Generative Stochastic Networks

**HOG** Histogram of Oriented Gradients

**HS** Half-Synthetic

**ICA** Independent Component Analysis

**ILSVRC** ImageNet Large Scale Visual Recognition Challenge

**LBP** Local Binary Patterns

**LDA** Linear Discriminant Analysis

**LDAE** Label Distribution Age Encoding

**LLE** Local Linear Embeddings

**LMA** Local Manifold Adaptation

**LOPO** Local One Person Out

**LSTM** Long Short-Term Memory

**MAE** Mean Absolute Error

**MLP** Multi-Layer Perceptron

**NN** Nearest Neighbours

**OLPP** Orthogonal Locality Preserving Projections

**PCA** Principal Component Analysis

**PPF** Privacy Protection Filter

**PReLU** Parametric Rectified Linear Unit

**RBM** Restricted Boltzmann Machines

**ReLU** Rectified Linear Unit

**RFA** Recurrent Face Aging

**RI** Re-Identification

**RNN** Recurrent Neural Network

**RVAE** Real-Value Age Encoding

**SGD** Stochastic Gradient Descent

**SIFT** Scale-Invariant Feature Transform

**SoI** Subject of Interest

**SVM** Support Vector Machine

**SVR** Support Vector Regression

**VAE** Variational AutoEncoder

**WLD** Weber Local Descriptors

Chapter 1

# General Introduction

**Contents**



## 1.1 Context and Motivation

Visual recognition of a person is a key aspect of the social communication. For example, we do not address our friends the same way as we address strangers, we do not react similarly to the same actions of a 3-years old child and an adult, and in many languages, the polite forms of referring to an unknown woman and an unknown man are different (*e.g.*, "madam" and "sir" in English).

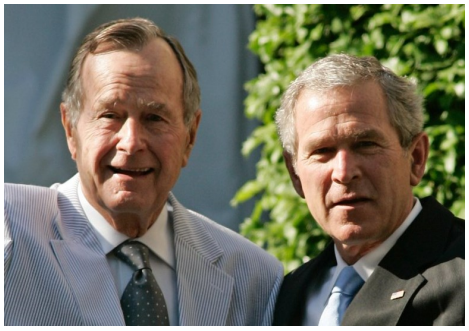
Moreover, being asked to spontaneously describe a given person, our resposne would strongly depend on whether we know the particular individual or not. Thus, given a photo from Figure 1.1.1-(a), the most probable answer would be "Barack Obama, the ex-president of the United States". At the same time, a reasonable description of (a priori unknown) individual in Figure 1.1.1-(b) would rather focus on the person's apparent traits (such as gender and age) resulting in something like "a man of 30-40 years old".

Indeed, gender recognition and age estimation from the visual appearance are tasks which are usually performed intuitively and amazingly fast by humans. This makes the concepts of *gender* and *age* especially useful for the semantic description of a stranger. Moreover, the human understanding of these concepts is much richer than a simple ability to estimate them from a person's appearance. For example, it is often possible to visually recognize a resemblance between a father and a son, or between siblings of different genders even without being acquainted with the particular people (*cf.* Figures 1.1.1-(c) and 1.1.1-(d)). It suggests that humans can subconsciously disentangle the gender, the age and the identity of a given person. So a natural question is whether the contemporary artificial intelligence is capable of performing likewise?

As a matter of fact, the automatic analysis of the human visual appearance is currently a highly demanded area of research. The tremendous development of the digital cameras and the Internet has significantly increased the amount of photos which are constantly created and shared by people all over the

(a)

(b)



(c)

(d)

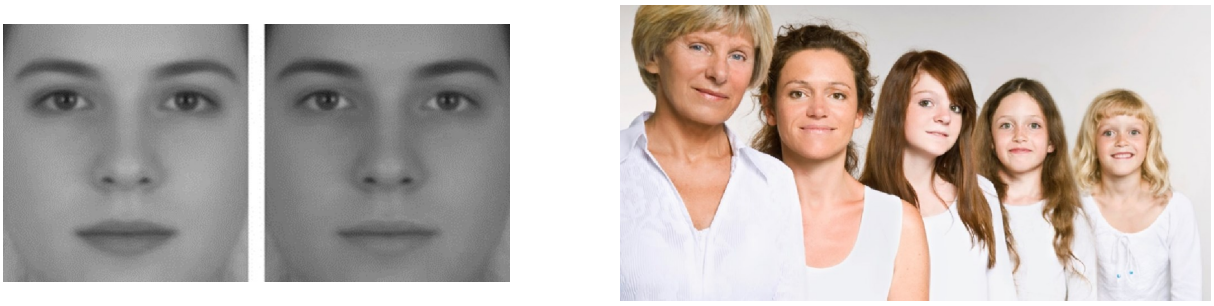
Figure 1.1.1 – (The photos are extracted from public online media sources). Different scenarios of the visual description of humans: (a) a known person who can be directly identified; (b) a person of unknown identity (instead, gender and age can be used for the description of the person's visual appearance); (c) two persons whose father-son relationship can be visually perceived; and (d) two persons whose brother-sister relationship can be visually perceived.

world. For example, according to a white paper published by Facebook, over 350 million of images were uploaded daily to this social network in 2013, and the vast majority of them are human-centred depicting the faces of one or several individuals. As a result, indexing and structuring of such huge collections of photos are not possible without effective solutions for automatic recognition and description of human faces.

In this context, finding the algorithms to extract the gender and age information from faces is particularly indispensable because on the one hand, these concepts are equally applicable for indexing the photos of unknown and known people (*cf.* the example above), and on the other hand, unlike the identity recognition, they preserve the privacy of the personal data.

However, it has appeared hardly possible to algorithmically formalize what makes a face to look more feminine / masculine, or younger / older. Indeed, let us consider the two faces illustrated in Figure 1.1.2-(a): the one on the left seems to belong to a woman, while the one on the right to a man. Despite the perceptive difference, the two faces are actually completely identical, and the visual effect of the gender swapping is achieved just by subtle alteration of the image contrast. In the same spirit, a human vision can effortlessly rank the ladies depicted in Figure 1.1.2-(b) according to their age. At the same time, all of them have smooth skin and no visible wrinkles.

Given the difficulty of the algorithmic solution, the automatic semantic analysis of human faces is



(a) (b)

Figure 1.1.2 – (The photos are extracted from public online media sources). (a) Human visual perception of gender is changed due to subtle contrast variations between the two photos of the *same* face. (b) Human vision can effortlessly rank the presented ladies according to their age even in the absence of the most obvious apparent age characteristics (such as wrinkles, white hair color, eyeglasses etc.)

usually performed by the means of machine learning. A machine learning model is *trained* to perform certain tasks (for example, gender recognition and age estimation) by *learning* from the provided set of examples. The key parameter of such model is its generalization capacity which shows how well the model performs on new examples outside of the training set.

Recently, a particular subset of machine learning models, which are based on artificial neural networks with many hidden layers (*cf.* Chapter 2 for details), has demonstrated an exceptional ability to scale up to large amounts of the training examples and to generalize remarkably better than all alternative approaches. These models, are currently known by the common name of "deep learning".

Due to the appearance of large collections of multimedia data which can be used for training, and to the development of the hardware (notably, Graphical Processor Units (GPUs)), deep learning methods have experienced an unprecedented growth in popularity which, for example, can be perceived via the evolution of the number of the related Google search requests in Figure 1.1.3. As one can observe, the broad interest in the domain started around 2012, and even according to this (very approximative) estimation, only during the period of this PhD study (*i.e.* 2014-2017), the popularity of deep learning has increased in five times!

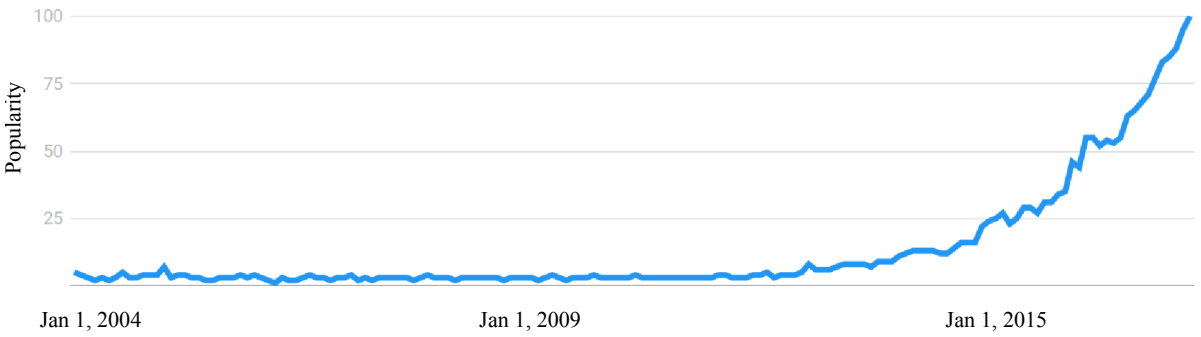



Figure 1.1.3 – Evolution of the interest to the Google research request "deep learning". Obtained via Google Trends (https://trends.google.com/trends/).

The growing interest in deep learning is explained by the record-breaking results which have been

obtained with the help of this technology during the past decade. Today, deep learning reigns in a very broad spectre of applications of artificial intelligence, such as speech recognition [Hin+12; Sai+13], natural language processing [Col+11; SVL14], medicine [Xio+15] and even particle physics [Cio+12]. However, the domain which has been mostly revolutionized by deep learning is arguably computer vision. Currently, almost all the state-of-the-art approaches in image and video classification [Bac+12; Kar+14], image segmentation [LSD15], image restoration and super-resolution [Don+14; RIM17], optical character recognition [Iba+14], saliency prediction [CBPA16], gesture detection and localization [Nev+14] and face recognition [Tai+14; SKP15] are deep learning-based.

Thus, being particularly inspired by the success of deep learning for face recognition, this PhD was initiated by the Multimedia contents Analysis technologieS (MAS) research team of Orange Labs with an objective of exploring the optimal ways of designing and employing of deep models in the frame of gender and age analysis from face images.

On top of that, from a more general perspective, the present work is a logical continuation of a number of deep learning studies which have been conducted in MAS since the mid-2000s. In this context, we can cite the seminal work of Garcia and Delakis [GD04] on face detection (more known as "convolutional face finder") and its embedded version by Roux et al. [RMG06], one of the first attempts to apply deep learning for text recognition by Saidane and Garcia [SG07] (the work which was subsequently improved by Elagouni et al. [Ela+12] and by Yousfi et al. [YBG15]), and the pioneering study of Baccouche et al. [Bac+12] on video classification with deep learning.

## 1.2 Problems and Objectives

In this manuscript, we consider the two principally different problem settings of gender and age analysis with deep learning, namely, "image to label" and "label to image", which are schematically illustrated in Figure 1.2.1.

In the first setting (*i.e.* "image to label"), a static face image is given at the input, and the goal is to recognize the gender and the age based on the image's content. Below, we refer to the described problem as the *gender/age prediction*, or more precisely, *gender recognition* and *age estimation*.

The second problem setting (*i.e.* "label to image") is somewhat opposite to the first one, and it gives rise to a pair of distinct problems of different complexity. The essence of the "image to label" setting is that gender and age are given at the input, and the goal is to generate a synthetic face image with the provided attributes. In a simpler case, there is no other constraints on the face to be generated apart from gender and age, and as a result, we have the *face synthesis problem*. But a more complex and practically important problem consists in *editing* a provided natural face image in order to change the visual perception of its gender and age in accordance with the input values. We refer to this last problem as the *face editing* one.

Gender/age prediction is a particular case of object recognition which is a well-studied problem of computer vision. As it is discussed below in Chapter 2, Convolutional Neural Networks (CNNs) [LeC+89] (a specific kind of deep models) have revolutionized computer vision, and can today be considered as the standard models for object recognition. Nevertheless, CNNs have plenty degrees of liberty both in terms of the neural architectures and in terms of the training techniques.

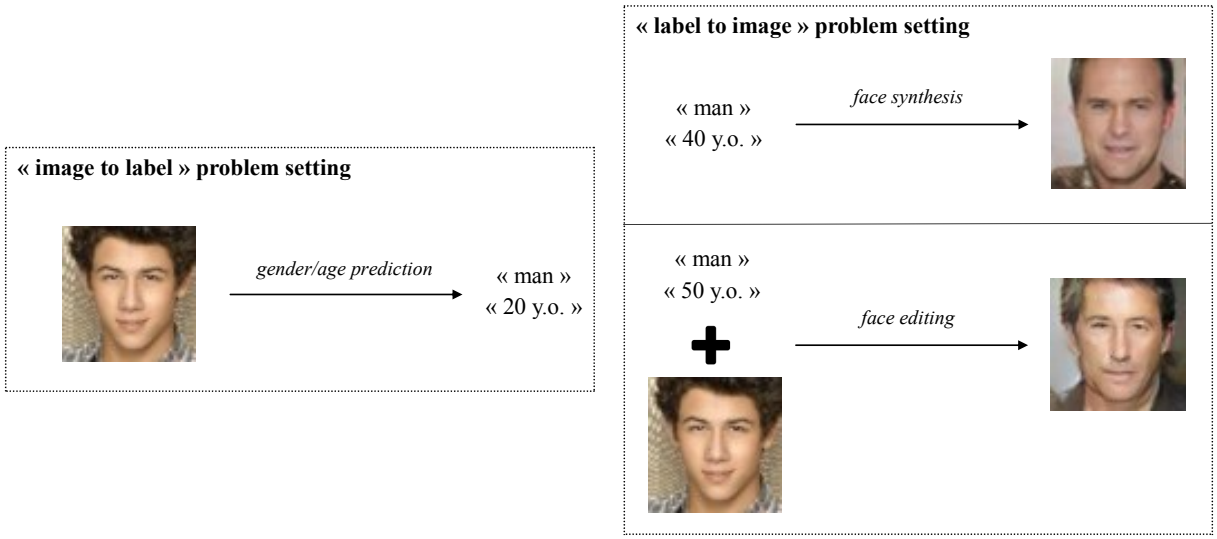



Figure 1.2.1 – Two problem settings and three main problems which are addressed in the present manuscript. In "image to label" setting, a face image is given at the input, and the goal is to recognize the person's gender and age. In "label to image" setting, two distinct problems are considered: synthesis of an arbitrary face with the required gender/age semantics, and editing of a given face to change the visual perception of its gender and/or age (but preserving the original person's identity).

Therefore, the **first objective of this PhD** is to find optimal CNN design and training strategies in the context of gender recognition and age estimation problems. Importantly, the goal is not only obtaining the top performing CNNs for the considered problems, but also explaining what makes them work and therefore, providing the future studies on the subjects with useful hints. The first objective also implicitly implies the comparison between the two sub-problems: gender recognition and age estimation.

If gender/age prediction problem is essential for indexing of large collection of photos (*cf.* Section 1.1), face synthesis and especially, face editing is often used for normalization of faces prior to the indexing. For example, before the verification of a person identity versus an old reference photo, it might be helpful to "age" the old photo so that the reference face is of the same age as the tested one. Contrary to object recognition which is done by *discriminative* models, image synthesis and editing usually requires *generative* models (the difference between these two kinds of machine learning models is formally explained in Chapter 2).

In this manuscript, we employ a particular type of deep generative models, which is called Generative Adversarial Networks (GANs) [Goo+15], and which has recently proved to be very promising for synthesis of images of high visual fidelity (*cf.* Chapter 2). However, training of GANs is an active area of research, and, unlike training of CNNs, it remains highly unstable. Moreover, while perfectly adapted for image sampling, GANs cannot be directly used for image editing.

Therefore, the face synthesis and the face editing problems are solved sequentially in the present manuscript. More formally, the **second objective of this PhD** is twofold: the first step is to train a GAN which performs face generation conditioned on gender and age attributes, and the second step is to design an algorithm which constrains the trained GAN to reproduce the original identity from a provided input face but with altered gender and age attributes.

## 1.3 Contributions

The present manuscript has three main contributions which are described below.

1. On the one hand, we demonstrate that when a training task is challenging enough (in particular, we use the problem of gender recognition from *pedestrian images* as an example, because it is known to be much more complicated than gender recognition from face images), a CNN can learn internal representations which generalize significantly better than the universal human-engineered image representations for object recognition even with limited amount of training data. This effect is further accentuated by the usage of transfer learning (*cf.* Chapter 2 for the definition), which allows to easily adapt the best deep models trained for other tasks to the problem of interest.

   On the other hand, we illustrate that neither the complexity of the neural architectures, nor the big number of training images is a guarantee of the high performances of a CNN. Indeed, we show that the gender recognition accuracy of a CNN trained on *face images* has a relatively small correlation with the depth of the architecture and the dataset size. Moreover, this accuracy is not better than the one obtained with human-engineered face representations (*i.e.* without deep learning).

   Summarizing, our first contribution consists in confirming the fact that deep learning is a very promising approach for estimation of visual human traits, but highlighting that its effectiveness depends on the complexity of the problem which is chosen for the neural network training.

2. Our second contribution corresponds to the first objective of the present PhD which has been formulated in Section 1.2.

   In particular, according to the previous studies, we identify five principal parameters of the CNN design and training which have the biggest impact on the resulting gender/age prediction accuracies. By experimentally selecting the optimal configurations of the selected parameters for the two studied problems, we design the state-of-the-art CNNs for gender recognition and age estimation which outperform existing alternatives on the most popular evaluation benchmarks, and reach the human-level performances on the unconstrained face images taken "in the wild" (*i.e.* spontaneous photos in real-life conditions which is opposed to the controlled photos in predefined conditions).

   Moreover, after adapting our age estimation CNN for *apparent* age estimation (*cf.* Chapter 3 for the definitions of biological and apparent ages), we participated and won an international competition on apparent age estimation.

3. Finally, our last contribution fulfils the second objective of this PhD stated in Section 1.2.

   More precisely, we design a first GAN which is able to synthesise arbitrary face images within the required gender/age categories. After that, we propose a novel approach which allows the usage of the designed generative model for editing of the visual perception of gender and age in natural face images. The particularity of our face editing approach is its universality meaning that it can be potentially applied not only for editing of gender and age, but also for editing of other facial traits.

   We demonstrate the effectiveness and the practical interest of the proposed face editing approach by employing it to improve the cross-age verification accuracy of an off-the-shelf face recognition software.

## 1.4 Organisation of the Manuscript

The rest of this manuscript is organized in six chapters which are split into the following two parts.

**Part I: State of the Art**. Chapters 2 and 3 constitute the overview of the most notable works in two domains related to this PhD: deep learning and gender/age analysis from face images, respectively:

- Chapter 2 is dedicated to the presentation of deep learning. In order to provide a broader view of the domain, we start Chapter 2 with a brief historical perspective illustrating the development of the basic ideas and algorithms of deep learning. Then we focus on the detailed introduction of two types of deep models which are used in the present manuscript, namely: CNNs and GANs.
- In Chapter 3, we present the existing approaches for the two main problems addressed in this manuscript: *i.e.* gender/age prediction from face images and editing of gender/age in face images. The majority of the deep learning studies on the considered subjects was published during the period of this PhD. For convenience, we separately present the works which employ the same classes of deep models as in this manuscript (*i.e.* CNNs for gender/age prediction and generative deep models for face synthesis/editing) at the beginnings of the respective contribution Chapters 5 and 6 highlighting their differences with our approaches.

**Part II: Contributions**. Three main contributions of this manuscript (which have been listed in Section 1.3) are subsequently presented in Chapters 4, 5 and 6. More precisely:

- Chapter 4 reports two preliminary studies which were performed during the first year of the PhD in order to qualitatively understand the advantages and the limitations of CNNs for visual analysis of human traits in different conditions. In particular, in the first study, we extensively compare learned and hand-crafted features (*cf.* Chapter 2 for the corresponding definitions) on gender recognition from pedestrian images, which is a complicated problem with limited training data. On the contrary, in the second study, we train CNNs for a much simpler task and for which much more training data is available: gender recognition from face images. Conclusions of Chapter 4 are used in the subsequent Chapter 5.
- Chapter 5 details the contribution 2 which is briefly described in Section 1.3 and which addresses the first primary objective of this PhD. In particular, in Chapter 5, (1) we identify the CNN design and training parameters which have the biggest impact on the resulting gender and age prediction accuracies, (2) we select the optimal training strategies in the context of the considered problems, (3) we train CNNs for gender recognition and age estimation, and (4) finally, we extensively evaluate both qualitative and quantitative aspects of the resulting deep CNNs.
- Chapter 6 details the contribution 3 which is briefly described in Section 1.3 and which addresses the second primary objective of this PhD. In particular, in Chapter 5, (1) we design and train a GAN which can synthesize versatile and naturally looking face images with the required gender/age semantics, (2) we propose a novel algorithm which allows applying of the designed GAN for face editing, and (3) finally, we demonstrate the practical interest of the proposed face editing approach by applying it for age normalization prior to face verification.

Finally, Chapter 7 concludes the present work summarizing its principal results, and highlighting the directions for the future work, while Chapter 8 is the extended summary of the manuscript in French.

# Part I

# State of the Art

Chapter 2

# Deep Learning for Image Analysis and Synthesis

**Contents**



## 2.1 Introduction

The problems which are addressed by the contemporary artificial intelligence can be roughly split into two categories: (1) the ones which can be described with a set of mathematical rules of a reasonable size, and (2) the ones which are extremely difficult to formalize. The problems of the first kind are often challenging for humans, but are approachable with classical computer science algorithms given enough computational power. A good example of such problem is the game of chess, where the human world champion, Garry Kasparov, was defeated as early as in 1997 by a computer called “Deep Blue” [Hsu02]. Indeed, the chess game environment is restricted by the board of only 64 squares and by 32 pieces which

makes the exploration of the game tree by even (quasi) brute force algorithms incomparably faster than it is done by a human brain.

On the contrary, the problems of the second category may seem trivial for humans, but are prohibitively difficult to put in a computer algorithm. Gender recognition and age estimation, which are studied in the present manuscript, are examples of such problems. Indeed, as already discussed in Chapter 1, it is hardly possible to explicitly name all possible aspects making a human face to be perceived more feminine (masculine) or older (younger). Therefore, almost all gender recognition and age estimation approaches which will be discussed in Chapter 3 are machine learning-based.

The term "machine learning" was coined by Arthur Samuel [Sam59] in 1959. It defines a specific domain of computer science which gives "*computers the ability to learn without being explicitly programmed*". The machine learning algorithms acquire knowledge (often in the form of regular patterns) from the training data, and this knowledge is further used to perform the required tasks (such as classification, clustering, information retrieval, etc.) Today, machine learning is used virtually everywhere: from search engines and e-commerce to private smartphones.

The performance of conventional machine learning algorithms strongly depends on how the raw data is preprocessed before being given at their input. For example, a logistic regression model for credit scoring expects a certain formal set of information about a bank client (such as her/his monthly income, marital status, age, etc.) in order to estimate her/his creditworthiness. This formal set of information is called a *feature representation* (or simply set of *features*) in machine learning.

In the same spirit, automatic analysis of images is rarely performed on raw RGB pixel values. Instead, a common practice of computer vision consists in encodings images with feature representations (such as LBP [OPH96], SIFT [Low99], HOG [DT05], VLAD [Jég+10], etc.) in order to keep only the most relevant information for decision making. Many of these image encodings were initially designed for a particular application, but later employed for other problems as well. Here and below, we refer to such feature representations as *hand-crafted* ones, because they are hand-engineered by human experts based on some prior knowledge about the target domain. Hand-crafted features have two major downsides: firstly, they are extremely difficult to design, and secondly (and more importantly), they are highly problem-dependent [Bac13; LBH15]. For example, as further mentioned in Chapter 3, LBP features happen to be effective for gender recognition but not for age estimation from face images.

A natural way to circumvent this problem is to *learn* feature representations for a particular problem. For example, *manifold learning* allows to learn a projection of input images into another space of lower dimensionality. The resulting low dimensional embeddings are often better adapted for the target problems than initial raw pixels. More generally, a subset of machine learning techniques (manifold learning is one of them) which focus on learning useful representations of the input data is called *representation learning*. The features which are obtained as a result of representation learning are referred as *learned features* in the present manuscript.

Deep learning methods, which are the primary instruments for analysis of human faces in this manuscript, are in their turn, a subset of feature representation techniques (*cf.* Figure 2.1.1 for the complete diagram). The key idea of deep learning is vaguely inspired by the functioning of a human cortex, and it consists in learning feature representations in a hierarchical way: the complex abstract features are composed based on simpler ones. As a matter of fact, the ensemble of techniques, which were recently

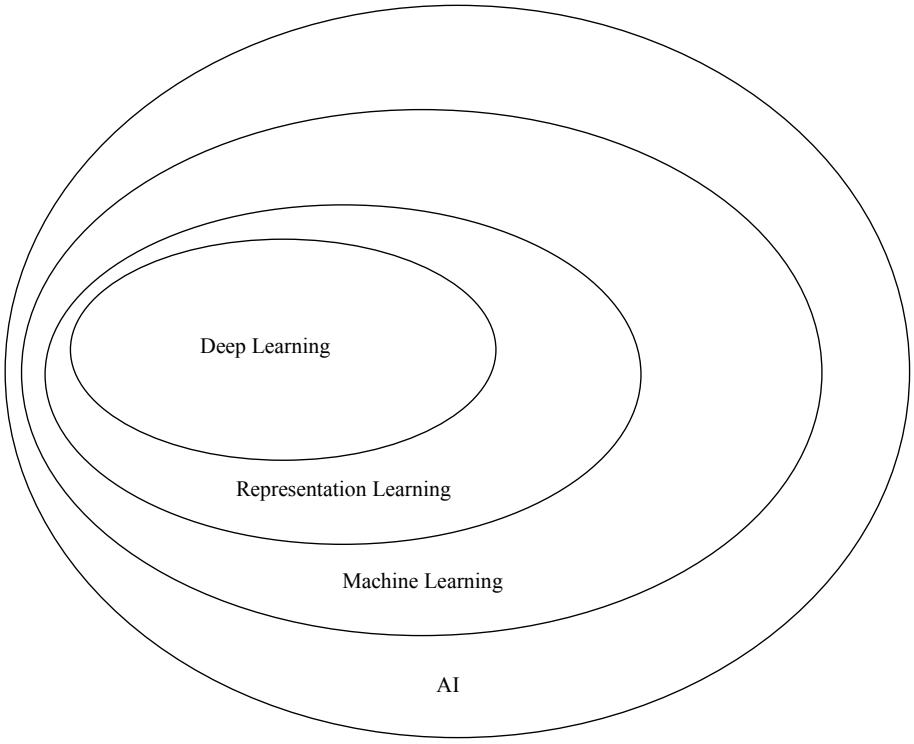



Figure 2.1.1 – (Reproduced from [GBC16]). Venn diagram demonstrating the place of deep learning in the context of representation learning, machine learning and Artificial Intelligence (AI) in general.

rebranded as "deep learning", have been known for a long time by the name of Artificial Neural Networks (ANNs) (*cf.* Section 2.2 for details).

Figure 2.1.2 illustrates a typical (deep) ANN for image classification. It consists of multiple layers, and each layer corresponds to an internal feature representation (informally, the deeper is the layer, the more abstract are the respective features). In particular, the model takes matrices of raw pixel values at its input (corresponding to a human face image, for example), and the learned features of the early layers detect elementary patterns in the input image, *i.e.* edges, colors and corners. The average layers are learned to arrange the detected trivial patterns into more meaningful object parts (in our case, eyes, nose, ears etc.) Finally, the features of the deepest layers operate with the most abstract concepts combining the obtained object parts into the target objects and allowing our model to correctly classify the input image as a human.

As a result, during the training, a deep model for image classification learns both feature representations for images and a classification model. Thus, one of the pioneers of the domain, Yann LeCun, describes deep learning as "end-to-end machine learning" [1].

The rest of this chapter is organised as follows: in Section 2.2, we briefly cover the milestones of the history of deep learning focusing on the basic notions, algorithms and models; then we present in more details two families of deep models which are used in this manuscript, namely convolutional neural networks and generative models (in particular, generative adversarial neural networks) in Sections 2.3 and 2.4, respectively; and finally, Section 2.5 summarizes this overview chapter.

1. The quote can be found here: http://spectrum.ieee.org/automaton/robotics/artificial-intelligence/facebook-ai-director-yann-lecun-on-deep-learning

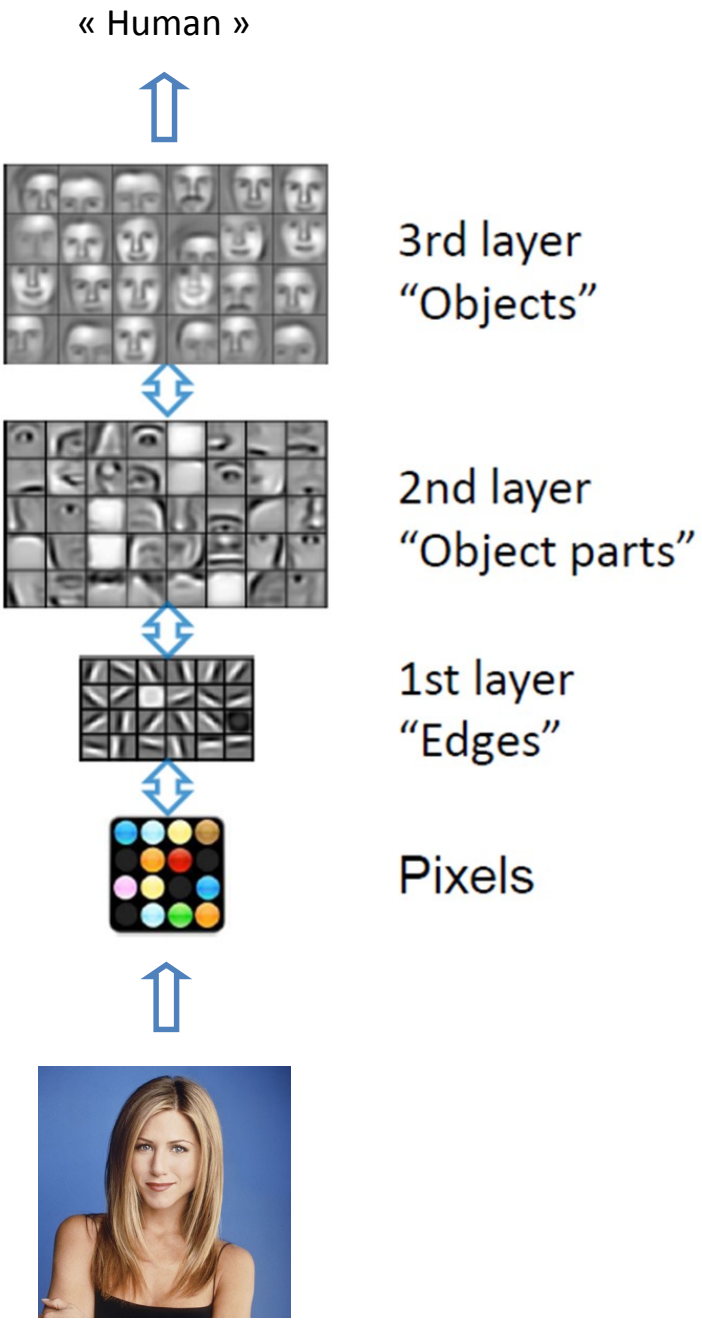



Figure 2.1.2 – (Better viewed in color). Schematic presentation of a typical deep model for image classification. A hierarchy of learned feature representations of increasing complexity allows to correctly classify the input image as a human.

## 2.2 Neural Networks: Key Periods, Models and Algorithms

Despite deep learning has become widely popular only in the recent years, the basic principles and ideas behind ANNs have been developing for more than half a century. Thus, a simplified timeline with the most significant milestones in the history of ANNs is presented in Figure 2.2.1. In this section, we briefly cover the highlights of the development of ANNs introducing the corresponding central notions, models and algorithms. For a more detailed review of the domain, we invite the reader to refer to the excellent book by Goodfellow et al. [GBC16].

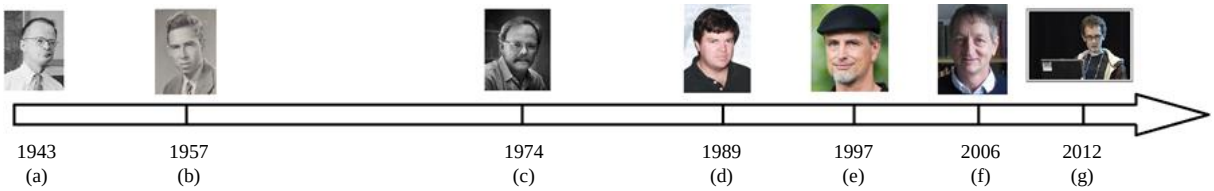



Figure 2.2.1 – The key moments of the history of Artificial Neural Networks (ANNs) and deep learning. (a) Invention of the artificial neuron [MP43]. (b) Invention of Perceptron [Ros58]. (c) Invention of back-propagation [Wer74]. (d) Invention of Convolutional Neural Networks (CNNs) [LeC+89]. (e) Invention of Long Short-Term Memory networks (LSTMs) [HS97]. (f) Invention of Deep Belief Networks (DBNs) [HOT06]. (g) *AlexNet* CNN wins *ImageNet* [KSH12].

### 2.2.1 Artificial Neuron and Perceptron

The history of ANNs dates back to 1943 when McCulloch and Pitts proposed the first model of the artificial neuron [MP43]. It was defined as a function $f$ with $n$ parameters $w_i$ (called *weights*) and $n$ inputs $x_i$:

$$f(x) = a(\sum_{i=0}^{n} x_i w_i) \tag{2.2.1}$$

where $a$ is called an *activation function*, and in the original model of McCulloch and Pitts, it is just a threshold function:

$$a(x) = \begin{cases} 1, & x > 0 \\ 0, & otherwise \end{cases} \tag{2.2.2}$$

The artificial neuron can be trained to perform as an elementary linear classifier using the algorithm called Perceptron which was proposed by Rosenblatt [Ros58]. Training of an artificial neuron means selecting its weights with respect to the provided training dataset with binary annotations. Perceptron is an online learning algorithm (meaning that the corresponding weights are updated after processing of each training example), and after random initialization of weights, it proceeds as following for each training example $x^{(j)}$ with the ground truth binary label $l^{(j)}$:

1. Firstly, the actual output of the classifier is calculated with current state of weights: $p^{(j)}(t) = f(x^{(j)})$.
2. Then, all weights of the classifier are updated with respect to the difference between the ground truth label $l^{(j)}$ and the predicted one $p^{(j)}(t)$: $w_i(t+1) = w_i(t) + (l^{(j)} - p^{(j)}(t))x_i^{(j)}$, $i \in \{0, \ldots, n\}$.

Perceptron learning algorithm seemed very promising for its time and attracted a lot of attention from the media. However, quickly it became obvious that the set of function which can be simulated by Perceptron is very narrow. For example, Minsky and Papert [MP69] demonstrated that XOR binary function cannot be expressed with an artificial neuron defined in Formula 2.2.1. This discovery has drastically cut down the interest in ANNs for a certain period.

### 2.2.2 MLP and Backpropagation

In order to circumvent the stated limitation of Perceptron, the model can be naturally extended to Mutli-Layer Perceptron (MLP) [RHW85]. In MLP, artificial neurons are organized in several layers

which are related between each other by weighted connections (there is a weight $w_{ij}$ which is associated to each connection between the artificial neurons $x_i$ and $x_j$). A typical MLP is illustrated in Figure 2.2.2.

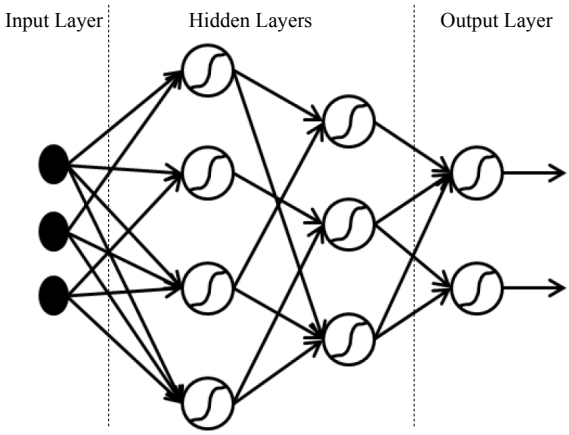



Figure 2.2.2 – (Extracted from [Bac13]). The oriented connection graph (*i.e.* architecture) of a Multi-Layer Perceptron (MLP).

There are three types of layers in an MLP (*cf.* Figure 2.2.2): (1) an *input layer* containing a set of artificial neurons which are fed with the input data, (2) an *output layer* containing the artificial neurons which are associated with the model outputs (for example, in case of classification, each output corresponds to one target class), and (3) *hidden layers* which are the intermediate layers between the input and output ones. The only difference between the original artificial neuron from [MP43] and the ones used in MLP (later, simply "neurons") is the difference in activation functions $a$. Instead of using binary thresholding, the neurons in MLP are usually activated with (rectified) linear, sigmoid or hyperbolic tangent functions. Here and below, we will refer to the outputs of each neuron of an MLP as an *activation* of this neuron.

When data is fed at the input (0-th) layer of an MLP, the model subsequently calculates the activations of neurons in all layers one by one until the output ($n$-th) layer. In particular, given the activations $x_j^l$ of the $l$-th layer, the activation $x_j^{(l+1)}$ of the $(l+1)$-th one are calculated as following:

$$x_j^{(l+1)} = a\left(\sum_{i=0}^{n^l} x_i^l w_{ij}^{l,(l+1)}\right) \tag{2.2.3}$$

where $w_{ij}^{l,(l+1)}$ is the weight associated with the connection between the neurons $x_i^l$ and $x_j^{(l+1)}$, while $n^l$ is the number of neurons (*i.e.* the size) in the $l$-th layer.

As it is the case for Perceptron, training of an MLP consists in finding an optimal configuration of its weights $w$ with respect to the training data. It has been independently shown by several research groups [Wer74; LeC85; RHW85], that MLP can be effectively trained using Stochastic Gradient Descent (SGD) optimization procedure, which is commonly referred as *backpropagation* in the context of ANNs. Backprogation is a generalization of the Perceptron training algorithm which is presented in Subsection 2.2.1. Its main idea is to iteratively update each particular weight $w_{ij}^{l,(l+1)}$ of MLP in the direction which

is the opposite to the gradient of the loss function $L$ with respect to this weight:

$$\Delta w_{ij} = -\alpha \frac{\partial L}{\partial w_{ij}^{l,(l+1)}} \tag{2.2.4}$$

Here, $\alpha$ is a real-valued hyperparamter of backpropagation called *learning rate* which controls the speed of training. The loss function $L$ is also a training hyperparameter. Basically, $L$ can be any differentiable function measuring the difference between the MLP predictions $p$ and the ground truth labels $l$.

The partial derivative $\frac{\partial L}{\partial w_{ij}}$ in Formula 2.2.4 is calculated sequentially, layer by layer, from the output layer down to the input one following the chain rule for differentiation of complex functions. In other words, the error signal from the loss function $L$ is propagated back in neural network (hence, the name "backpropagation"). In practice, the weight updates (*cf.* Formula 2.2.4) are calculated either based on each training example separately (*fully stochastic training*) or based on small batches of several training examples (*mini-batch training*).

MLP trained with batchpropagation proved to be very effective supervised machine learning algorithm and revived the interest in ANNs during the late 80s [LBH15]. Even today, backpropagation remains the basic training algorithm for the majority of deep learning models, while a number of improvements have been proposed to facilitate the convergence of the underlying SGD optimization (*i.e.* introduction of momentum [Pla+86], Nesterov optimization [Nes83], ADAM [KB14], etc.)

### 2.2.3 Data Dependent Models

As mentioned in Subsection 2.2.2, the invention of backpropagation resulted in resurgence of the interest in ANNs. In particular, two specific neural models appeared during the 80's, namely: Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs).

CNNs are deliberately designed to deal with data which has some spatial topology (*i.e.* images, videos, sound spectrograms etc.), and are the central instrument for analysis of human visual traits in the present manuscript. Therefore, in this chapter, we devote a separate Section 2.3 for their introduction, while in the rest of this subsection, we focus on RNNs.

MLP, which is described in Subsection 2.2.2, is also called *feed-forward neural network*, because at test stage, the signal passes in a direct manner from the input layer up to the output one. More formally, an oriented connection graph of a feedforward ANN does not contain cycles (*cf.* Figure 2.2.2).

Unlike feedforward ANNs, RNNs contain one or more cycles in their oriented connection graphs. Thus, Figure 2.2.3-(a) illustrates a typical RNN which has the basic structure of an MLP, but each of its hidden neurons is also connected with itself.

RNNs are designed to process sequences of data, one element at a time. The easiest way to understand how it works in practice is to refer to Figure 2.2.3-(b) which illustrates *unfolding of the RNN* from Figure 2.2.3-(a) *in time*. Thus, an activation $x_j^{(l+1)}(t+1)$ of the neuron $j$ of the $(l+1)$-th layer at time $(t+1)$ depends both on the activations of neurons $x_i^l(t+1)$ from the previous layer $l$ and on the activation of the same neuron $x_j^{(l+1)}(t)$ at the previous time stamp $t$:

$$x_j^{(l+1)}(t+1) = a\left(\sum_{i=0}^{n^l} x_i^l(t+1) w_{ij}^{l,(l+1)} + x_j^{(l+1)}(t) w_{jj}^{(l+1),(l+1)}\right) \tag{2.2.5}$$

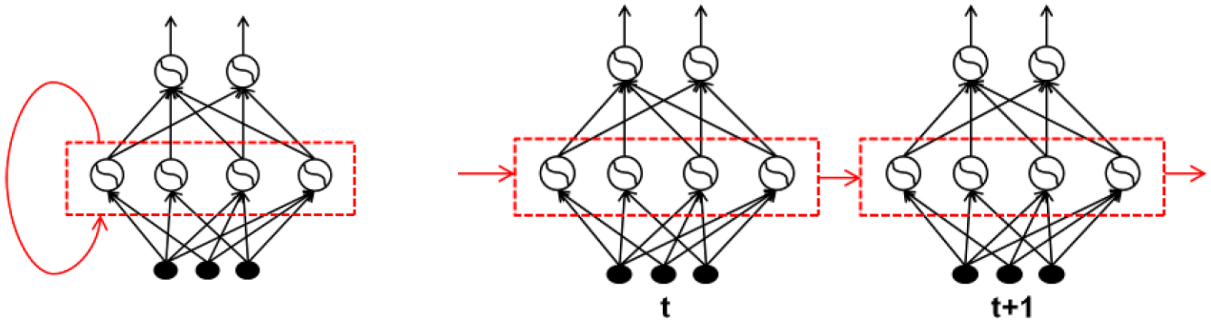



Figure 2.2.3 – (Extracted from [Bac13]). (a) The oriented connection graph (*i.e.* architecture) of a Recurrent Neural Network (RNN), and (b) its unfolding in time.

Backpropagation training algorithm can be effortlessly adapted for the case of RNNs. Indeed, an unfolded version of an RNN within a finite number of time stamps (*cf.* Figure 2.2.3-(b)) is actually a feedforward neural network, meaning that backpropagation can be applied for its training. This particular case of backpropagation is known as BackPropagation Through Time (BPTT) and was firstly proposed by Williams and Zipser [WZ95].

As already mentioned above, the main domain of application of RNNs is modelling of data sequences (such as speech, text, videos, etc.) However, it was found by a number of studies [BSF94; Hoc+01] that RNNs which are trained with BPTT fail to "remember" long-term context from the past of the sequence due to the *vanishing gradient* problem. The problem is formally defined and thoroughly studied in [Hoc+01], but intuitively, it is very easy to understand. Indeed, due to the fact that neurons do not have an explicit memory to store values, the influence of an activation $x_i^l(t)$ at time $t$ is negligible for the activation $x_i^l(t+n)$ of the same neuron at time $(t+n)$, when $n$ is sufficiently big.

The problem of the vanishing gradient was effectively solved by the introduction of Long Short-Term Memory networks (LSTMs) [HS97] which extend RNNs by substituting the artificial neurons of the hidden layers by the specific LSTM cells with the elements of memory. The detailed description of an LSTM is out of the scope of this overview chapter, we therefore refer an interested reader to the original work of Hochreiter and Schmidhuber [HS97]. Today, LSTM RNNs are extensively used in natural language processing [SVL14], speech recognition [GMH13] and in video classification [Bac+12].

Despite the fact that the first applicative studies on CNNs and RNNs demonstrated a number of very promising results (such as recognition of hand-written digits [LeC+98] or phoneme classification [GS05]), ANNs experienced one more significant drop in popularity in the late 90s because of two following main reasons. Firstly, it was (wrongly) thought that it is infeasible to optimize ANNs of many hidden layers with SGD [GBC16] due to the existence of poor local minima in the optimization surface. Secondly (and more importantly), the newly invented machine learning algorithms, such as Support Vector Machine (SVM) [CV95] and graphical models [Jor98] were much easier to train and demonstrated similar or even better performances. Moreover, unlike ANNs with many trainable weights, SVMs are quite robust to *overfitting* (a recurrent problem in machine learning, when a model fits to noisy variations in training data which are irrelevant to the underlying relationship).

### 2.2.4 Deep Learning Revolution

It was not until 2006 that ANNs regain attention of the machine learning community for the third time. The breakthrough was made by Hinton et al. [HOT06] who showed an effective way of unsupervised training of neural networks with many hidden layers. More precisely, the authors proposed an iterative algorithm for training Deep Belief Networks (DBNs) [HS06], a multi-layer unsupervised model which is composed of several single-layer Restricted Boltzmann Machines (RBMs) [Smo86]. Each layer of a DBN learns a more abstract and less redundant representation of the preceding one, meaning that as a whole, the model performs hierarchical representation learning.

DBNs can be used on their own [HOT06], and they can also serve as a good starting point for further supervised *fine-tuning* [KH11], which means initialization of the layers of an ANN with the layers issued after training of another ANN (in this context, the DBN). Fine-tuning is a particular case of *transfer learning* [PY10], a common technique in machine learning allowing to use the knowledge learned from one problem for another one.

Hinton et al. argued [HOT06] that the key of the DBN success is the *depth* (*i.e.* the number of hidden layers) of the trained models which provides the hierarchical structure of the learned features. They even coined the term "deep learning" which became so popular later. This fundamental idea of the importance of the depth for learning effective feature representations has been intuitively motivated in the introductory Section 2.1, but a thorough and a theoretically sound explanation can be found in [BCV13].

The first major success of deep learning was in the domain of speech recognition in 2009 [MDH09] when DBN pretraining allowed effective fine-tuning for predicting probabilities of various fragments of speech with limited amount of training data. The development and the accessibility of GPUs significantly accelerated the progress of deep learning reducing an average training time of an ANN in more than 10 times. Thus, in a couple of years after the paper of Mohamed et al. [MDH09], their speech recognition solution was already implemented on Android smartphones.

Unlike speech recognition, the first record-breaking result of deep learning in computer vision was obtained with a fully-supervised CNN. Thus, in 2012, Krizhevsky et al. [KSH12] trained the deepest neural network of its time containing 8 trainable layers and won the *ImageNet* challenge outperforming all other (non-deep learning) approaches by a very significant margin. *ImageNet* challenge [Rus+15] (its full name is *ImageNet Large Scale Visual Recognition Challenge (ILSVRC)*) is the most prestigious annual challenge on large scale general image classification with 1000 fine-grained classes, about 1.2M training images and about 150K validation and test images.

The work of Krizhevsky et al. [KSH12] revolutionized computer vision, as since 2012, CNNs have become an indispensable part of virtually all state-of-the-art approaches of the domain [Kar+14; LSD15; SKP15]. Moreover, similarly to the usage of DBNs in the case of speech recognition, it was shown in [SR+14] that a CNN pretrained on *ImageNet* classification can be effectively used for further fine-tuning for other computer vision problems with limited training data (we explore this point in more details in Section 4.2 of Chapter 4).

Today, the training of CNNs has been standardized in many aspects (*cf.* Section 2.3), and the fundamental research in deep learning has shifted from fully supervised models (such as CNNs and RNNs) to unsupervised, semi-supervised and reinforcement learnings. For example, very promising results on

modelling of complex data distributions have been recently obtained by the new type of generative models, which are called Generative Adversarial Networks (GANs) [Goo+15]. They are introduced in details in Section 2.4 and are used in Chapter 6 for editing of gender and age in face images.

## 2.3 Convolutional Neural Networks

As briefly introduced in Subsection 2.2.3, CNN is a particular case of ANN which is designed to process the data with a spatial topology (for example, static images). CNN is simply defined as ANN which (at least ones) uses a convolution operation for connecting the neurons of a pair of its consecutive layers [GBC16]. In this section, we firstly explain the basic principles and the intuitions behind CNNs in Subsection 2.3.1, and then in Subsections 2.3.2 and 2.3.3, we introduce the well-established training strategies and CNN architectures which have proved to be effective for a large variety of applicative problems.

### 2.3.1 Typical CNN: Basic Principles and Definitions

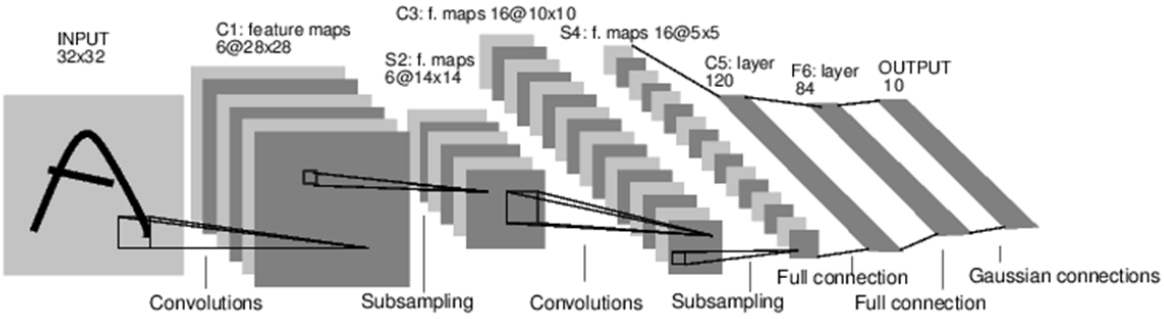



Figure 2.3.1 – (Extracted from [LeC+98]). *LeNet-5* CNN for recognition of handwritten digits.

The easiest way to explain the basic principles of a CNN is to refer to a particular example. Thus, Figure 2.3.1 illustrates *LeNet-5*, one of the first CNNs successfully applied for a real-life problem. The CNN was designed by LeCun et al. [LeC+98] for automatic recognition of handwritten digits. A typical CNN, such as *LeNet-5*, is composed of three types of layers: (1) convolutional ones (labelled by the letter $C$ in Figure 2.3.1), (2) pooling (or subsampling) ones (labelled by the letter $S$ in Figure 2.3.1), and (3) fully-connected ones (labelled by the letter $F$ in Figure 2.3.1). The latter ones are the "standard" layers from MLP (*cf.* Subsection 2.2.2). Usually, convolutional and pooling layers alter each other at the beginning of the network, and a few fully-connected layers are stacked at its end (the output layer is almost always fully-connected).

In image processing, a convolution operator requires two inputs (an image $I$, and a kernel $W$) and produces a single output (an image $O$) which is calculated as following:

$$O(i,j) = (I * W)(i,j) = \sum_m \sum_n I(m,n) W(i-m, j-n) \tag{2.3.1}$$

In the context of CNNs, every convolutional layer is associated with a certain number of kernels which in their turn define the number of *feature maps* of the layer. Using the general ANN terminology from

Section 2.2, the kernels are the trainable parameters of the convolutional layer (*i.e.* its weights) while the feature maps are its activations. For example, the first convolutional layer $C1$ of *LeNet-5* corresponds to 6 kernels of size 6x6 pixels, and when an input image of size 32x32 is convolved with each of these kernels according to Equation 2.3.1, 6 feature maps of size 28x28 are produced [2]. As in case of fully-connected layers, the outputs of a convolutional layer are usually processed by a non-linear activation function.

Thus, the convolutional operation provides the *local connectivity* between the neurons of two consecutive layers which is opposed to the *full connectivity* between a pair of layers in MLP discussed in Subsection 2.2.2. In particular, as illustrated in Figure 2.3.1, a local region defined by the size of the convolutional kernel is connected with a neuron of the resulting feature map. Moreover, the convolutional operator is defined in a way that the same kernel "slides" over all input image meaning that the weights connecting the neurons of the resulting feature map and different regions of the input image are *shared*.

Summarizing, a convolutional layer has two particularities distinguishing it from a fully-connected one, namely: (1) local connectivity and (2) weight sharing, both of which can be intuitively motivated in the frame of image processing. Indeed, the motivation behind the local connectivity is the high correlation between local groups of pixels of a typical image which suggests that they can be effectively treated together. A the same time, weight sharing makes CNNs to be invariant to particular spatial locations of image patterns. It is also interesting to notice that a fully-connected layer can be seen as an extreme case of a convolutional layer when the kernel size is of 1x1. Thus, for historical reasons, in Figure 2.3.1 (which is extracted from [LeC+98]) the fully-connected layer $C5$ is labelled as a convolutional layer by the letter $C$.

If the role of convolutional layers is to detect regular motifs in input data, the goal of pooling layers is to merge semantically similar features in order to construct more complex and abstract features (*i.e.* hierarchical feature learning). Thus, contrary to convolutional and fully-connected layers, the pooling ones do not have trainable parameters. They simply perform downsampling of input feature maps by a certain factor (in case of *LeNet-5* from Figure 2.3.1, by a factor of 2). Before, pooling was usually done by averaging of neighbouring pixels (*average pooling*), but in contemporary deep CNNs, *max-pooling* has been shown to be more effective.

Finally, backpropagation algorithm for MLPs which is discussed in Subsection 2.2.2 is very easy to adapt for training CNNs. Indeed, the constraint of local connectivity in convolutional layers is trivial to implement, while the weight sharing constraint is fulfilled via the synchronised updates of all shared weights by the same value.

### 2.3.2 Established Training Practices

A large body of research has been recently devoted to CNNs, and as a result, a number of universal training practices have been found which significantly improve and simplify the process of the CNN optimization. In this subsection, we discuss the most important and largely adopted of them.

**Activation Function** In early days of ANNs, in general, and CNNs, in particular, sigmoid $\sigma(x) = \frac{1}{1+e^{-x}}$ and hyperbolic tangent $tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$ were usually used as activation functions. However, when CNNs

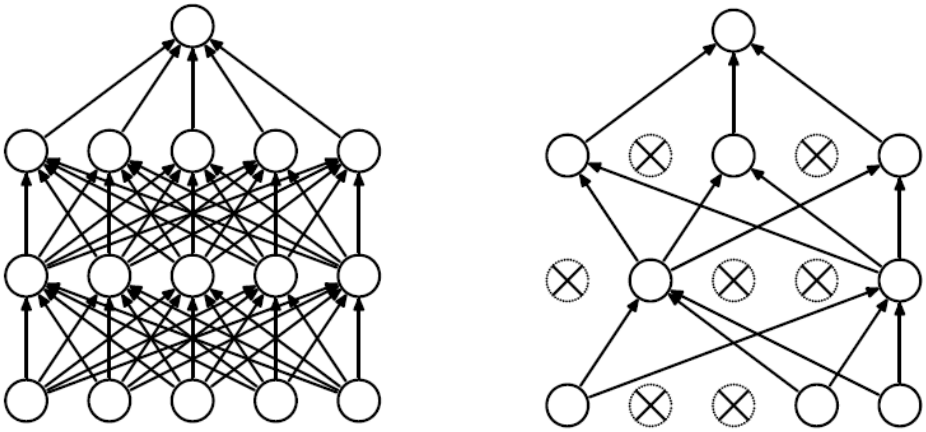
2. Actually, the exact size of the output of the convolutional operator depends not only on the sizes of the input and of the kernel, but also on how border effects are handled.

became deeper, it turned out that sigmoid and hyperbolic activations create the problem of *vanishing gradients*. Indeed, the derivative of sigmoid $\sigma'(x) = \frac{e^x}{(1+e^x)^2}$ is close to zero when the inputs $x$ have a big amplitude (and it often happens at the beginning of the training). However, once there is a zero derivative in a backward pass of backpropagation, all subsequent derivatives in earlier layers are also zeroed (due to the chain rule) meaning that the weights will not be updated and the CNN training will stuck.

In order to avoid this problem, Krizhevsky et al. [KSH12] proposed using *Rectified Linear Unit (ReLU)* activation function which is simply defined as $relu(x) = \max(0,x)$. ReLU activations partly resolve the problem of vanishing gradients in CNNs allowing the convergence of very deep networks. Later, a number of slight improvements to ReLU activations were proposed (such as Parametric ReLU (PReLU) [He+15] and Exponential Linear Units (ELU) [CUH16]). ReLU and ReLU-like activation functions are used in the vast majority of the state-of-the-art CNNs, and in all CNNs of this manuscript.

**Regularization** Deep CNNs are powerful models with millions of trainable weights meaning that they can overfit even on a large training dataset. Therefore, preventing the overfitting is a major issue which defines how well a trained CNN will generalize on unseen data.

One of the simplest techniques in this context is the weight regularization which penalizes the CNN weights with large amplitudes. There are two mostly used weight regularization approaches, namely, $L_1$ and $L_2$, which basically follow the same idea of adding a separate regularization term to the loss function: $\lambda||W||_{L_1}$ and $\lambda||W||_{L_2}$, respectively (where $W$ is the matrix with all weights of a CNN, and $\lambda$ is a real-valued constant controlling the trade-off between the main optimization objective and the regularization term).



(a) (b)

Figure 2.3.2 – (Extracted from [Sri+14]). Dropout algorithm for ANN regularization. During each iteration of training some randomly selected neurons (and respective connections) of an ANN are "switched off". (a) Standard ANN. (b) ANN after applying dropout.

Srivastava et al. [Sri+14] proposed another extremely simple and effective approach to prevent neural networks from overfitting which is called *dropout*. During the training, dropout is implemented by keeping a neuron active with some probability $p$ or setting it to zero otherwise (*cf.* Figure 2.3.2). In other words, during each iteration of training some (randomly selected) neurons are "switched off". Dropout

forces a neural network to learn multiple independent representations preventing its neurons from co-adaptation and making the overfitting less likely (*cf.* the original work [Sri+14] for more details).

Today, dropout is widely adopted by the deep learning community and is employed with and without weight regularization. In the context of CNNs, it is more often used for fully-connected layers.

**Normalization** Before being processed by a CNN, images are usually *centered* and sometimes *normalized*. For RGB images, centering is done by subtracting the mean RGB values from all pixels of the respective image channels (the mean values are calculated over the whole training dataset). Image normalization involves per-channel division of the centered images by the correponding standard deviation values. In practice, image normalization is less important than image centering.

However, neither of the described approaches fixes a common problem of the CNN training which was described by Ioffe and Szegedy [IS15] as *internal covariate shift*. The problem is easy to apprehend intuitively: the distribution of inputs to a hidden layer of a neural network changes all along the training (because the weights connecting the layer with the preceding one change), and therefore, in order to converge, the network has to adapt to this distribution shift which slows down the training. Ioffe and Szegedy proposed [IS15] the *batch normalization* algorithm which compensates internal covariate shifts and significantly stabilizes the training. The main idea is to force the activation of each layer to take on a standard Gaussian distribution throughout the training by performing the normalization on a mini-batch level. The authors demonstrated that this is possible because normalization is a differentiable operator (we invite the interested readers to refer to the original paper [IS15] for details).

Batch normalization has appeared to be surprisingly effective not only for speeding up the training, but also for improving the performances of CNNs [He+16]. Thus, batch normalization layers are currently a standard and an indispensable part of the state-of-the-art CNN architectures.

### 2.3.3 State-of-the-Art CNN Architectures and CNN Applications

| Winner | CNN Name | Year | Number of Weight Layers | Top-5 Classification Error (%) |
|---|---|---|---|---|
| [Per+10] | — | 2011 | — | 25.8 |
| [KSH12] | ***AlexNet*** | 2012 | 8 | 16.4 |
| [ZF14] | *ZFNet* | 2013 | 8 | 14.8 |
| [SZ15] (2nd place) | ***VGG*** | 2014 | 16/19 | 6.8 |
| [Sze+15] | *GoogLeNet* | 2014 | 22 | 6.7 |
| [He+16] | ***ResNet*** | 2015 | 34-152 | 3.6 |

Table 2.3.1 – *ImageNet* competition winners from 2011 to 2015. CNN architectures which are used in the present manuscript are highlighted in bold. Top-5% classification error is one of the competition's metrics which measures how often the target class is not among the 5 most probable classes according to the prediction model (the lower, the better).

Over the last years, CNNs have achieved a tremendous progress in computer vision tasks, and the depth of the state-of-the-art architectures has increased significantly. The evolution of the results of the *ImageNet* general image classification competition, which is presented in Table 2.3.1, often serves as an illustration to these words.

Thus, 2011 was the last year when the winner [Per+10] of *ImageNet* did not use CNNs (and deep

learning). Comparison of the scores in the first two lines of Table 2.3.1 demonstrates the gap between CNN-based and CNN-free approaches on large-scale image classification. As one can observe, *AlexNet* [KSH12] outperforms the result of the previous year winner by more than 9 points. Not only *AlexNet* was the first CNN to obtain the state-of-the-art results on large-scale image classification, but this architecture was also innovative in many aspects. In particular, it was very deep for its time (5 convolutional layers and 3 fully-connected ones), it was one of the first to employ *ReLU* activations instead of the sigmoid ones, and finally Krizhevsky et al. [KSH12] were the pioneers to employ dropout for CNN regularization. Later, Zeiler and Fergus [ZF14] slightly improved *AlexNet* proposing a similar *ZFNet* CNN. Their work is also known for the excellent illustrations of the features which are learned in each layer of a CNN.

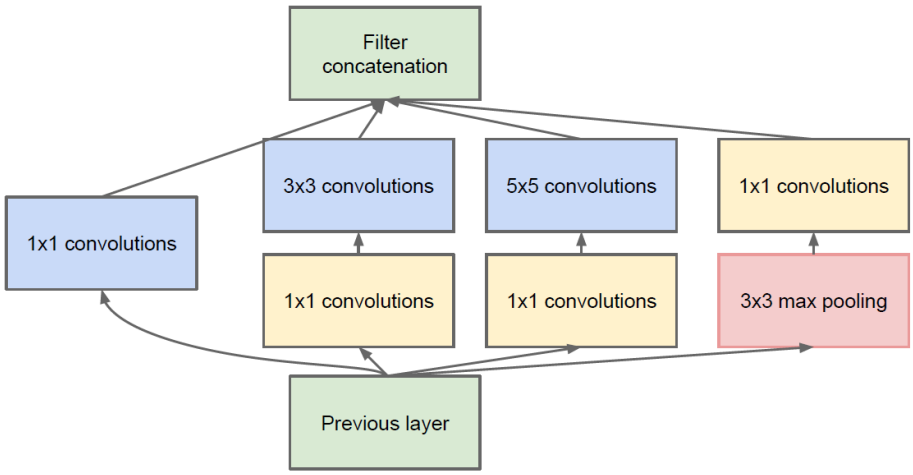



Figure 2.3.3 – (Extracted from [Sze+15]). Inception module which is a building block of *GoogLeNet* CNN.

In 2014, two principally new CNN architectures were proposed, namely: *VGG* [SZ15] and *GoogLeNet* [Sze+15] (the latter one is also known as *Inception*), which demonstrated very close performances in the competition. These CNNs are much deeper than *AlexNet*: 16 or 19 layers for *VGG* (depending on the particular version), and 22 layers for *GoogLeNet*. In both cases, this increase in depth is obtained due to the bigger number of convolutional layers suggesting that convolutional layers have the decisive impact on the CNN performance. Unlike *AlexNet* which uses the convolutional kernels of different sizes, *VGG* is based only on convolutions with kernels of size 3x3. At the same time, being composed of the so-called *Inception* modules (*cf.* Figure 2.3.3), *GoogLeNet* has much less trainable weights than *VGG* and is the one of the first non-linear CNN architectures.

The trend of increasing the depth of the state-of-the-art CNNs continued in 2015 when He et al. [He+16] designed the record-breaking *ResNet* CNN, one of the versions of which contains 152 hidden layers. The convergence of so deep CNNs is not possible without the *residual blocks* which are the main novelty in the approach of He et al., and which also gave the name to the CNN architecture. The key idea of *ResNet* is very simple: the authors noticed that the intermediate layers of a deep CNN fail to *implicitly* learn the identity mapping. Hence, He et al. added *explicit* identity connections from the input to the output of a residual block allowing to circumvent the weighted connections (*cf.* Figure 2.3.4). As illustrated in Figure 2.3.4, mathematically, it means that instead of learning an original mapping $H(x)$, a *ResNet* layer learns the residual mapping $F(x) = H(x) - x$.

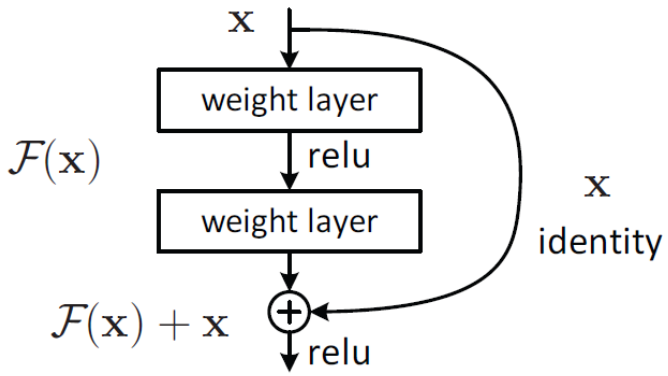



Figure 2.3.4 – (Extracted from [He+16]). Residual blocks which allow the intermediate layers of *ResNet* CNN to learn the residual mapping $F(x) = H(x) - x$ instead of the original one $H(x)$.

In this manuscript, for different tasks, we employ *AlexNet*, *VGG* and *ResNet* CNN architectures, they are highlighted in bold in Table 2.3.1.

It is important to precise that despite the state-of-the-art CNN architectures are evaluated and compared between each other on the *ImageNet* dataset, general image classification is by far not the sole application of CNNs. Thus, transfer learning (introduced in Subsection 2.2.4) allows adapting *ImageNet* CNNs from Table 2.3.1 to other visual problems even with limited training data. More generally, as it has already been mentioned above, CNNs are presently used almost in all domains of computer vision. For example, in the domain of object detection and object localization, the performances have been continuously growing with region-based CNNs [Gir+14; Red+16]. Similarly, semantic segmentation is currently unimaginable without fully convolutional CNNs [LSD15; Lia+16]. CNNs have been successfully applied not only for image analysis, but also for image enhancement: in [Don+14] for super-resolution and in [Pat+16] for inpainting. On multiple occasions, CNNs have been used for video processing including video classification [Bac+12; Kar+14], video prediction [MCL16] and optical flow learning [Fis+15]. Not to mention, that the human-level results of CNNs for automatic face recognition [Tai+14; SKP15] have been among the main motivations behind the present PhD thesis. Finally, in Chapter 5, we separately present and analyse the existing CNN-based approaches which are directly related to the subject of the thesis: gender recognition and age estimation from face images.

Today, CNNs are started to be used not only for pure vision tasks, but also for the problems which lie on the frontiers of several domains. For instance, very promising studies on image captioning [Vin+15; JKFF16] and visual question answering [MRF15; Wu+16] have emerged from the mixture of computer vision and natural image processing. In the same spirit, the symbiosis of CNNs and reinforcement learning has resulted in AlphaGo program [Sil+16] which defeated the world Go champion [3], and in artificial agents which learn to play complex 3D video games such as DOOM [4] based exclusively on high-dimensional sensory streams [DK17].

3. https://en.wikipedia.org/wiki/AlphaGo_versus_Lee_Sedol
4. A legendary first person shooter video game. First version was released in 1993.

## 2.4 Deep Generative Models

CNNs, which were presented in Section 2.3, are examples of *discriminative models* meaning that they are designed to predict a target label given an observation (*i.e.* an image) at their input. In Chapter 5, CNNs are used to address the first objective of this manuscript, the one of predicting gender and age from static face images.

At the same time, the second manuscript objective concerns the synthesis of new face images with the required labels (*i.e.* gender and age). This problem cannot be solved with discriminative models, but instead, it requires the usage of *generative* ones. Indeed, generative models learn to imitate the joint distribution of images and labels, and, unlike discriminative ones, allow to sample from this distribution.

More formally, if we denote the human faces by $x$ and the studied face labels (gender and age) by $y$, we may assume that all natural human faces follow some (unknown) joint distribution $p(x,y)$. According to this notation, discriminative models (for example, CNNs) learn to estimate the conditional distribution $p(y|x)$, while generative ones are focused on modelling of the joint distribution $p(x,y)$.

Below, we briefly introduce the most notable existing deep generative models in Subsection 2.4.1, and then, in Subsection 2.4.2, we focus on Generative Adversarial Networks (GANs) [Goo+15], a particular class of generative models which we further use for face synthesis and editing in Chapter 6.

### 2.4.1 Overview of Deep Generative Models

As already mentioned in the introductory Section 2.1, generative models are currently an area of active research in the deep learning community. Below, we give a general overview of different families of existing deep generative models focusing on their advantages and downsides.

In this manuscript, we focus on *conditional* generative models because our goal is the synthesis of face images with the required gender and age conditions rather than generating arbitrary faces. More formally, the objective of a conditional generative model is to learn the joint probability distribution $p(x,y)$ of data (*e.g.* images) $x$ and attributes or conditions (*e.g.* gender and age) $y$ based on the labelled training dataset $(x^{(i)},y^{(i)})$, $i \in \{1,\dots,N\}$, where $N$ is the dataset size. Usually, this is done via the principle of *Maximum Likelihood* which consists in finding the set of model parameters $\theta$ maximizing the probability which the model assigns to the training data:

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \prod_{i=1}^{N} p_{model}(x^{(i)},y^{(i)};\theta) \tag{2.4.1}$$

Goodfellow proposed [Goo16] a simple taxonomy which organizes deep generative models into distinct families based on how they address the stated problem of Maximum Likelihood. The taxonomy as well as examples of models belonging to each family are presented in Figure 2.4.1.

The left branch of the presented taxonomy corresponds to the deep generative models which explicitly estimate the density $p(x^{(i)},y^{(i)};\theta)$ in order to solve the optimization problem 2.4.1. Depending on whether it is feasible to precisely calculate this density, the "explicit" branch is further split into "tractable" and "approximative" parts.

Recently proposed PixelCNN [Oor+16] is a canonical example of a model which defines the probability density in a tractable manner. In particular, PixelCNN assumes that the conditional probability

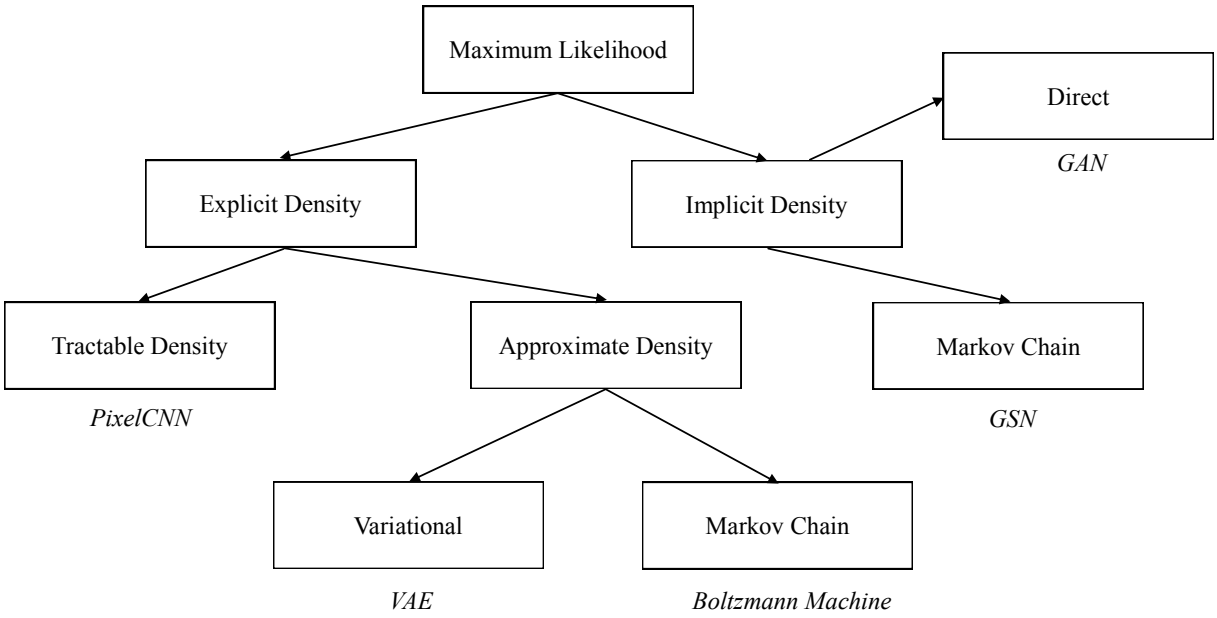



Figure 2.4.1 – (Reproduced from [Goo16]). Taxonomy of deep generative models.

of every pixel $x_i$ in an image depends only on the conditions $y$ and the previous pixels $(x_1, \ldots, x_{i-1})$ (the pixel dependencies are in raster scan order, *i.e.* from left to right and from top to bottom):

$$p(x|y) = \prod_{i=1}^{n^2} p(x_i|x_1, \ldots, x_{i-1}, y) \tag{2.4.2}$$

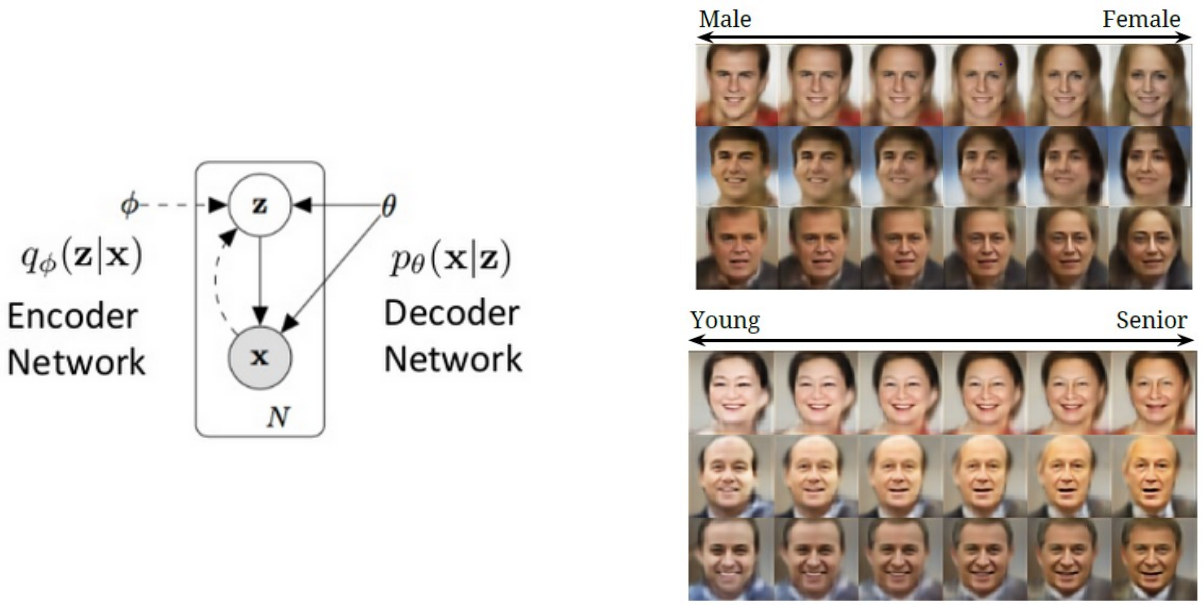
Note that the joint density distribution can be estimated as $p(x,y) = p(x|y)p(y)$ given the fact that it is usually considered that the conditions $y$ follow some predefined regular distribution (for example, uniform or Gaussian).

Despite PixelCNNs can synthesize very plausible images and excel in the task of image synthesis, their major drawback is the time which is needed to generate samples. For example, it takes about 11 minutes to generate 16 RGB images of size 32x32 on a modern GPU according to the approach described in [Oor+16].

Unlike PixelCNN, the probability density functions in Boltzmann Machines (BMs) [HSA84] and in Variational AutoEncoders (VAEs) [KW14] are computationally intractable and therefore, must be approximated to solve the optimization problem 2.4.2. Thus, BMs rely on Markov chains both for training and for sample generation. As mentioned in Subsection 2.2.4, BMs played a very important role in resurrection of interest in deep learning in 2006. However, these models are currently out of favour because of the difficulty to scale them for large datasets (such as *ImageNet*).

On the contrary, VAEs are one of the three (among GANs and PixelCNNs) most popular generative models at the time of writing of the present manuscript. Vanilla (*i.e.* non-conditional) VAE is an autoencoder which is composed of two ANNs (usually, CNNs in the context of image modelling): an *encoder* $q(z|x;\phi)$ and a *decoder* $p(x|z;\theta)$ (*cf.* Figure 2.4.2-(a)).

Vanilla VAE assumes that the probability density $p(x)$ of natural images $x$ depends on a latent variable $z$ with a predefined (usually, Gaussian) prior density $p(z)$. More formally, the log-likelihood




(a) (b)

Figure 2.4.2 – (a) (Extracted from [KW14]). Schematic presentation of Variational AutoEncoder (VAE). (b) (Extracted from [Yan+16]). Examples of cVAE-generated synthetic face images with varying gender and age conditions.

$\log p(x)$ of observing an image $x$ can be calculated as following:

$$\log p(x) = \sum_z q(z|x) \log p(x) = \sum_z q(z|x) \log \frac{p(x,z)}{p(z|x)}$$
$$= \sum_z q(z|x) \log \frac{p(x,z)}{q(z|x)} \frac{q(z|x)}{p(z|x)} = \sum_z q(z|x) \log \frac{p(x,z)}{q(z|x)} + KL[q(z|x)||p(z|x)] \quad (2.4.3)$$

In the equation above, the encoder $q(z|x) = q(z|x;\phi)$ approximates the intractable true posterior $p(z|x)$, while the decoder models the conditional distribution $p(x|z) = p(x|z;\theta)$.

The second term of the log-likelihood 2.4.3 is the Kullback–Leibler divergence $KL[q(z|x;\phi)||p(z|x)]$ illustrating the closeness of the true posterior $p(z|x)$ to its approximation $q(z|x;\phi)$. While infeasible for calculation, this term is always non-negative (due to the properties of the Kullback–Leibler divergence). Therefore, $\log p(x) \geqslant \sum_z q(z|x;\phi) \log \frac{p(x,z;\theta)}{q(z|x;\phi)} = L(\theta,\phi,x)$, where $L(\theta,\phi,x)$ is called *variational lower bound*.

Training of a VAE requires the maximization of the variational lower bound $L(\theta,\phi,x)$ (which gave the name to the generative model itself). We invite the interested readers to consult the original paper [KW14] for details on how $L(\theta,\phi,x)$ is maximized in practice.

Obviously, a vanilla VAE can be extended to a conditional VAE (cVAE) which approximates the joint probability density $p(x,y)$ by integrating the conditions $y$ to the log-likelihood maximization 2.4.3 (details can be found in [Yan+16]). Once a cVAE is trained, its decoder $p(x|z,y) = p(x|z,y;\theta)$ can be used to synthesize images. To this end, it is enough to sample an arbitrary latent vector $z$ from the latent distribution with the predefined prior $p(z)$: $z \sim p(z)$, and set the conditions (*i.e.* attributes) $y$. Examples of cVAE-generated face images from [KW14] with different latent vectors $z$ and gender/age conditions

$y$ are presented in Figure 2.4.2-(b).

VAEs (and cVAEs) are theoretically sound generative models which are relatively easy to optimize and which often converge to very good log-likelihood values [Goo16]. Nevertheless, the quality of the synthetic images produced by VAE is significantly lower that of PixelCNN and GAN. In particular, VAE-generated images are known to be blurry and to contain little details (*cf.* Figure 2.4.2-(b)).

Up to now, we have discussed only the left branch of the taxonomy presented in Figure 2.4.1, where the generative models have an explicit representation for the joint probability density function $p(x,y)$. On the contrary, the right branch of the taxonomy presents other generative models which, instead of directly defining of the probability density, offer a mechanism of indirect interaction (for example, sampling) with $p(x,y)$. Thus, Generative Stochastic Networks (GSNs) [Ben+14] learn a Markov chain operator which, when it is run multiple times, produces samples from the model's joint distribution. However, similarly to BMs, GSNs fail to scale for high dimensional problems, and similarly to PixelCNNs, they are computationally costly at test time.

Finally, training algorithm of GANs, which are introduced in Subsection 2.4.2, requires only the model's ability to generate samples. Moreover, unlike GSNs, GANs scale well for high dimensional problems, and they can produce one sample at a time (*i.e.* no need for Markov chains).

### 2.4.2 Generative Adversarial Networks

As it has already been mentioned in Subsection 2.4.1, the objective of a GAN is to model an unknown image distribution via the ability to sample from it. The basic idea behind GANs is so unusual and simple that Y. LeCun called them one of the most important recent developments in deep learning [5].

Informally, training of a GAN sets a game between two players, namely: a *generator* and a *discriminator*. The generator learns to draw synthetic images which should resemble the natural ones, while the discriminator learns to distinguish them. Training of the two players is done in parallel, and in order to succeed in the game, the generator must constantly improve its drawings, while the discriminator must look for more and more fine details to find the differences between the "fake" and "authentic" images. This competition between the players is very important, and training of one is not possible without the other (that is why the model is called "adversarial"). The whole process is illustrated in Figure 2.4.3.

**Definition** More formally, a vanilla (non-conditional) GAN [Goo+15] is a pair of differentiable functions: the first one is called the generator $G(z)$ and the second one is called the discriminator $D(x)$. These functions are modelled by ANNs (CNNs in the context of images) with the weights $\theta_G$ and $\theta_D$, respectively.

Similarly to VAEs, GANs assume that the natural image distribution $p_{data}(x)$ is conditioned on some predefined prior distribution $p(z)$. The generator of a GAN plays the same role as the decoder of a VAE. Thus, the generator maps vectors $z$ from the latent space $N^z$ to the image space $N^x$ ($G(z) : N^z \to N^x$).

The discriminator maps vectors from the image space $N^x$ to scalar values $\mathbb{R}$ ($D(x) : N^z \to \mathbb{R}$). More precisely, $D(x)$ represents the probability that an image $x$ is sampled from the image distribution $p_{data}(x)$ rather than from the generator distribution $p_G(x)$. Therefore, the discriminator is trained to assign high probabilities to the natural images $x \sim p_{data}(x)$ (*i.e.* from the training dataset) and low probabilities to the

5. The quote can be found here: http://qr.ae/Tbcybl

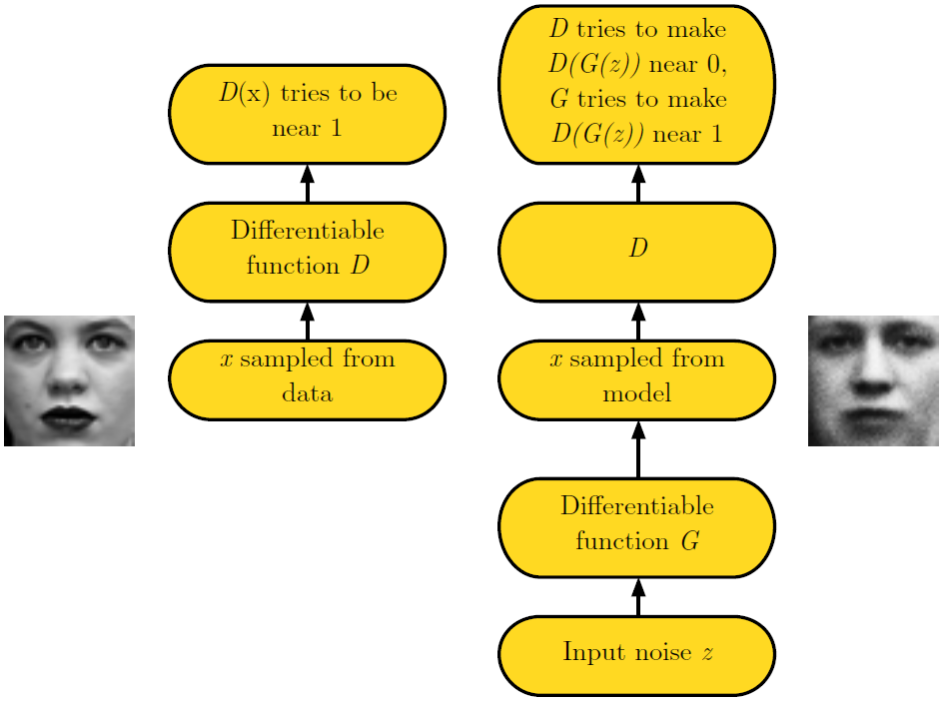



Figure 2.4.3 – (Extracted from [Goo16]). Illustration of the training process of a GAN for generating human face images. Two ANNs, the generator $G$ and the discriminator $D$, are optimized in parallel with the opposite objectives: synthesis of the realistic faces, and discrimination between the synthetic and the natural faces, respectively.

synthetic images $x \sim p_G(x)$ (*i.e.* generated by $G$). At the same time, the generator $G$ is trained to fool $D$ trying to imitate the distribution $p_{data}(x)$ of natural images.

In other words, $D$ and $G$ play the following two-player minimax game with the value function $V(\theta_G, \theta_D)$:

$$\min_{\theta_G} \max_{\theta_D} V(\theta_G, \theta_D) = \mathbb{E}_{x \sim p_{data}}[\log D(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))] \tag{2.4.4}$$

Of course, a vanilla GAN can be extended to model the joint probability $p(x,y)$ of images $x$ and conditions $y$. Arguably, the most natural way to do it was independently proposed in [MO14] and in [Gau14]. Both works modify the minimax problem 2.4.4 by introducing the vector of conditions $y \in \mathbb{R}^{N_y}$ as additional input of both $G$ and $D$. The value function $V(\theta_G, \theta_D)$ of the conditional GANs (cGAN) is presented below:

$$\min_{\theta_G} \max_{\theta_D} V(\theta_G, \theta_D) = \mathbb{E}_{x,y \sim p_{data}}[\log D(x,y)] + \mathbb{E}_{z \sim p_z(z), \tilde{y} \sim p_y}[\log(1 - D(G(z,\tilde{y}),\tilde{y}))] \tag{2.4.5}$$

**Convergence Issues** Today, GANs (and cGANs) are largely praised as the generative models which produce the most variative and visually plausible images [RMC16; Lar+16]. However, training of GANs is notorious to be highly unstable and difficult to control [Sal+16].

Indeed, the particularity of GAN optimization is the fact that the generator and the discriminator must balance each other during the whole process of the mini-max optimization (*cf.* Equation 2.4.4). If at one moment of training, either generator or discriminator outperforms its respective opponent too

much, the loss function will saturate and the training process stops.

For example, one particularly common failure scenario is when the gradients of all synthetic images point at the same area of the image space which is believed to be highly realistic by the current state of the discriminator. This makes the generator collapse to a state when it always outputs the same image (the one which is believed realistic). After the collapse has occurred, the discriminator rapidly understands that the regularly repeated image is the synthetic one, but the gradient descent optimization cannot split the identical outputs of the generator. This problem happens so often that it has even been named *Helvetica Scenario* [Goo+15].

In order to avoid Helvetica Scenario and other problems with the convergence of GANs, a number of heuristics have been found by the deep learning community. Many of them are intuitive and are the results of numerous trials and errors. Thus, there was even a tutorial organized at NIPS 2016 [6] which was entitled "How to Train a GAN? Tips and tricks to make GANs work".

Many of such GAN training hints are summarized in the Deep Convolutional GAN (DCGAN) which was designed by Radford et al. [RMC16], and which was one of the first successfully trained GANs using deep CNN architectures both for the generator and the discriminator. The key principles of the DCGAN architecture are described by its authors as following:

1. Strided and fractional-strided convolutions are used instead of pooling and upsampling layers.
2. Batch normalization is used both for $G$ and $D$.
3. Both $G$ and $D$ are fully-convolutional CNNs (*i.e.* without fully-connected layers).
4. ReLU is used as the activation function in all layers of $G$ except for the output layer where the hyperbolic tangent activation is employed.
5. LeakyReLU is used as the activation function in all layers of $D$.

These basic principles of the DCGAN architecture have appeared to be applicable both for vanilla GANs and for cGANs and have allowed training of the models for the large variety of different applications (some of them are cited below). In Chapter 6 of the present manuscript, we also apply DCGAN for synthesis and editing of face images with the required gender and age conditions.

**Applications** Due to the high visual fidelity of the generated images, GANs have attracted a lot of attention of the computer vision community.

Arguably, the most straightforward application of GANs is the generation of synthetic (labelled, in case of cGANs) training data for the domains where the data collection is costly. Thus, Sixt et al. created a synthetic dataset of images of barcode-like markers that are attached to honeybees [SWL16]. In the same spirit, Tan et al. [Tan+17] proposed a GAN-model to synthesize paintings which imitate well-known artists.

Contrary to VAE, GAN is not an autoencoder and does not have an explicit mechanism for reconstructing an input image. Nevertheless, a number of studies proposed different approaches to circumvent this limitation [Zhu+16; Lar+16; Dum+17] (more details are provided in the beginning of Chapter 6). As a result, GANs (and cGANs) have been successfully applied for different types of natural image editing

6. International conference on advances in Neural Information Processing Systems (NIPS) is one of two major annual scientific meeting on deep learning. The summary of the tutorial on GANs in NIPS 2016 can be found here: https://github.com/soumith/ganhacks.

and enhancement, such as manipulating the image content [Zhu+16; Per+16], domain (style) transferring [Iso+17; TPW17], inpainting [Yeh+16] and super-resolution [Led+16]. Inspired by these works, in Chapter 6, we design a cGAN-based approach for editing of human visual demographic traits: gender and age.

Finally, GANs can be used as a part of other non-generative models. Thus, Luc et al. [Luc+16] improved the state-of-the-art image segmentation by integrating an adversarial loss, while Ho and Erman [HE16] used a GAN in the reinforcement learning setting as a part of "imitation learning".

## 2.5 Conclusion

This chapter has been devoted to a brief overview of the most prominent techniques of deep learning for image analysis and synthesis. Therefore, we have mainly focused on the deep learning algorithms and models which are essential for understanding the rest of the manuscript. The more detailed and extensive presentation of the field can be found in the excellent book by Goodfellow et al. [GBC16].

In Section 2.1, we have introduced deep learning as a subdomain of machine learning explaining the difference between the hand-crafted and the learned features. The take-away message of this section is that deep learning is a particular case of representation learning where, in the context of computer vision, the features are learned in a hierarchical manner starting from the elementary edge detectors and up to the complex, problem dependent features.

Deep learning is often presented in the media as a field which has appeared from nowhere during the last decade. However, as it is reported in Section 2.2 of this chapter, the key concepts of deep learning have been developing for more than 60 years. More precisely, in Section 2.2, we have focused on the presentation of the fundamental ideas and algorithms (such as artificial neuron, training via backpropagation, ANN depth, transfer learning etc.) which are the basis of the contemporary deep learning.

CNNs are primary deep learning models which are used in all contributions of the present manuscript. They have been presented in Section 2.3 as the models which have revolutionized various domains of computer vision. In Section 2.3, we have focused not only on the design aspects of different CNN architectures, but also on the largely adopted training principles which significantly ameliorate the CNN convergence and generalization (such as rectified activations, dropout and batch normalization).

Finally, Section 2.4 has been devoted to the introduction of GANs, the generative deep models, which are employed for editing of gender and age perception in human faces in Chapter 6. We have introduced GANs by comparing their advantages and downsides with respect to alternative deep generative models. In particular, our choice of using GANs is motivated by the fact that once trained, they can produce synthetic images of high visual fidelity without significant computational costs.

Chapter 3

# Gender/Age Prediction and Editing from Faces

**Contents**



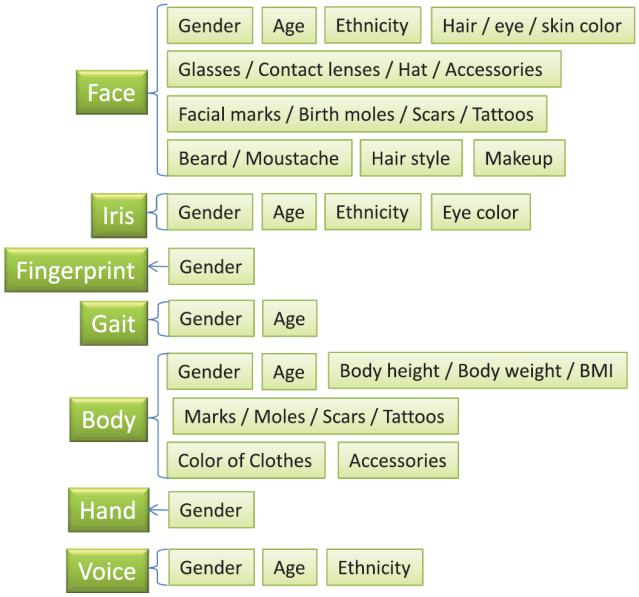
## 3.1 Introduction

Soft biometrics traits are defined by Jain et al. [JDN04] as "*characteristics that provide some information about the individual, but lack the distinctiveness and permanence to sufficiently differentiate any two individuals*". The history of soft biometrics dates back to a French police officer Alphonse Bertillon [Rho56] who created the law enforcement system (*bertillonage*) based on the anthropometric and anatomical characteristics of people in 1888. An example of a real police card of a criminal which was created as a part of bertillonage is presented in Figure 3.1.1-(a).

The contemporary soft biometrics studies a wide spectrum of human traits which are depicted in Figure 3.1.1-(b). The big advantage of soft biometrics with respect to "traditional" biometrics (the goal of which is to identify a person) is that the former provides only a partial description of an individual preserving her/his privacy. This allows to use soft biometrics for anonymized statistics collection.

For example, human gender and age, studied in this manuscript, are particular cases of *soft biometrics* traits which are often called *demographics*, because they are commonly used for population analysis


(a)




(b)

Figure 3.1.1 – (a) A French police card of a criminal which was created as a part of *bertillonage*, the procedure of collecting of the anthropometric and anatomical characteristics of suspects. Bertillonage is often given as a first example of large collection of the soft biometrics data. (b) (Extracted from [DER16]). The list of the soft biometrics traits and the biometric modalities from which these traits are extracted.

[Dan+11]. Recognition and synthesis of gender and age is an active research area involving the studies with various data modalities such as voice [MP09], iris [Tho+07], body [LT10], hand [Sha11] and others (*cf.* Figure 3.1.1-(b)).

However, among all human-related data modalities, the face (which was poetically named the "window to the soul" [Zeb97] by Zebrowitz) is the richest source of information about a person [RNJ07]. Indeed, apart gender and age, a human face can reveal a person's identity [Zha+03], mood [TKC11], ethnic origin [XLS12] and many other details. Moreover, as mentioned in Chapter 1, the development of social networks has dramatically increased the amount of face images (especially of celebrities) which are publicly uploaded in the Internet. Not only does it underpin the practical interest of creating automatic systems of face analysis, but it also offers an opportunity to train complex machine learning models (for example, deep ANNs), which was not possible even a decade ago. This explains why we have chosen human face images as the input data for predicting and artificially synthesizing of gender and age.

More formally, face is one of the primary aspects of the sexual dimorphism (*i.e.* the secondary sex characteristics allowing to distinguish men and women) [LI00]. Indeed, an average female face is rounder than an average male one, while men often have more facial hair than women. Nevertheless, Loth and Iscan [LI00] showed that not a single face characteristic can be solely used to confidently recognize gender. Moreover, the difficulty of gender recognition can be significantly increased by the presence of make-up and facial accessories (eyeglasses, scarf, etc.) Despite all that, humans are very good in recognizing gender from faces [IS13], because of its importance for social interactions.

As a side remark, it is important to notice that from the linguistic point of view, human *sex* refers to an ensemble of biological characteristics which differentiate men and women, while the notion of human

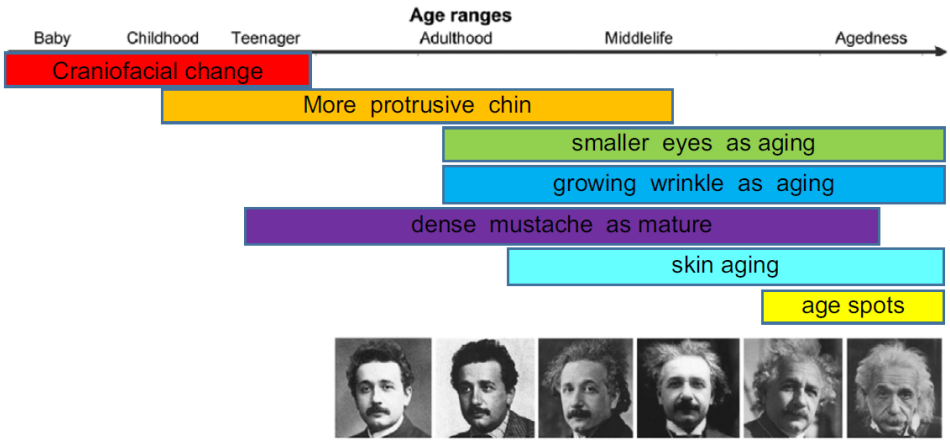



Figure 3.1.2 – (Extracted from [Shu+16].) An illustration of human aging progress. At various stages of life, aging affects different face parts.

*gender* is more related to socio-cultural aspects which are associated with two sexes. Nevertheless, in the present manuscript, we follow the majority of the previous computer vision studies on automatic estimation of biological sex from face images which used the term "gender recognition" when describing their approaches [DER16].

Gender and age are two principally different biological characteristics as the former one is a part of a person's identity, while the latter one changes throughout the person's life. However, similarly to the ability to recognize gender, since the early childhood, humans develop a skill of estimating an age from the face with a high accuracy [GH95]. The facial cues revealing a person's age are different for children, adults and seniors. As illustrated in Figure 3.1.2, the craniofacial changes are characteristic only for children and teenagers. Chin gradually becomes more salient starting from the teenage and until the middle age. Finally, the skin alterations such as wrinkles and age spots are typical for the senior age.

Finally, unless said otherwise, in the present manuscript, when we speak about automatic age estimation, we implicitly understand predicting of a person's *biological age*, or in other words, the time since the person's birth date (for example, in years). Nevertheless, a slightly different problem of *apparent age* estimation is also recognized by the research community [FGH10], and in 2015, there was even the first public competition on the subject [Esc+15]. Apparent age of a face is defined as the age which would be perceived by an average human looking at this face, and in practice, it is calculated by averaging several human annotations. Real use cases of automatic age estimation usually require estimation of the biological age rather than the apparent age, and therefore, the previous studies mostly focused on the former one. In Chapter 5, we demonstrate that the two problems are actually highly correlated, and a model for biological age estimation can be effortless adapted for estimation of apparent ages.

The rest of this chapter is dedicated to the literature overview of two main problems which are addressed in this PhD: gender/age prediction from face images (*cf.* Section 3.2), and gender/age editing in face images (*cf.* Section 3.3). Below, we explicitly focus only on the conventional approaches which are not based on the classes of deep models further used in this manuscript (*i.e.* CNNs for gender/age prediction and neural generative models for face editing). Instead, we separately present the remaining (most recent) works on the considered subjects at the beginnings of Chapters 5 and 6 (because the on-place comparison allows us to highlight in what aspects our contributions improve existing alternatives).

## 3.2 Gender/Age Prediction from Face Images

This section presents a typical pipeline of gender/age prediction systems and makes an overview of the existing algorithms for addressing the considered problems in Subsections 3.2.1 and 3.2.2, 3.2.3, respectively. We end this section by illustrating some potential areas of application for the systems of automatic prediction of gender/age from faces in Subsection 3.2.4.

It is important to highlight that below, we deliberately do not provide the gender recognition and age estimation accuracies of the described approaches from the original articles. The reason is that there is a big discrepancy between the used testing datasets and the various evaluation protocols (*i.e.* same-dataset, cross-validation, cross-dataset), which makes the resulting scores incomparable between each other. This is especially true for early works on the subjects. Instead, in Tables 5.3.2 and 5.3.3 of Chapter 5, we summarize all scores which were reported on three most popular contemporary gender and age benchmark datasets with well established evaluation protocols.

### 3.2.1 Standard Pipeline for Gender and Age Prediction from Face Images

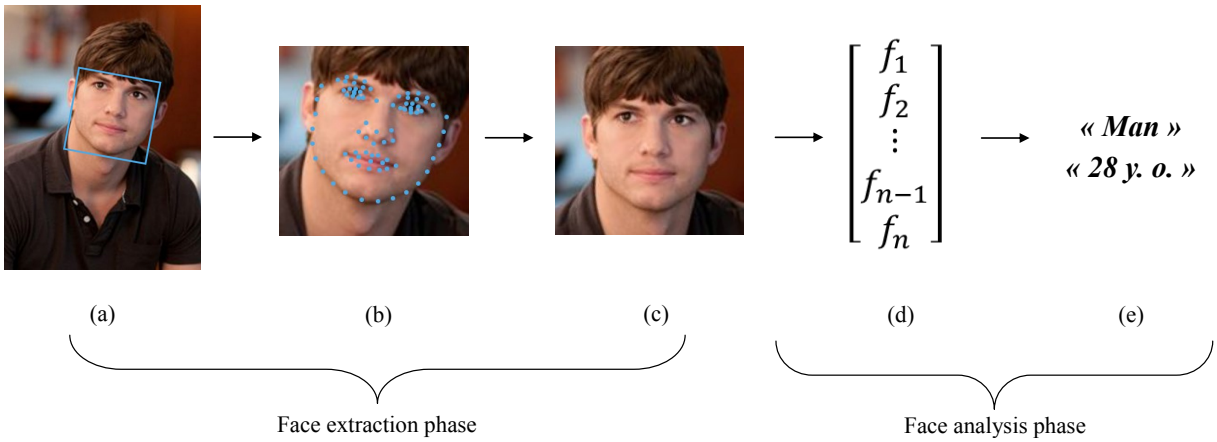



Figure 3.2.1 – Typical pipeline of automatic gender recognition and age estimation systems. It consists of two phases (face extraction, and face analysis) and five steps: (a) face detection, (b) face landmark detection, (c) face alignment, (d) feature extraction, and (e) gender/age prediction.

Automatic gender recognition and age estimation systems usually follow a typical pipeline which is presented in Figure 3.2.1. It consists of two principle phases, namely: (1) face extraction (or image preprocessing), and (2) face analysis. The objective of the first phase is to extract a face (in a predefined form) from an input image, while the objective of the second phase is to predict gender and age based on the extracted face.

In its turn, face extraction phase is normally composed of three smaller steps. The first one, face detection, is indispensable and is present in all systems of automatic face analysis. Basically, its goal is to detect a face (or faces) in an input image and to output the respective delimiting region (or regions) in the image. Face detection is a classical problem of computer vision with plenty of existing open-source solutions such as [VJ01; Mat+14]. It should be noted that the form of the delimiting face region (square, rectangle, oval etc.) depends on the particular face detector. For example, in this manuscript, we employ a private face detector which is based on [Zha+07] and outputs square-sized face regions.

Sometimes, the output of a face detector is directly given to the face analysis component of gender/age prediction system. But more often, the input faces are expected to be in a certain normalized form. For example, a face image can be normalized by rotating it so that the two eyes lie on a horizontal line, and by scaling the resulting image in order to set the distance between the eyes in pixels to some predefined value. The described normalization method is simple but it is very sensible to precise estimation of the position of the eyes. Instead, a much more general and robust approach consists in detecting a set of landmarks (their exact number depends on the particular implementation) and then performing a (2D or 3D) affine transformation of the detected set to the set of the predefined landmark positions. This process is called *face landmark alignment* and is illustrated in Figure 3.2.1 (steps (b) and (c)). 2D (or 3D) face rotation and scaling is done implicitly by face alignment as a part of the affine transformation.

If not said otherwise, in the present manuscript, we use private face detection and 2D face alignment solutions of Orange which are based on [Zha+07] and [Bel+13], respectively. At the same time, the contributions of this work concern the face analysis phase from the pipeline in Figure 3.2.1, which is discussed below.

Being composed of two distinct steps: feature extraction and gender/age prediction, the face analysis phase of the pipeline in Figure 3.2.1 is the typical one for the majority of non-deep learning approaches for object recognition in computer vision. As already discussed in the introductory Section 2.1 of Chapter 2, the features representing the input data have a decisive impact on the effectiveness of the subsequent machine learning algorithms for classification or regression. Therefore, the previous studies on gender recognition and age estimation tried a large variety of approaches for gender/age-aware feature extraction from face images. We make an overview of the attempted methods in Subsection 3.2.2. After that, in Subsection 3.2.3, we report on different machine learning algorithms which have been tested on the extracted face features in order to predict gender and age.

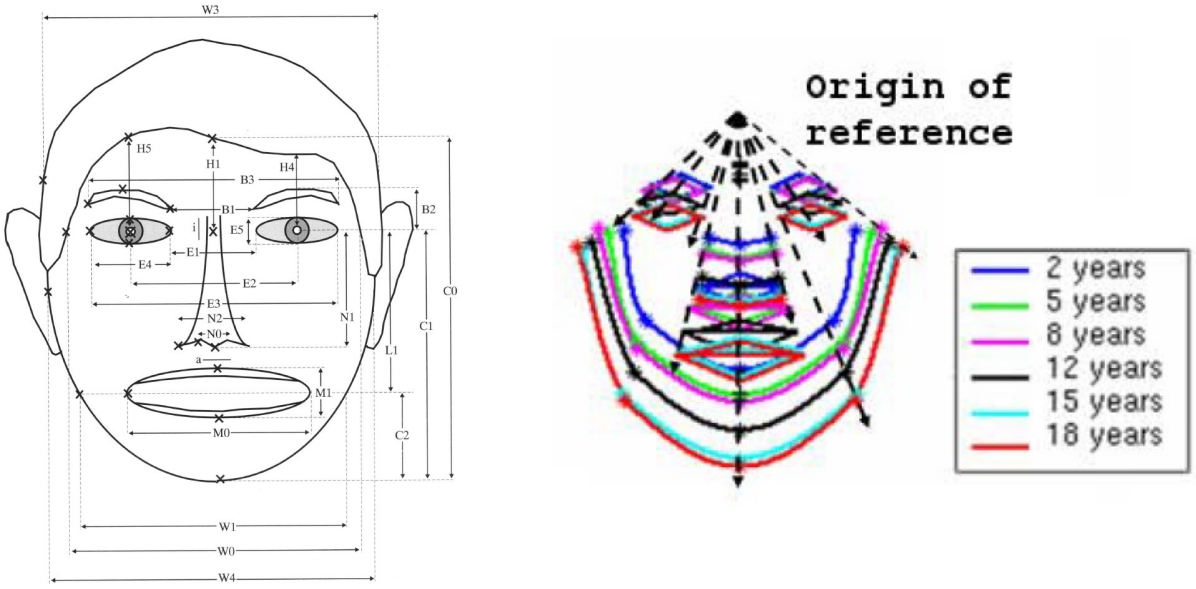
### 3.2.2 Gender/Age-Aware Feature Extraction

Below, we present different approaches for description of face images (*i.e.* feature extraction) which have been utilized prior to gender recognition and age estimation in previous studies. We have organized the feature extraction methods based on their nature following the methodologies from [FGH10; Han+15; DER16].

#### 3.2.2.1 Anthropometry-Based Features

Anthropometry-based features are a set of distances and ratios which are calculated between fiducial (landmark) points in a frontal normalized face. The idea is to use these distances in order to describe the topological differences between male and female faces or between faces of different ages. Basically, the anthropometric features derive the *geometric* dimensions of the *skull* based on the provided face image.

For example, one of the first attempts to distinguish cranial shapes of male and female faces was performed by Poggio et al. [PBP92]. To this end, the authors used 15 fiducial distances: pupil to eyebrow separation and nose width appeared to be the most discriminative among them. Later, Fellous [Fel97] extended the previous study [PBP92] by proposing 24 fiducial distances for gender recognition (the selected distances are illustrated in Figure 3.2.2-(a)). The conclusions of Fellous are complementary to the ones Poggio et al. Indeed, in addition, to the distance between the eyes and the eyebrows and to




(a) (b)

Figure 3.2.2 – (a) (Extracted from [Fel97]). The set of 24 face fiducial distances for anthropometric prediction of gender. (b) (Better viewed in color). (Extracted from [RC06b]). Growth pattern of a child's skull during the aging.

the width of the nose, the total width of the face and the distance between the two eye pupils were also found useful for gender recognition in [Fel97].

As it is presented in Figure 3.1.2, the cranial changes are characteristic only for children and teenagers under 18 years old. Therefore, anthropometric features are useful only for age estimation of minors. Thus, Ramanathan and Chellappa [RC06b] defined a growth pattern based on 8 fiducial proportions which models the children age progression (*cf.* Figure 3.2.2-(b)). Similarly, Gunay and Nabiyev [GN07] applied anthropometry-based features for age estimation from children faces.

Finally, a common downside of anthropometric features is the fact that they are very sensitive to precise estimation of the fiducial points in the face and to the face pose [Han+15]. In other words, if an input face is not well aligned, or a part of the face is occluded, the anthropometric features become almost useless. Due to this limitation, these features are rarely used for gender recognition and age estimation in real-life applications.

#### 3.2.2.2 Texture-Based Features

Many studies on gender recognition and age estimation from face images are based on extraction of both holistic and local texture information. The most straightforward way to achieve that is to directly use the pixel intensities. This simple approach was employed by several gender recognition studies [GWP98; MY02; BR07] with various classification algorithms. Raw pixels contain a lot of redundant information which can be removed using dimensionality reduction methods. To this end, Khan et al. [KMM05] used Principal Component Analysis (PCA) [WEG87] while Jain and Huang [JH04] employed Independent Component Analysis (ICA) [HKO04] in the context of gender recognition. Age estimation is a more sophisticated problem than gender recognition (we elaborate more on that point in Chapter 5), and it was shown by Guo et al. [Guo+08] that pixel intensities are hardly employable for age estimation even after

dimensionality reduction with PCA and ICA. Instead, the authors proposed using manifold learning to extract texture features for age estimation (*cf.* Paragraph 3.2.2.3 for details).

The other popular possibility to extract texture information from a face image is the usage of the general-purpose hand-crafted features of computer vision. For example, Local Binary Patterns (LBP) [OPH96] are one of the most basic and popular hand-crafted features. They were broadly utilised for both problems considered in the present section. Interestingly, LBP features appeared to be much more effective for gender recognition [Sha12; TP13; JC15] than for age estimation [YA07; GN08]. On the contrary, Biologically Inspired Features (BIF) [RP99] were attempted for gender recognition in [Han+15], but they proved to be particularly effective for age estimation [Guo12] which was confirmed in a number of works [GM10; GM11; GM14].

Some other hand-crafted features were also tried for gender recognition and age estimation, though less frequently than LBP and BIF. For example, Wang et al. [Wan+10] employed Scaled Invariant Feature Transforms (SIFT) [Low99] for gender recognition, Gabor filters [FS89] were used by Xia et al. [XSL08] for gender recognition and by Liu and Wechesler [LW02] for age estimation, and Haar-like features [VJ01] allowed Zhou et al. [Zho+05] to train a boosting model for age estimation.

Moreover, a very promising approach is combining various texture-based features in one model. Thus, in the recent work by Castrillón-Santana et al. [CS+16], the authors analysed and compared different methods of fusion of various hand-crafted features, including LBP, Histogram of Oriented Gradients (HOG) [DT05], Weber Local Descriptors (WLD) [Che+10] and others, in one gender recognition model. Similarly, Moeini et al. [Moe+17] combined LBP features and raw pixel intensities extracted from different regions of faces to learn a regression dictionary for gender recognition and age estimation. In the same spirit, Liu et al. [LYK15] combined LBP, HOG and BIF features to train a hierarchical age estimation model obtaining very good performances.

#### 3.2.2.3 Manifold Learning Features

The objective of the manifold learning in the context of gender recognition (age estimation) is to find a low dimensional manifold as well as the projection to this manifold from the space of face images which together allow to separate manifold embeddings corresponding to face images of different genders (ages). Contrary to hand-crafted features discussed in Paragraph 3.2.2.2, manifold learning is a way to obtain *learned features* based on the training images. In fact, PCA and ICA dimensionality reduction approaches, which are mentioned in Paragraph 3.2.2.2, can be seen as elementary linear manifold learning techniques. However, in this paragraph, we discuss studies which employed two alternative non-linear manifold learning algorithms for gender recognition and age estimation.

Thus, Hadid and Pietikainen proposed [HP09] Local Linear Embedding (LLE) [RS00] to learn features for gender recognition. The LLE algorithm is a non-linear manifold learning approach which exploits local symmetries of class reconstructions. LLE face representations allowed the authors improving the gender classification accuracy by about 10 points with respect to the LBP baseline [HP09].

Despite the mentioned success of the LLE algorithm for gender recognition, Guo et al. [GM11] demonstrated that LLE fails to project face images to sufficiently discriminative subspaces for age estimation. Instead, they employed Orthogonal Locality Preserving Projections (OLPP) [Cai+06] which is yet another manifold learning approach. The particularity of OLPP is that is preserves the initial space

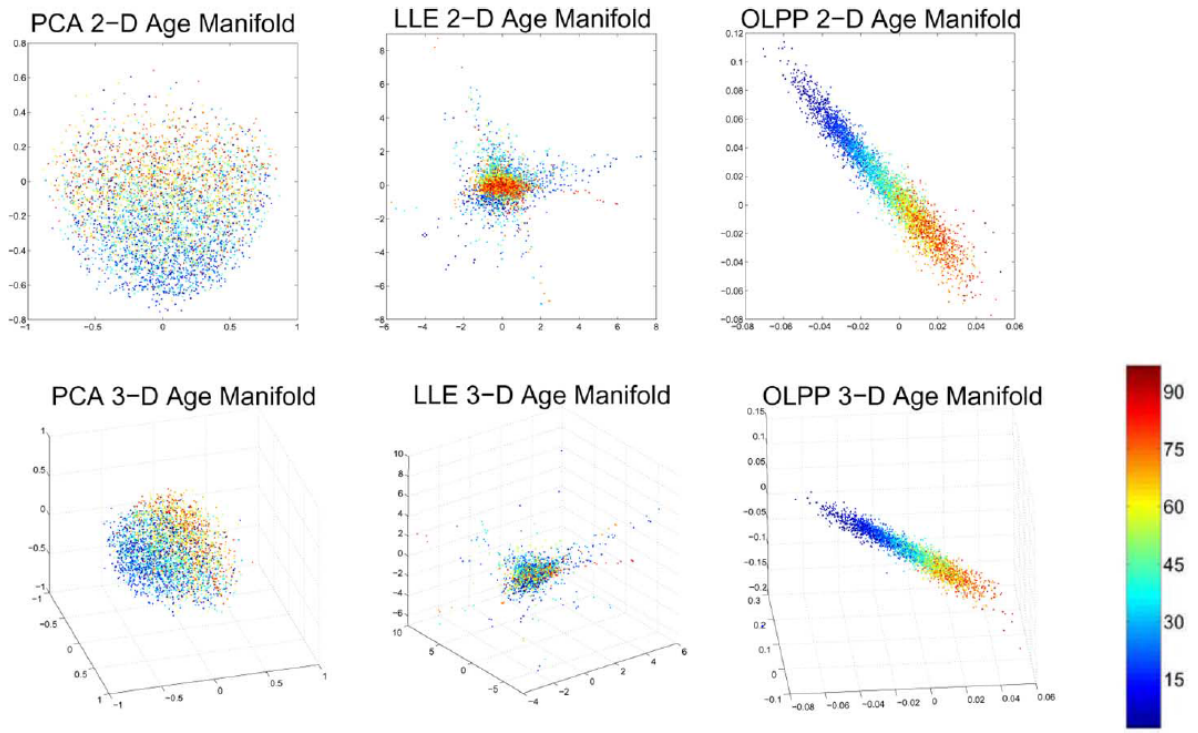



Figure 3.2.3 – (Better viewed in color). (Extracted from [Guo+08]). Illustration of 2D and 3D age manifolds learned with three different manifold learning techniques: PCA, LLE, and OLPP. The colours of points indicate the age of people in the respective face images.

structure by evaluating local neighbourhood distances. Figure 3.2.3 illustrates examples of 2D and 3D age manifolds learned by PCA, LLE and OLPP, and one can clearly see that OLPP much better separates faces of different ages than alternative manifold learning algorithms.

#### 3.2.2.4 Appearance-Based Features

Finally, the appearance features are based both on shape and texture information.

A typical method for extracting appearance features is Active Appearance Models (AAM) which was initially proposed for image coding [CET+01]. Using the training dataset, AAM separately applies PCA to learn a statistical shape model and an intensity model of face images. Lanitis et al. [LTC02] extended AAM for age modelling by proposing an aging function to explain variations in ages. Later, AAM was independently applied for gender recognition by Xu et al. [XLS08] and by Shih [Shi13]. Contrary to anthropometric features, AAM-based features can deal with all age categories, and not only with children.

The famous AGing pattErn Subspace (AGES) algorithm for age estimation [GZSM07] also uses AAM. The basic idea of AGES is to model the aging pattern, which is defined as a sequence of a particular individual's face images sorted in time order, by constructing a representative subspace. The proper aging pattern for a previously unseen face image is determined by the projection in the subspace that can reconstruct the face image with minimum reconstruction error, while the position of the face image in that aging pattern will then indicate its age. In AGES, each face is firstly encoded with AAM-based features. However, the practical usage of AGES is strongly limited by the fact it assumes the

existence of multiple face images of the same person in the dataset.

In general, the appearance features which are extracted with AAM suffer from imprecise estimation of fiducial points similarly to the anthropometric features.

### 3.2.3 Gender/Age Prediction

In this subsection, we focus on the last step of the face processing pipeline presented in Figure 3.2.1, *i.e.* on algorithms which perform gender/age prediction based on the image descriptors discussed in Subsection 3.2.2. In Paragraph 3.2.3.1, we introduce the principal prediction algorithms which were utilized for the studied problems, while in Paragraph 3.2.3.2, we discuss the evaluation metrics which are used to compare automatic systems of gender/age prediction.

#### 3.2.3.1 Algorithms

**Gender Recognition** A number of well-established classification algorithms were applied for gender recognition from face images. Thus, long before a vast arrival of CNNs (the respective gender recognition studies are discussed in Chapter 5), MLPs were applied for gender recognition by Golomb et al. [GLS90] and by Khan et al. [KMM05]. Boosting classifiers (and in particular, Adaboost [FS97]) were widely used between 2000 and 2009 as the models which automatically perform feature selection. So numerous studies [Sun+06; YA07; BR07] employed Adaboost for gender recognition during this period. Statistical approaches, like Linear Discriminative Analysis (LDA) and Bayesian classifiers, were also tried for gender recognition by Bekios-Calfa et al. [BCBB11] and Toews and Arbel [TA09], respectively. However, the most popular classification algorithm for gender recognition is by far SVM. It was used by a large variety of recent works [XSL08; Hu+11; TP13; JC15; CSLNRB16] with different feature representations which have been discussed in Subsection 3.2.2.

**Age Estimation** Age estimation problem is usually approached as a classification problem (with coarse or fine-grained classes) or as a regression one.

Among the studies of the first category, we can highlight the work of Lanitis et al. [LDC04] showing the superiority of MLP over the Nearest Neighbour (NN) classifier for age estimation. Otherwise, Ueki et al. [UHK06] trained 11 Gaussian models corresponding to 11 age categories using Expectation Maximization (EM) algorithm. At test stage, each image is fit to all 11 Gaussians, and the age class which corresponds to the model with the maximum likelihood is selected. Of course, SVM was also tested for age classification on multiple occasions [GZSM07; ZMZ11].

However, it was shown by several research groups [Xia+09; ZY10] that the classification formulation is less advantageous than the regression one for age estimation. Indeed, in general, classification assumes that separate classes are uncorrelated, which is obviously untrue for age estimation problem. In other words, human age evaluates continuously, making the regression formulation a more natural way to address the problem.

Thus, Lanitis et al. [LTC02] compared age estimation with linear, quadratic and cubic regression. The optimal model was selected as a combination of the three approaches. Later, quadratic regression was also successfully used by Guo et al. [Guo+08]. An original idea of modelling age evolution via a Gaussian Process was firstly proposed by Xiao et al. [Xia+09]. This idea was further developed by

Zhang et al. [ZY10], who proposed an algorithm for personalized age estimation via multi-task Gaussian Processes where one task corresponds to one person. Support Vector Regression (SVR) [CV95] is an adaptation of SVM for the case of regression. Age estimation was approached with SVR based on BIF texture features [Guo+09].

There are hybrid approaches combining classification and regression in one model. For example, Han et al. [Han+15] proposed a hierarchical model for age estimation: firstly, they performed a coarse classification to identify an approximative age category for a given face, and then, the fine-grained age estimation was done within the selected category.

Apart from the most used classification and regression formulations, Chang et al. [CCH11] addressed age estimation as an ordinal ranking problem via the algorithm which they called *OHRANK*. Given $K$ different ages in a training dataset, OHRANK solves $(K-1)$ binary classification problems where the goal is to estimate whether a face is older or younger than $k$ years old. Having trained $(K-1)$ binary classifiers $f_k(x)$, at test time, the ranking of a given face is evaluated as $r(x) = \sum_{k=1}^{K-1}[[f_k(x) > 0]]$, where $[[\cdot]]$ is 1 when inner condition is true and 0 otherwise. In practice, the ordinal ranking formulation has appeared to be equally effective for age estimation as the regression one. However, the principal downside of ranking with respect to classification and regression is that it requires $(K-1)$ age estimations instead of a single one.

Finally, Geng et al. [GYZ13] proposed a way to extend the classification formulation of age estimation. The idea consists in using real-valued "soft" class labels instead of binary one-hot [1] vectors during the training. The experimental results reported in [GYZ13] are very promising, because they demonstrate that distributed labels introduce the notion of the age continuity to the pure classification formulation. We present label distribution encoding in more details in Chapter 5, where we use it for training of age estimation CNNs.

#### 3.2.3.2 Metrics

**Gender Recognition** Gender recognition is a binary classification problem, and the accuracy of automatic gender recognition algorithms is most often measured by a *Classification Accuracy (CA)*, which is simply defined as the ratio between the number of correct predictions $N_c$ and the total number of predictions $N$: $CA = \frac{N_c}{N}$.

The biggest downside of CA is that it does not reflect the partial prediction scores for men and women. Thus, sometimes the standard binary classification metrics such as True Positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN) rates are used together with CA. Otherwise, ROC curve is often employed in addition to CA for binary classification problems. A simple way to summarize a ROC curve in one real value is the usage of an *Area Under Curve (AUC)* [HM82] which allows the direct comparison between a pair of ROC curves.

**Age Estimation** As discussed in Paragraph 3.2.3.1, age estimation can be approached both as a classification and as a regression problem.

In the first case, age prediction accuracy is estimated with the already introduced CA. However, in

1. One-hot is a group of bits among which the legal combinations of values are only those with a single high (1) bit and all the others low (0).

the regression scenario, the most used metric is *Mean Absolute Error (MAE)*. MAE is simply defined as a mean value of absolute differences between predicted ages $p$ and real ages $t$ (the averaging is done on $N$ testing examples):

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|p_i - t_i| \tag{3.2.1}$$

MAE is a single real value which allows an easy comparison between different age estimation approaches, and that is why this metric is used in quasi-all research studies on the subject.

### 3.2.4 Practical Interest

The significant amount of research studies (presented in this section) which is devoted to automatic gender recognition and age estimation is explained by the high practical interest of these problems. For example, gender and age prediction systems are widely used in marketing and in the analysis of the customer's demographics. Owners of shops and boutiques are more and more interested in automatically collecting the demographics of the clients which are interested in certain groups of goods. Indeed, the recognition of the clients' gender and age can be used to automatically propose the targeted products. Thus, Adidas is planning to use "digital walls" in their shops located in the USA and UK to automatically profile the clients and to eventually speed up the service [2]. In the same spirit, commercials in the billboards can be potentially adapted depending on the pedestrians who pass by.

Automatic age estimation is also used to prevent children from accessing to services with age restriction (like purchasing of tobacco and alcohol from the vending machines, trying roller coasters in amusement parks, browsing through adult websites, etc.) In the same spirit, visual demographics recognition is essential for humanoid robots [3] which must adapt their interaction depending on the people in front of them.

Moreover, being the primary soft biometrics characteristics, gender and age have a lot of applications in the domain of video surveillance and security. Indeed, intelligent security systems can locate a person of interest suspect based on a specific set of soft biometrics attributes.

In Orange, we are interested in gender recognition and age estimation from face images due to three main reasons. Firstly, as mentioned in the introductory Chapter 1, estimation of facial traits is indispensable for automatic indexing of photo collections which are stored by our clients in the Orange cloud. Secondly, similarly to the use cases presented above, demographics recognition systems can be used in Orange boutiques for collecting anonymized information about the customers and their preferences. Finally, gender recognition and age estimation can potentially be integrated at home TV terminals in order to propose the targeted content with respect to the audience.

## 3.3 Gender/Age Synthesis and Editing in Face Images

The problem of gender/age prediction which has been considered in Section 3.2 and the problem of gender/age synthesis and editing which is discussed in the present section are both focused on the same facial traits, but the latter one is arguably much more difficult. Indeed, unlike the prediction

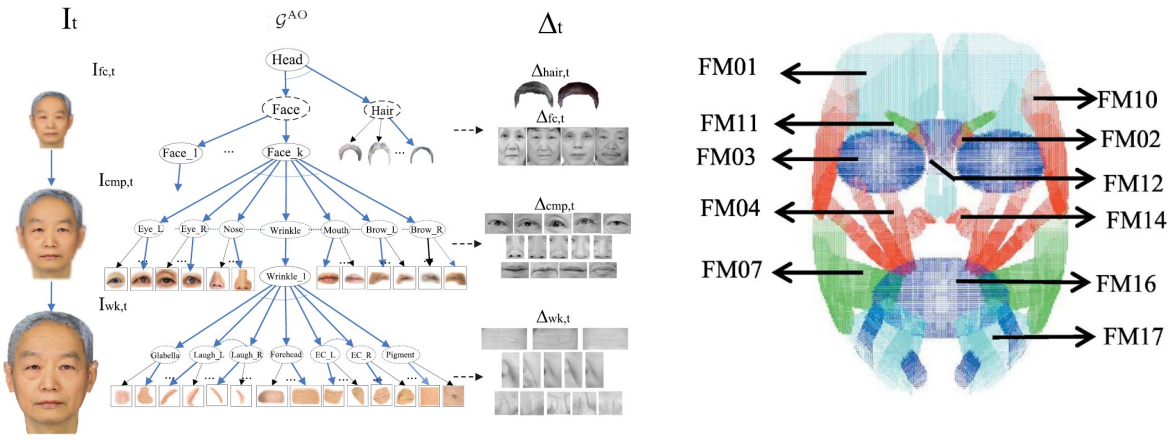
2. http://www.orange-business.com/fr/magazine/tendances/reconnaissance-faciale-explorez-la-tete-de-vos-clients
3. https://www.theguardian.com/travel/2015/aug/14/japan-henn-na-hotel-staffed-by-robots

task, gender/age synthesis and editing requires a comprehensive representation of all face characteristics (including identity traits, facial expression, etc.) in addition to gender and age. Below, we separately present how such a complex modelling was performed in the most significant works on automatic aging/rejuvenation and gender swapping.

### 3.3.1 Face Aging/Rejuvenation

Also known as age progression/regression [Shu+15] and age synthesis [FGH10], automatic face aging/rejuvenation is defined as aesthetically rendering a face image with natural aging and rejuvenating effects on the individual face [FGH10]. Synthetic modelling of an aging process is an extremely complex and ill-posed problem because an appearance of a particular person at a certain age is conditioned on a number of external factors such as living style, health state, genetic heritage, amount of emotional stress, disease processes, exposure to ultraviolet radiation, etc. [FGH10].

In this subsection, we make a brief presentation of the most notable works on the synthetic face aging/rejuvenation which form two distinct families, namely modelling-based and prototype-based. An interested reader can find a more extensive overview of aging/rejuvenation approaches in [FGH10] (for works before 2010) and in [Shu+16] (for more recent works).




(a) (b)

Figure 3.3.1 – (a) (Extracted from [Suo+10]). *And-or graph* for modelling-based aging/rejuvenation: "and" nodes split a human face into multiple coarse-to-fine parts, while "or" nodes propose a selection of templates for each considered face part (*i.e.* eyes, ears, nose etc.) For aging/rejuvenation, and-or graphs corresponding to various age categories are connected in a Markov chain. (b) (Better viewed in color). (Extracted from [Shi+12]). Aging/rejuvenation based on separate modelling of facial muscles.

**Modelling-Based Approaches** Modelling-based aging/rejuvenation methods employ parametric models to simulate the physical aging mechanism of muscles, skin and skull of an individual. Different modelling-based approaches were proposed including AAMs [LTC02], craniofacial growth [RC06a], graphs [Suo+10], statistical-based [Pay10] and 3D models [She+11].

For example, the craniofacial approach of Ramanathan et al. [RC06a] consists of two models: shape transformation one which detects the aging-caused skull deformations in a human face, and a texture transformation one which focuses on the skin changes and wrinkles.

And-or graph [Suo+10] is another typical example of a modelling-based aging/rejuvenation approach. As illustrated in Figure 3.3.1-(a), and-or graph is a hierarchical model, where "and" nodes split a human face into multiple coarse-to-fine parts, while "or" nodes propose a selection of templates for each considered face part (*i.e.* eyes, ears, nose etc.) There are as many and-or graphs as there are different age categories, and aging between them is performed by training a Markov chain. At test time, firstly, the current age category is estimated by selecting the and-or graph which maximizes the posteriori probability of a given face, and then, the aged version of the face is obtained via sampling through the Markov chain.

Modelling-based models often exploit the knowledge of the face physiology. Thus, Shihfeng et al. [Shi+12] simulated face aging via preselected groups of face muscles (*cf.* Figure 3.3.1-(b)). Deformation of each muscle is modelled separately, and the parameters of the corresponding growth functions are estimated during the training.

The common limitation of all modelling-based methods is similar to the one of anthropometry-based approaches for age estimation (which are discussed in Paragraph 3.2.2.1), namely the requirement of the input faces to be perfectly frontal and aligned. Obviously, this requirement cannot be always fulfilled for the photos taken "in the wild". Moreover, many of the modelling-based methods of aging/rejuvenation also require long-term aging sequences of the same person. Unfortunately, it is very costly to collect big enough training datasets with such sequences. Finally, modelling-based methods are also known to be computationally expensive. As a result, the vast majority of the contemporary face aging/rejuvenation methods are prototype-based.

**Prototype-Based Approaches** Prototype-based methods [BP95; TBP01; KSSS14] define average faces calculated on training images of certain age categories as their prototypes. Differences between the prototype faces constitute the aging patterns which are further used to transform an input face image into the target age category. A typical pipeline of face aging/rejuvenation with a prototype-based approach is illustrated in Figure 3.3.2.

The prototype approaches are often fast, but due to the fact that they discard personalized information, these methods are prone to lose the identity while aging. Therefore, contemporary prototype-based methods use explicit mechanisms to preserve the original identity.

One of the most well known and typical prototype-based approaches was developed by Kemelmacher-Shlizerman et al. [KSSS14]. Leveraging a huge Internet-based photo collection, the authors introduced an original method for constructing average image subspaces. The resulting prototypes depict averaged men and women aged between 0 and 80 years old keeping the texture differences between modelled ages. One of the key parts of the algorithm is the simulation of the initial image lightning in the aged/rejuvenated one, which surprisingly improves the human perception of the results. As indicated by the authors, the model of Kemelmacher-Shlizerman et al. [KSSS14] is particularly effective for aging of children.

The two state-of-the-art studies [Shu+15; Wan+16] on face aging/rejuvenation are also prototype-based. Shu et al. [Shu+15] proposed Coupled Dictionary Learning (CDL). The authors learn a dictionary per each age category, and the aging pattern is encoded by the dictionary bases. Pairs of neighbouring dictionaries are learned jointly, and the identity information is regarded as the reconstruction error which

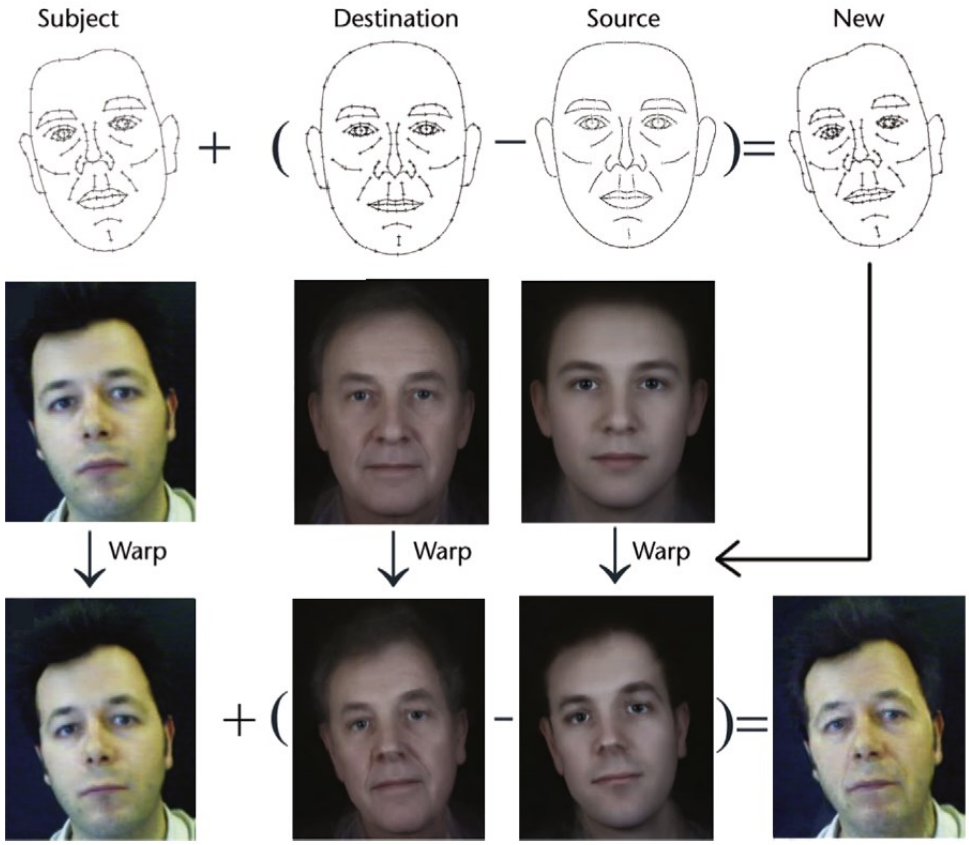



Figure 3.3.2 – (Extracted from [BP95]). A typical pipeline of face aging/rejuvenation with a prototype-based approach. Input face is aligned (warped) with the pre-calculated face prototypes for the target and initial age categories. After that, aging is performed by simply adding the aging pattern (which is just the difference between the two prototypes) to the initial face.

is added into the aged face directly. Despite convincing aging results, CDL suffers from the ghosting artefacts. The other Recurrent Face Aging (RFA) aging/rejuvenation method proposed by Wang et al. [Wan+16] uses Recurrent Neural Networks (RNNs) for age pattern transition between the coefficients of eigenfaces calculated in different age categories. Contrary to CDL, RFA smooths the ghosting artefacts but poorly preserves the original person's identity.

Despite aging and rejuvenation are equally important, once trained, the vast majority of the discussed methods can change ages only in one direction (*i.e.* either to age or to rejuvenate, but not both). This is a serious limitation which can be naturally addressed by employing generative models (*cf.* Chapter 6).

### 3.3.2 Gender Swapping

Swapping gender of a human face is a challenging problem of face editing where the goal is to change the visual perception of gender in a given photo without altering other face attributes. Contrary to face aging/rejuvenation, gender swapping is a problem for which it is difficult to find training examples with the ground truth. Indeed, there are relatively few persons who have physically changed gender during their lives, and, to the best of our knowledge, there is no public dataset for the gender swapping problem. Therefore, this problem received less attention than the problem of synthetic aging/rejuvenation. Nevertheless, some of the universal techniques which are presented above in the context of synthetic

aging/rejuvenation were adapted for the problem of gender swapping as well.

For example, Rowland and Perrett [RP95] used a basic prototype-based algorithm for gender swapping. More precisely, the authors calculated the male and female prototypes, and the difference between the two was added to an input face in order to swap its gender (similarly to the pipeline presented in Figure 3.3.2). In the same spirit, Suo et al. adapted and-or graph presented above (*cf.* Figure 3.3.1-(a)) for gender swapping in [Suo+11].

To the best of our knowledge, the most recent (non-deep learning) approach for gender swapping was proposed by Othman and Ross [OR14]. It consists in morphing two (priorly aligned) faces (a female and a male ones) and selecting an intermediate face in a trade-off between the identity preservation and gender alteration. This morphing approach is simple and fast to implement, but the resulting face images suffer from numerous ghosting artefacts.

### 3.3.3 Practical Interest

The discussed approaches for aging/rejuvenation and gender swapping in faces are not only interesting from the scientific point of view, but also have a number of immediate and potential applications.

There is a huge demand on age synthesis solutions in police and law enforcement services [FGH10]. Indeed, automatic face aging/rejuvenation algorithms may serve as an alternative to forensic artists [4], the humans with professional drawing skills, who create sketches of the wanted suspects, escaped fugitives, or lost persons whose existing photos are outdated.

Synthetic aging and rejuvenation are also required in film production to hinder the real age of actors. Such effects are usually obtained via special visual effects or make-up. A classical example which is often given in this context, is the Hollywood film "The Curious Case of Benjamin Button" [5], where the appearance of the character portrayed by Brad Pitt is convincingly changed throughout all ages.

The ability to stop physical aging or, at least, to change the visual appearance in order to look younger has always been a dream of many people. The contemporary cosmetology proposes plastic surgeries which can physically change the shape and the texture of a human face. However, such surgeries are often associated with a certain risk, and the synthetic rejuvenation techniques offer an opportunity to imagine the surgery results prior to the physical intervention [Koc+96].

Finally, face aging/rejuvenation methods are often used to enhance face recognition software. Indeed, as it is further discussed in Section 6.4 of Chapter 6, despite a tremendous progress of face recognition models in recent years [PVZ15; SKP15], the sensitivity to human age variations remains an Achilles' heel of the majority of the state-of-the-art face verification software. A possibility to synthetically normalize ages in a pair of compared faces can therefore be crucial to improve the quality of cross-age face verification. As a matter of fact, this face verification use case is the most relevant one for Orange, and this is the reason why in Chapter 6, we explicitly show that the face editing method which is proposed in this manuscript can be used for age normalization in the cross-age identity verification scenario.

Gender swapping can also be potentially used in different contexts. For example, one can imagine employing gender and age editing in a pair with a face verification software to estimate the kinship [Lu+15] or siblings [Vie+14] relationships. Not to mention that it is simply fun to discover a male or

4. An example of a website of a professional forensic artist can be consulted here: http://www.forartist.com
5. http://www.imdb.com/title/tt0421715/

female version of yourself. Thus, there are some entertainment online applications which propose to masculinize or to feminize a given face. [6]

## 3.4 Conclusion

The presented chapter has been devoted to the overview of the most notable *non-deep learning* works on two main problems which are addressed in the present manuscript, namely: (1) gender/age prediction (*cf.* Subsection 3.2) and (2) gender/age editing (*cf.* Subsection 3.3) based on images of human faces. As explained above, the analysis of the existing *deep learning* studies is done separately in the beginning of Chapters 5 and 6.

Below, we present the summary of the observations which we believe the most important in this overview chapter:

1. Gender recognition and age estimation are different problems in terms of hand-crafted feature representations which are mostly effective for them (LBP for gender recognition and BIF for age estimation).
2. Features which are learned with different manifold learning algorithms have been proven competitive with the hand-crafted ones for both tasks.
3. Age estimation is more effectively solved as a regression problem than a classification one. However, label distribution encoding [GYZ13] is a promising way to introduce the notion of the age continuity in the classification formulation of age estimation.
4. Modelling approaches for synthetic aging/rejuvenation often require input sequences of faces of the target person taken at different ages, which is hardly realistic for real life applications. At the same time, the prototyping approaches are prone to produce "too averaged" faces which often lose the original person's identity after aging/rejuvenation.
5. Gender swapping problem has attracted less attention of the research community than synthetic aging/rejuvenation due to the lack of the training datasets with the ground truth.

In the rest of this manuscript, we describe our contributions comparing the resulting deep models for gender/age prediction and editing with the state-of-the-art studies presented above.

6. https://corporate.moonjee.com/action/gender.php.

# Part II

# Contributions

Chapter 4

# Preliminary Studies

**Contents**



## 4.1 Introduction

CNN (which has been introduced in Chapter 2) is a fundamental deep model which we extensively use to address the two primary objectives of the present manuscript, namely: gender/age prediction from face images and face synthesis/editing in Chapters 5 and 6, respectively. Therefore, given the primordial importance of CNNs for this PhD, we have started our research with two preliminary studies (detailed in Sections 4.2 and 4.3, respectively) which are aimed at better understanding of advantages and limitations of these deep models.

The first study is devoted to the comparison between hand-crafted and CNN-learned image representations (the respective definitions can be found in Chapter 2). In particular, we are interested in evaluating the capacities of the compared feature representations (1) to adapt to heterogeneous training data, and (2) to generalize to unknown data. Indeed, the first of these two aspects is extremely important in the context of the contemporary training datasets which are often automatically collected from various sources in the Internet, while the second one is simply essential for real-life applications of a machine learning

model. For the experimental comparison, we have selected the problem of gender prediction from images of pedestrians, which we believe to be particularly suitable for the stated goals (our motivations for selecting this problem are provided in Subsection 4.2.1). Despite a limited amount of pedestrian images available for training, the first preliminary study demonstrates the superiority of CNN-learned features over hand-crafted ones, and in addition, it shows the high effectiveness of transfer learning via fine-tuning of an already pretrained CNN.

In our second preliminary study, we approach one of the primary problems of this PhD: gender recognition from face images. In the frame of this study, we explore the importance of the complexity of the CNN architecture (*i.e.* number of trainable layers, and number of weights in each layer) for this particular problem when training is done *from scratch* (*i.e.* without fine-tuning of a pretrained CNN). More precisely, our goal is to investigate the relation between the size of the CNN architecture and the resulting gender classification accuracy. At the first sight, the answer seems obvious. Indeed, as mentioned in Chapters 1 and 2, it has been previously shown on multiple occasions that given enough training images, deep CNNs generally outperform shallow CNNs (in other words, the bigger is the CNN architecture the better are its performances). Nevertheless, despite the abundance of face images with gender annotations for training (about 450K), the results of the second study demonstrate that gender recognition accuracy is saturated with a very shallow CNN. We suspect that the reason for that is a relative simplicity of the problem of gender recognition from face images (for example, with respect to gender recognition from images of pedestrians), the conjecture which is further confirmed in Chapter 5 by comparing gender recognition and age estimation problems.

The conclusions of the two preliminary studies are used in Chapter 5, where we design the state-of-the-art solutions for gender and age prediction from face images.

## 4.2 Study 1: CNN-Learned vs. Hand-Crafted Features

As discussed in Chapter 2, deep CNNs have revolutionized the domain of computer vision, and are currently the standard models for object recognition in images. Nowadays, it is a common practice to reuse the features which are learned by deep CNNs on large datasets (typically, on *ImageNet*) for other problems. These CNN-learned features (which are either used "as is" or fine-tuned for the target domain) in the majority of cases, outperform the hand-crafted features even if the latter are specifically designed for the particular problem [SR+14].

However, what is the principal advantage of learned features with respect to hand-crafted ones? In an attempt to answer to this question, in this section, we compare these two classes of image representations in the context of gender recognition from pedestrian photos. The results of our experiments suggest that one of the key difference of the learned features with respect to the hand-crafted ones is the ability of the former to better absorb the heterogeneity of the training data which results in better generalization of the learned features on unseen datasets. This flexibility to variations in training images is achieved during the training process (which does not exist for hand-crafted features), and as shown in this section, even a relatively small number of training images and a very compact CNN architecture are enough to learn features with the stated above properties.

### 4.2.1 Gender Recognition from Images of Pedestrians

#### 4.2.1.1 Motivation

Despite this PhD is focused on the analysis of face images with deep learning, we have selected the problem of gender recognition from *pedestrian images* (and not from face images) for the comparison between CNN-learned and hand-crafted features in the present section. There are three main reasons motivating our decision which are presented hereafter.

– As stated above, one of the objectives of this section is the evaluation of how the two compared classes of feature representations adapt to heterogeneous training data. Given all possible camera positions, body orientations, weather conditions, etc., there are obviously much more variations in images of pedestrians than in images of faces.

– The second reason is in part the consequence of the first one. Indeed, gender recognition from pedestrian images is generally more difficult than gender recognition from faces (thus, the CA for the former are in the vicinity of 80%, while for the latter, the state-of-the-art CA are above 95%). At the same time, comparison of features on a more challenging problem is more representative, as in this case, the quality of image descriptors can really make the difference.

– Finally, the last but not the least reason is the fact that there are relatively few pedestrian images at public access. This gives us a possibility to evaluate the CNN-learned features in the context of limited training data, as well as to test the effectiveness of transfer learning.

Finally, on its own, the problem of pedestrian gender recognition is also very important from the applicative point of view. We elaborate more on that in the following paragraph.

#### 4.2.1.2 Problem Description

The ability to automatically profile pedestrians based on their gender is a very important issue which has immediate applications in video surveillance and security. However, in these particular domains, obtaining a clear shot of a person's face might be infeasible due to technical, environmental and privacy reasons. Indeed, in the context of video surveillance by CCTV cameras, the Subjects of Interest (SoI) do not cooperatively look head on into the camera. Therefore, it is likely that SoI is seen from the back or her (his) face or the face is occluded. Moreover, the video surveillance scenario implies that SoI takes a little part of the camera's field of view. Hence, the pixel resolution of the face region is often prohibitively low for robust face analysis. Thus, it is important to also be able to estimate a gender of a pedestrian having only a general image of her (his) body.

There are many clues which add humans to instantly estimate a gender of a person without seeing her (his) face. For example, a silhouette of the body is a strong clue. However, the silhouette details are often hidden behind the clothes which in its turn can be useful to guess a person's gender. Otherwise, the hairstyle is one of the most obvious clues though it can also be a source of confusion. Despite the difficulty of the problem, the study presented below in Subsection 4.2.4 shows that in about 90% of cases, human observers correctly estimate a person's gender given a single image of her (his) body.

Here and below, by *pedestrian* (or *body*) image, we understand a static color image containing a full body of a single human (including head, torso, arms and legs).

#### 4.2.1.3 Related Work

Being focused on face-centred studies, we have not covered the existing works on gender recognition from body images in Chapter 3. Therefore, an overview of the state-of-the-art for the considered problem is performed below.

Cao et al. [Cao+08] were the first to address the problem of automatic gender recognition from static body images. Authors preprocessed input images by centring and normalizing the height of bodies and subsequently splitting them into overlapping patches which correspond to specific body parts. HOG features were further extracted from these patches and concatenated to form a single feature vector which was provided at the input of a boosting classifier.

In the same spirit, Collins et al. [Col+09] also employed HOG-like features for body-based gender recognition. They proposed their own feature descriptor called PixelHOG which is based on very dense HOG features computed on a custom edge map. Collins et al. combined these features with color information extracted from hue and saturation values of images pixels.

Unlike previous studies, Guo et al. [GMF10] employed manifold learning on Gabor-like BIF features and the SVM classifier for gender recognition. Only C1 BIF features were used as the authors reported that C2 BIF features degraded the performance (as in the case of gender recognition from face images).

A new feature representation named "poselet" was proposed Bourdev et al. [BMM11]. Basically, a poselet combines HOG features, color information extracted at the pixel level and skin features.

The first CNN for the considered problem was trained by Ng et al. [NTG13]. In particular, the authors used a tiny CNN of about 60K trainable weights which were organised in 2 convolutional and 2 fully-connected layers.

Finally, body-based gender recognition model was a part of a multi-distance gender recognition system proposed by Gonzalez et al. [GS+16]. As in the majority of the related works, their model employed HOG hand-crafted features to describe body images. More precisely, body images were split into patches to extract HOG features while the gender classification was done by linear SVM.

| Reference | Dataset | Features | Classifier | Evaluation Protocol | CA (%) |
|---|---|---|---|---|---|
| [Cao+08] | *MIT* (frontal, back) | HOG | Adaboost variant | 5-CV | 75.0 |
| [Col+09] | MIT (frontal) | PixelHOG, color | SVM | 5-CV | 76.0 |
| | *VIPeR* (frontal) | | | | 80.6 |
| [GMF10] | *MIT* (frontal, back) | BIF | SVM | 5-CV | 80.6 |
| [BMM11] | Private (unconstrained) | HOG, color, skin | SVM | single test | 82.4 |
| [NTG13] | *MIT* (frontal, back) | Learned (CNN) | CNN | single test | 80.4 |
| [GS+16] | *Tunnel* (frontal) | HOG | SVM | 5-CV | 87.9 |

Table 4.2.1 – Related work on gender recognition from pedestrian (body) images. Classification Accuracies (CAs) are provided for indicative purposes, but they cannot be directly compared between each other due to the differences in evaluation datasets and protocols. CV = Cross Validation.

The key aspects of all studies mentioned above are summarized in Table 4.2.1. As one may notice from Table 4.2.1, there is no common protocol for evaluating gender recognition models on body images. The experimental parts of the existing studies vary in terms of the used datasets (*MIT* [Ore+97], *VIPeR*

[GBT07], *Tunnel* [See+08] and different private datasets), allowed human body poses (only frontal, frontal and back, unconstrained), and even evaluation protocols (single test, CV (Cross Validation)). Therefore, the mentioned approaches (and, in particular, feature representations) cannot be compared between each other based on the scores in Table 4.2.1. Moreover, the previous studies did not evaluate their respective models in the cross-dataset scenario which is essential to measure the practical utility of the designed models in real-life conditions.

On the contrary, the experiments of this section are performed on a collection of heterogeneous datasets (presented in Paragraph 4.2.3.1) both in the same-dataset and cross-dataset scenarios, which allows a complete and fair comparison of learned and hand-crafted features.

### 4.2.2 Compared Feature Representations

In this section, we compare the effectiveness of six image features, three hand-crafted ones (*cf.* Paragraph 4.2.2.1) and three CNN-learned ones (*cf.* Paragraph 4.2.2.2), to represent the pedestrian images in the context of gender recognition task. Below, we present the selected feature representations and motivate our choices. For each type of features, we detail the used configuration of hyperparameters and the resulting size of the features vector (the size of input pedestrian images is standardized before feature extraction as further detailed in Paragraph 4.2.3.2).

#### 4.2.2.1 Hand-Crafted Features

A number of various hand-crafted feature representations have been designed for different problems of computer vision. Often such features are well-adapted for certain problems, and are much less effective for another ones. For our experiments, we have selected three particular types of feature representations, namely: person Re-Identification, LBP and HOG, which are mostly suitable for gender recognition from pedestrian images according to the previous studies.

**Person Re-Identification Features** Introduced in [Lay+12], Re-Identification (RI) features of an image contain low-level color and texture information. They were initially proposed for person recognition (identification) in the crowd [Lay+12], hence the name. We have selected RI features for comparison as they were used for pedestrian gender recognition in the original work describing the *PETA* collection of images [Den+14] (the dataset which we use in our experiments, *cf.* Paragraph 4.2.3.1 for more details).

The size of the RI features vector is independent of the size of an input image. In particular, the complete vector is composed of six 464-dimensional vectors each of which is extracted from 6 equally-sized horizontal strips from a body image. These strips are described using 8 color channels (RGB, HSV and YCbCr) and 21 texture filters (Gabor, Schmid) derived from the luminance channel. Histograms with a bin size of 16 are used to describe each channel. Therefore, the resulting features vector is of size $2784 = (8+21) \times 16 \times 6$.

**LBP Features** Local Binary Pattern (LBP) features [OPH96] are arguably the most popular hand-crafted features for texture description. Selection of LBP features for our experiments is natural given the fact that, as mentioned in Chapter 3, they were successfully employed for gender recognition from face images.

In our implementation of the LBP encoding, square cells of size 5x5 are used to compare the central pixel with its 8 neighbours which are located at a distance of 2 pixels. The rotation invariance of the descriptors is ensured by counting only the uniform patterns in histograms as recommended in [Bar+13] (therefore, the resulting histograms are composed of 10 bins). All local histograms are normalized before concatenation. LPB features have been extracted from body images of size 150x50 which results in 3000-dimensional feature vectors.

**HOG Features** As one may notice from Table 4.2.1, Histogram of Oriented Gradients (HOG) features were the most used ones in relevant works on gender recognition from pedestrian images. This is not surprising given the fact that HOG features were initially introduced for detection of pedestrians from static photos [DT05]. This explains the choice of HOG for the comparison in this section.

In particular, we use 8x8 square cells which are organized in 2-by-2 blocks to extract HOG features. The number of histogram bins is set to 9. Thus, given a body image of size 150x50, the resulting features vector contains 2448 values.

#### 4.2.2.2 CNN-Learned Features

As mentioned in Chapter 2, trained CNNs can be used as feature extractors. In particular, in order to extract CNN-learned features, a given image must be processed by a CNN. After that, a features vector is normally composed of the activations of one of the CNN layers (convolutional or fully-connected). Usually, the deepest layers (*i.e.* those which are closer to the network output) are selected for feature extraction, because the respective image representations are more problem-dependent (*cf.* Chapter 2 for the details).

In this section, we have selected two CNN architectures for feature learning, namely: *pedestrian_CNN* and *AlexNet* [KSH12] (the two are presented in Table 4.2.2).

The role of the *pedestrian_CNN* in our comparison is to verify whether it is possibly to learn decent feature representations with a small amount of training images. Therefore, the CNN has been designed to be trained from scratch for gender recognition on available pedestrian images. Due to the very limited size of the training dataset (in total, 10K images, but for certain experiments only about 1K images are used for training (*cf.* Subsection 4.2.3)) the architecture of *pedestrian_CNN* is very compact in order to avoid overfitting. In particular, the design of *pedestrian_CNN* is largely inspired by the one of *LeNet-5* (*cf.* Chapter 2).

Alternatively, the role of *AlexNet* is to evaluate the effectiveness of transfer learning for the selected problem. Therefore, we have used the CNN which was initially pretrained on the *ImageNet* dataset. As a side remark, it is important to mention that any other deep CNN pretrained on *ImageNet* (such as *VGG-16/19*, *ResNet* etc.), could have also been employed for the stated purpose, but, for simplicity, we have selected the smallest of these architectures (*i.e. AlexNet*). As explained above, there are two possibilities of transfer learning from a pretrained CNN: (1) using the network "as is", or (2) priorly fine-tuning it on the available training images. In this section, we explore both these options by using the original *AlexNet* and the one fine-tuned on the pedestrian images. The latter network is further referred as *AlexNet-FT*.

Below, we provide some details on training (fine-tuning) of the presented CNNs.

| *pedestrian_CNN* | *AlexNet* |
|---|---|
| Input: 150x50 | Input: 227x227 |
| Conv: 100@5x5 | Conv: 96@11x11 |
| MaxPool: 2x2 | |
| Conv: 100@5x5 | Conv: 256@5x5 |
| MaxPool: 2x2 | |
| — | Conv: 384@3x3 |
| — | Conv: 384@3x3 |
| — | Conv: 256@3x3 |
| — | MaxPool: 2x2 |
| — | FC: 4096 |
| **FC: 100** | **FC: 4096** |
| Softmax: 2 | |

Table 4.2.2 – CNN architectures used to extract the learned features. The layers from which the features are extracted are highlighted in bold. "Conv: N@MxM" denotes a convolutional layer with N kernels of size MxM. "MaxPool: MxM" denotes downsampling by a factor of M using Max-Pooling. "FC: N" denotes a fully-connected layer with N neurons.

**Features Learned by *pedestrian_CNN*** As detailed in Table 4.2.2, *pedestrian_CNN* is composed of two convolutional layers, a fully-connected layer of 100 neurons and the final layer of 2 neurons which is used for 2-class gender classification with the Softmax activation function. 0.5 dropout is employed after the second convolutional layer. Contrary to *LeNet5*, *pedestrian_CNN* is trained with ReLU activations instead of Sigmoid activations, as empirically, it has appeared to significantly improve the convergence. The *pedestrian_CNN* features are extracted from the activations of the fully-connected layer, so the resulting feature vectors are composed of 100 values (which is more than 100 times smaller than all hand-crafted feature representations presented above).

**Features Learned by *AlexNet* and *AlexNet-FT*** Both for *AlexNet* and *AlexNet-FT*, the activations of the last fully-connected layer of 4096 neurons are used as learned features. In order to fine-tune *AlexNet-FT*, we substitute the original final layer of 1000 neurons with the same layer of 2 neurons and the Softmax activation as in the case of *pedestrian_CNN*. During the fine-tuning, we have been using the small learning rate ($10^{-4}$) for the convolutional layers to preserve the original features learned on *ImageNet* and bigger learning rate ($10^{-2}$) for the fully-connected layers to better adapt to the target problem (*i.e.* gender classification from pedestrian images).

### 4.2.3 Experiments

#### 4.2.3.1 *PETA* Collection of Datasets

We compare learned and hand-crafted features on the *PETA* (PEdesTrian Attribute) collection of datasets [Den+14]. To the best of our knowledge, *PETA* is the largest open-access collection of pedestrian images with gender annotations. Originally *PETA* was composed of 10 datasets of different sizes: *CUHK*, *PRID*, *GRID*, *MIT*, *VIPeR*, *3DPeS*, *CAVIAR*, *i-LIDS*, *SARC3D* and *TownCentre* (following the same naming as in [Den+14]), with a total of about 19,000 images. Two of these datasets (*MIT* and *VIPeR*) were used in many previous works on gender recognition from full body images as reported in

Subsection 4.2.1. Appearances of body images hugely vary between different datasets of *PETA* in terms of image resolutions (from 39x17 to 365x169), camera angles (pictures are taken either by ground-based cameras or by surveillance cameras which are set at a certain height) and environments (indoors or outdoors). This variety allows the experimental comparison of gender recognition approaches in different conditions and scenarios (*cf.* Paragraph 4.2.3.2). Examples of *PETA* images from each of 10 datasets are presented in Figure 4.2.1.

(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Figure 4.2.1 – Examples of pedestrian images from the *PETA* collection: (a) *CUHK*; (b) *PRID*; (c) *GRID*; (d) *MIT*; (e) *VIPeR*; (f) *3DPeS*; (g) *CAVIAR*; (h) *i-LIDS*; (i) *SARC3D*; (j) *TownCentre*. Original image proportions are preserved.

Authors of *PETA* performed some preliminary gender recognition experiments on the whole collection of 19,000 images [Den+14] using RI features (which have been presented above). They *randomly* split the total collection into 9,500 images for training, 1,900 for validation and 7,600 for testing. The reported CAs vary between 79.7% and 81.4% depending on the used classifier. We have successfully reproduced these results using several random splits of 19,000 images in the same proportions as it was done by authors of *PETA* [1]. However, the *PETA* collection contains many images of the same persons which are taken few seconds away from each other by surveillance cameras. Images like that are almost identical and can considerably bias the resulting prediction rates (if they are split between training and test). After manually removing all quasi-identical images from *PETA* the CAs of the approach proposed in [Den+14] drop down to 63-65%.

This drastic drop in performances proves the importance of the manual filtering of *PETA* collection. In addition to quasi-identical images, *PETA* contains a few images of a very low resolution (height is less than 120 pixels or width is less than 40 pixels) and images where SoI is occluded by side objects (for example, by strollers or by other persons). Such images have also been manually removed from *PETA*. As a result, 8,365 images have been left which is less than half as many as the initial size of *PETA*. Further in this section, the "filtered" version of the *PETA* collection is referred as *PETA_cleaned*. Table 4.2.3 provides the details on the number of images before and after the manual filtering in each of the 10 datasets.

| Collection | *CUHK* | *PRID* | *GRID* | *MIT* | *VIPeR* | *3DPeS* | *CAVIAR* | *i-LIDS* | *SARC3D* | *TC* |
|---|---|---|---|---|---|---|---|---|---|---|
| *PETA* | 4563 | 1134 | 1275 | 888 | 1264 | 1012 | 1220 | 477 | 200 | 6967 |
| *PETA_cleaned* | 3809 | 1043 | 1028 | 876 | 1258 | 100 | 68 | 100 | 41 | 42 |

Table 4.2.3 – Number of images in the *PETA* and *PETA_cleaned* collections.

| Dataset | Train size (♂+ ♀) | Test size (♂+ ♀) |
|---|---|---|
| *CUHK* | 3432 = (2420 + 1012) | 377 = (189 + 188) |
| *PRID* | 942 = (449 + 493) | 101 = (50 + 51) |
| *GRID* | 928 = (531 + 397) | 100 = (50 + 50) |
| *MIT* | 792 = (532 + 260) | 84 = (42 + 42) |
| *VIPeR* | 1138 = (556 + 582) | 120 = (60 + 60) |
| *3DPeS* | 0 | 100 = (50 + 50) |
| *CAVIAR* | 0 | 68 = (34 + 34) |
| *i-LIDS* | 0 | 100 = (50 + 50) |
| *SARC3D* | 0 | 41 = (21 + 20) |
| *TC* | 0 | 42 = (21 + 21) |

Table 4.2.4 – Datasets of the *PETA_cleaned* collection: training and test parts per dataset.

#### 4.2.3.2 Experimental Protocol

In order to fairly compare the presented hand-crafted and learned features taking into the account different evaluation scenarios, we perform three separate experiments of increasing complexity.

**Experiment 1** provides the easiest conditions for gender recognition models: they are separately trained and tested on the five biggest datasets of the *PETA_cleaned* collection: *CUHK*, *PRID*, *GRID*, *MIT* and *VIPeR*. In many aspects, body images from the same dataset are closer to each other than body images from different datasets: for example, they are usually taken by the same camera, preprocessed in the same way etc. We further refer to training images coming from a single dataset as *homogeneous* training data. Therefore, Experiment 1 can be summarized as "homogeneous training data; same-dataset evaluation scenario".

**Experiment 2** is one step more difficult than the first one. The features are compared on the same five biggest datasets of *PETA_cleaned*, but in this case, for each feature representation, a single gender recognition model is trained on a mixture of the datasets. Then the obtained model is evaluated on the five testing parts of the respective datasets as in Experiment 1. This way, we test the ability of the compared features to adapt to *heterogeneous* training data from different sources (datasets). In other words, Experiment 2 can be summarized as "heterogeneous training data; same-dataset evaluation scenario".

**Experiment 3** represents the most adverse conditions for gender recognition systems. As in Experiment 2, the gender recognition models are trained on a mixture of five biggest datasets (*i.e.* on heterogeneous data), but contrary to the first two experiments, they are tested on five completely unseen datasets: *3DPeS*, *CAVIAR*, *i-LIDS*, *SARC3D* and *TownCentre* ensuring the cross-dataset evaluation scenario. As a result, Experiment 3 can be characterized as "heterogeneous training data; cross-dataset evaluation scenario" and is the closest experiment to the real-life conditions.

The details of split of all datasets of *PETA_cleaned* collection into training and test parts are provided

1. We thank the authors of the *PETA* dataset [Den+14] for providing the source codes which allowed reproduction of their scores.

in Table 4.2.4. To ensure an objective evaluation, all test datasets are balanced between genders. Moreover, we compare not only resulting gender CAs, but also AUCs as the latter is invariant to the choice of the decision threshold of the classifier.

The goal of the presented experiments is the features comparison and not designing the best gender classifier for pedestrian images. Therefore, for all compared feature representations, we employ the same SVM classifier with a linear kernel (in particular, we use the SVM implementation by Joachims [Joa98]). In case of learned features, the CNNs are firstly trained in a classical manner using back-propagation, and then at test phase, the final classification layer of 2 neurons is cut off and the learned networks are used just as feature extractors.

As presented in Paragraph 4.2.3.1, the original images of the *PETA_cleaned* collection are of different sizes. Before feature extraction, we standardize the image dimensions, making them of 150x50 pixels for all hand-crafted features and for *pedestrian_CNN*, and of 227x227 pixels for *AlexNet* (the latter is imposed by the respective CNN architecture). When rescaling to the new dimensions, it is essential to preserve the original proportions of body images (as they can help in gender recognition). Thus, we firstly rescale an input image so that the resulting image height is 150 pixels while the resulting image width is adapted proportionally. Then, depending on the obtained width, we either symmetrically crop it or symmetrically add empty "black" pixels on both sides of the image to fit the target width of 50 pixels (*cf.* Figures 4.2.2 -(a) and 4.2.2-(b), respectively).


Figure 4.2.2 – Examples of rescaling images from the *PETA_cleaned* collection. Firstly, original images are rescaled so that the resulting image height is 150 pixels. Then the image widths are adapted proportionally by either (a) symmetric cropping; or (b) symmetric adding of "black" pixels.

#### 4.2.3.3 Experiment 1: Homogeneous Training Data; Same-dataset Evaluation Scenario

The results of Experiment 1 are summarized in Table 4.2.5. For convenience, we present CAs and AUCs on the five test datasets and the corresponding average values as well.

There is a clear difference in performances of the SVM models using three different types of hand-crafted features. We observe, that RI features with the average CA of 59.3% are hardly applicable for the problem of gender recognition from body images. Presumably, successful results obtained using

| Features | | Evaluation Metric | *CUHK* | *PRID* | *GRID* | *MIT* | *VIPeR* | Average |
|---|---|---|---|---|---|---|---|---|
| Hand-crafted | RI | CA (%) | 69.5 | 47.5 | 58.0 | 57.1 | 64.2 | 59.3 |
| | | AUC | 0.7714 | 0.5059 | 0.6340 | 0.5612 | 0.6778 | 0.6301 |
| | LBP | CA (%) | 68.4 | 59.4 | 63.0 | 66.0 | 66.7 | 64.7 |
| | | AUC | 0.7595 | 0.6663 | 0.7412 | 0.5890 | 0.7261 | 0.6964 |
| | HOG | CA (%) | 84.1 | 77.2 | 83.0 | **81.0** | 73.3 | **79.7** |
| | | AUC | 0.9199 | 0.8588 | 0.9088 | **0.9002** | 0.8097 | **0.8795** |
| CNN-learned | *pedestrian_CNN* | CA (%) | 82.2 | 75.3 | 79.0 | 73.8 | **75.8** | **77.2** |
| | | AUC | 0.9105 | 0.8424 | 0.8736 | 0.8458 | 0.8111 | **0.8567** |
| | *AlexNet* | CA (%) | 78.3 | 58.4 | 78.0 | 69.1 | 70.0 | 70.8 |
| | | AUC | 0.8924 | 0.7220 | 0.8984 | 0.7988 | 0.7814 | 0.8186 |
| | *AlexNet-FT* | CA (%) | **89.1** | **80.2** | **88.0** | 78.6 | 75.0 | **82.2** |
| | | AUC | **0.9563** | **0.8663** | **0.9660** | 0.8577 | **0.8372** | **0.8967** |

Table 4.2.5 – Learned vs. hand-crafted features. Experiment 1: homogeneous training data; same-dataset evaluation scenario.

RI features in [Den+14] are due to a significant number of quasi-identical images between training and testing datasets (as it is explained in Paragraph 4.2.3.1). While the CA scores of LBP features are superior to that of RI ones, they remain quite moderate. On the contrary, HOG features demonstrate by large the best CA and AUC performances among the hand-crafted features. These results are quite expected given the fact that the majority of previous related works employed HOG features for the considered problem (*cf.* Subsection 4.2.1).

We believe that the advantage of HOG features with respect to other hand-crafted alternatives is that HOG is particularly suited for detecting the shape of an object (in our case, the body silhouette), while, for example, LBP is more sensible to texture differences. At the same time, in pedestrian gender recognition, a body silhouette is a more reliable characteristic than particular texture details.

All three learned feature representations demonstrate decent gender recognition scores comparable or better than the ones obtained by HOG features. The fine-tuned *AlexNet-FT* significantly outperforms the original *AlexNet* highlighting the importance of fitting to the target domain in transfer learning. The 100-dimensional feature vector of a very compact *pedestrian_CNN* obtains the CA score of 77.2% which is almost on par with the score of HOG features and significantly better than the one of original *AlexNet* features. This result proves that good enough features can be learned even with a small CNN on a tiny dataset.

Experiment 1 is the simplest one among the three performed in the present section. The observed results show that when evaluation is done within controlled conditions (*i.e.* the same source of images for training and test, and homogeneous training data), the performances of the best CNN-learned and hand-crafted features are globally comparable.

#### 4.2.3.4 Experiment 2: Heterogeneous Training Data; Same-dataset Evaluation Scenario

Experiment 2 evaluates the ability of the compared features to adapt to the heterogeneous training data. The obtained results, which are presented in Table 4.2.6, significantly vary between hand-crafted and learned features.

Thus, the CA (respectively, AUC) score of HOG features in Experiment 2 is about 12 (respectively,

| Features | | Evaluation Metric | *CUHK* | *PRID* | *GRID* | *MIT* | *VIPeR* | Average |
|---|---|---|---|---|---|---|---|---|
| Hand-crafted | RI | CA (%) | 65.5 | 49.5 | 51.0 | 58.3 | 60.0 | 56.9 |
| | | AUC | 0.7730 | 0.6624 | 0.6863 | 0.5663 | 0.6669 | 0.6690 |
| | LBP | CA (%) | 69.0 | 61.4 | 55.0 | 60.7 | 68.3 | 62.9 |
| | | AUC | 0.7724 | 0.6604 | 0.5888 | 0.6508 | 0.7536 | 0.6852 |
| | HOG | CA (%) | 79.8 | 64.4 | 66.0 | 66.7 | 63.3 | 68.0 |
| | | AUC | 0.9169 | 0.7824 | 0.6860 | 0.6477 | 0.7731 | 0.7612 |
| CNN-learned | *pedestrian_CNN* | CA (%) | 87.3 | 77.2 | 68.0 | 77.4 | 70.0 | **76.0** |
| | | AUC | 0.9485 | 0.8490 | 0.7371 | 0.8861 | 0.7961 | **0.8434** |
| | *AlexNet* | CA (%) | 81.7 | 55.4 | 72.0 | 67.9 | 71.7 | 69.7 |
| | | AUC | 0.8897 | 0.5957 | 0.8284 | 0.8458 | 0.7992 | 0.7918 |
| | *AlexNet-FT* | CA (%) | **90.5** | **79.2** | **84.0** | **81.0** | **78.3** | **82.6** |
| | | AUC | **0.9567** | **0.8502** | **0.9064** | **0.8736** | **0.8481** | **0.8870** |

Table 4.2.6 – Learned vs. hand-crafted features. Experiment 2: heterogeneous training data; same-dataset evaluation scenario.

0.12) points lower than in Experiment 1 (not to mention RI and LBP which demonstrate poor results even in the easiest Experiment 1). This drastic drop of performances illustrate that hand-crafted features suffer from heterogeneous training data.

On the contrary, all three learned features show a good capacity to absorb the variety of training data by maintaining almost the same level of performances as in Experiment 1.

In our opinion, the results of the performed experiment are of the uttermost importance, because they illustrate the fundamental difference between the two compared classes of feature representations. Indeed, unlike hand-crafted features, the CNN-learned ones have little sensibility to the heterogeneity of the training images, because they have the possibility to *adapt* to the particular data during the training process. We believe that this is one of the main reasons of why learned features better scale up to problems with large datasets and many classes, such as *ImageNet* classification.

#### 4.2.3.5 Experiment 3: Heterogeneous Training Data; Cross-dataset Evaluation Scenario

| Features | | Evaluation Metric | *3DPeS* | *CAVIAR* | *i-LIDS* | *SARC3D* | *TC* | Average |
|---|---|---|---|---|---|---|---|---|
| Hand-crafted | RI | CA (%) | 59.0 | 57.4 | 67.0 | 51.2 | 66.7 | 60.3 |
| | | AUC | 0.6332 | 0.8750 | 0.7197 | N/A | 0.6984 | N/A |
| | LBP | CA (%) | 53.0 | 70.6 | 65.0 | 43.9 | 50.0 | 56.5 |
| | | AUC | 0.6040 | 0.7093 | 0.7168 | 0.5238 | 0.4996 | 0.6107 |
| | HOG | CA (%) | 47.0 | 61.8 | 57.0 | 70.7 | 57.1 | 58.7 |
| | | AUC | 0.5553 | 0.5133 | 0.7088 | 0.7571 | 0.6485 | 0.6366 |
| CNN-learned | *pedestrian_CNN* | CA (%) | 78.0 | 80.9 | 70.0 | **82.9** | 61.9 | **74.7** |
| | | AUC | 0.7900 | 0.8841 | 0.7424 | 0.9143 | 0.7392 | **0.8140** |
| | *AlexNet* | CA (%) | 70.0 | 60.3 | 64.0 | 82.9 | 69.0 | 69.2 |
| | | AUC | 0.7684 | 0.6445 | 0.6916 | 0.9095 | 0.7438 | 0.7516 |
| | *AlexNet-FT* | CA (%) | **81.0** | **82.4** | **75.0** | 80.5 | **78.6** | **79.5** |
| | | AUC | **0.8984** | **0.8997** | **0.7384** | **0.9286** | **0.8798** | **0.8690** |

Table 4.2.7 – Learned vs. hand-crafted features. Experiment 3: heterogeneous training data; cross-dataset evaluation scenario.

The results of the third experiment confirm the above conclusions. In particular, unable to adapt to variations in training datasets, hand-crafted features naturally fail to generalize to new datasets.

Indeed, the difference between hand-crafted and learned features becomes even more significant in Table 4.2.7 summarizing the results of Experiment 3. More precisely, when tested on completely unseen datasets, HOG features lose about 21 (respectively, 0.25) points of CA (respectively, AUC) comparing to Experiment 1, and drop to the same level of performances as RI and LBP.

At the same time, even if the performances of the CNN-learned features in Experiment 3 are also lower than in Experiment 1, the corresponding gaps are much smaller: about 3 (respectively, 0.04) points of CA (respectively, AUC) which is completely expectable given more severe conditions of Experiment 3. Summarizing, all compared learned features generalize well on unseen data which is not the case for hand-crafted features.

### 4.2.4 Pedestrian Gender Recognition in Presence of Privacy Protection Filters

*The study presented in this subsection is a side research project which has been derived from the main topic of this section (comparison between CNN-learned and hand-crafted features). It has been performed in the frame of a collaboration with Natacha Ruchaud (Eurecom), Pavel Korshunov (EPFL) and Touradj Ebrahimi (EPFL).*

In particular, the goal of this study is to evaluate the sensitivity of our best pedestrian gender recognition model to altering of test images with privacy protection filters (which are often used in practical applications of video surveillance). More precisely, we compare the gender CAs of our model with that of human estimators (with and without privacy protection) via a crowdsourcing campaign.

Based on the results of Experiments 1-3 presented above, we have selected *AlexNet-FT* CNN as our best pedestrian gender recognition model. Contrary to experiments presented above, in this subsection, *AlexNet-FT* is used both for feature extraction and for gender classification which results in better recognition performances.

#### 4.2.4.1 Privacy Protection Filters

By *Privacy Protection Filters* (PPF), we understand an ensemble of techniques which allow to hide sensitive information in images (for example, PPFs are used to hide human faces and car plates in Google Street View [2]). A number of PPFs has been proposed in literature. Below, we present the most popular among them which are further used in our experiments: Masking, Morphing, Pixelization, Gaussian Blur and k-Means.

*Masking* PPF is a simple approach which consists in mixing an original image $x$ with a completely black image $x_{bk}$. The opacity constant $\alpha$ controls the level of privacy protection (the bigger is $\alpha$, the more obscure is the resulting protected image $x_p$): $x_p = (1-\alpha) \times x + \alpha \times x_{bk}$.

*Morphing* PPF [KE13] is an extension of the Masking PPF, where an original image $x$ is combined with a target one $x_t$ (which in our case can be an average body image, for example). The method firstly divides both images into Delaunay triangles and mixes the vertices of $x$ and the vertices of $x_t$ (in the same way as in Masking PPF). Then the pixel intensities are interpolated according to the closest vertices.

2. https://www.google.com/intl/en/streetview/

(a) (b) (c) (d) (e) (f)

Figure 4.2.3 – Privacy Protection Filters (PPFs) used in our experiments. (a) original body image; (b) body image protected by Masking PPF; (c) body image protected by Morphing PPF; (d) body image protected by Pixelization PPF; (e) body image protected by Gaussian Blur PPF; (f) body image protected by k-Means PPF.

A simple image downsampling can also serve as a PPF which is known as *Pixelization*. In this case, the strength of PPF is controlled by a downsampling ratio.

*Gaussian Blur* PPF consists in filtering an input image with a Gaussian kernel $G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$, where the standard deviation $\sigma$ controls the PPF's strength.

Finally, as indicated by its name, *k-Means* PPF consists in clustering input image pixels in $k$ categories using the k-Means algorithm. Clustering is based on pixel intensities and usually applied separately on three image channels (RGB). The lower is $k$ the stronger is PPF.

An example of applying the presented PPFs on a body image is presented in Figure 4.2.3. In Table 4.2.8, we present the listed PPFs as well as the respective hyper-parameters which have been used in our experiments.

| PPF | Strength Hyper-parameter |
|---|---|
| Masking | $\alpha$: 0.4, 0.7, 0.9 |
| Morphing | $\alpha$: 0.4, 0.7, 0.9 |
| Pixelization | Downsampling Ratio: 3, 5, 7 |
| Gaussian Blur | $\sigma$: 2, 4, 6 |
| k-Means | $k$: 6, 4, 2 |

Table 4.2.8 – Privacy Protection Filters (PPFs) used in the experiments and the respective hyper-parameters which control the strength of the PPFs.

#### 4.2.4.2 Evaluation of Human Gender Classification Accuracy by Crowdsourcing

In order to compare the gender recognition performances of *AlexNet-FT* with that of humans in the presence of the listed PPFs, we have organised a crowdsourcing campaign.

We have randomly selected 300 images from 10 test datasets of *PETA_cleaned* for the evaluation in the crowdsourcing experiment. The usage of all test images from *PETA_cleaned* (as in experiments of

Subsection 4.2.3) has appeared to be prohibitively expensive. The selected images have been protected by five presented PPFs at three different strength levels (*cf.* Table 4.2.8), resulting in $300 \times 5 \times 3 = 4500$ images evaluated in this crowdsourcing study. Each crowdsourcing worker has been asked to look at a body image and answer the question "What is the gender of the person?" with the following options: "male", "female" and "I don't know".

To ensure a statistically significant number of evaluations for each image, 40 subjects were assigned to each image, with a total of 2652 subjects participating in the evaluations. By automatically filtering out 198 unreliable workers, 2454 evaluations have been finally used to estimate human scores.

#### 4.2.4.3 *AlexNet* vs. Human Estimators: Results

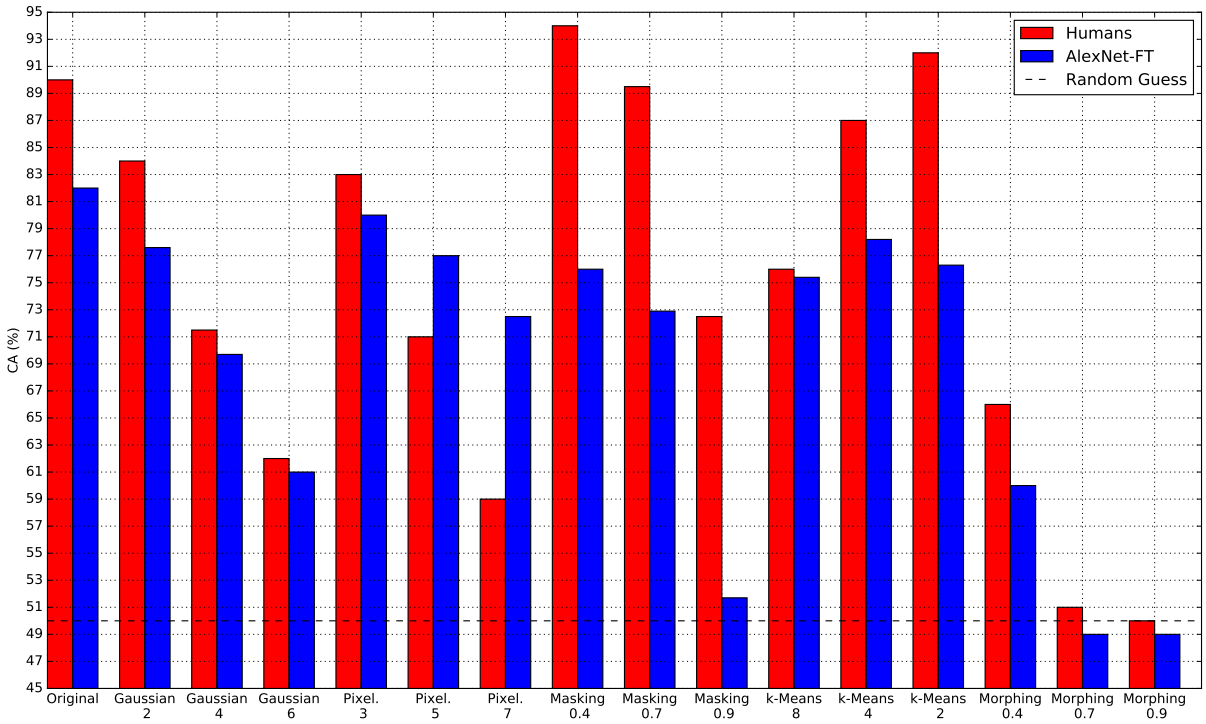



Figure 4.2.4 – Gender recognition from body images: comparison of CAs by human estimators and by our best model *AlexNet-FT* on original images and in presence of PPFs.

Figure 4.2.4 compares the gender recognition CAs of *AlexNet-FT* and human evaluators on original and protected images. The outputs of *AlexNet-FT* are binary ("female", "male") while the human evaluators had three options ("female", "male", "I don't know") during the crowdsourcing. Therefore, in order to calculate CAs for human annotators we assume that 50% of "I don't know" answers are correct.

On original images, humans outperform our *AlexNet-FT* by about 8 points, which indicates the improvement margin for the proposed gender recognition solution. We believe that CA of *AlexNet-FT* can be significantly increased by using a bigger dataset for fine-tuning (as in our experiments, we have used less than 10K images for this purpose which is very little even for transfer learning of a deep CNN). In presence of various PPFs, the performance gap between humans and *AlexNet-FT* is generally reduced with the single exception: Masking PPF, which does not seem to have a major impact on human scores.

### 4.2.5 Summary of the First Preliminary Study

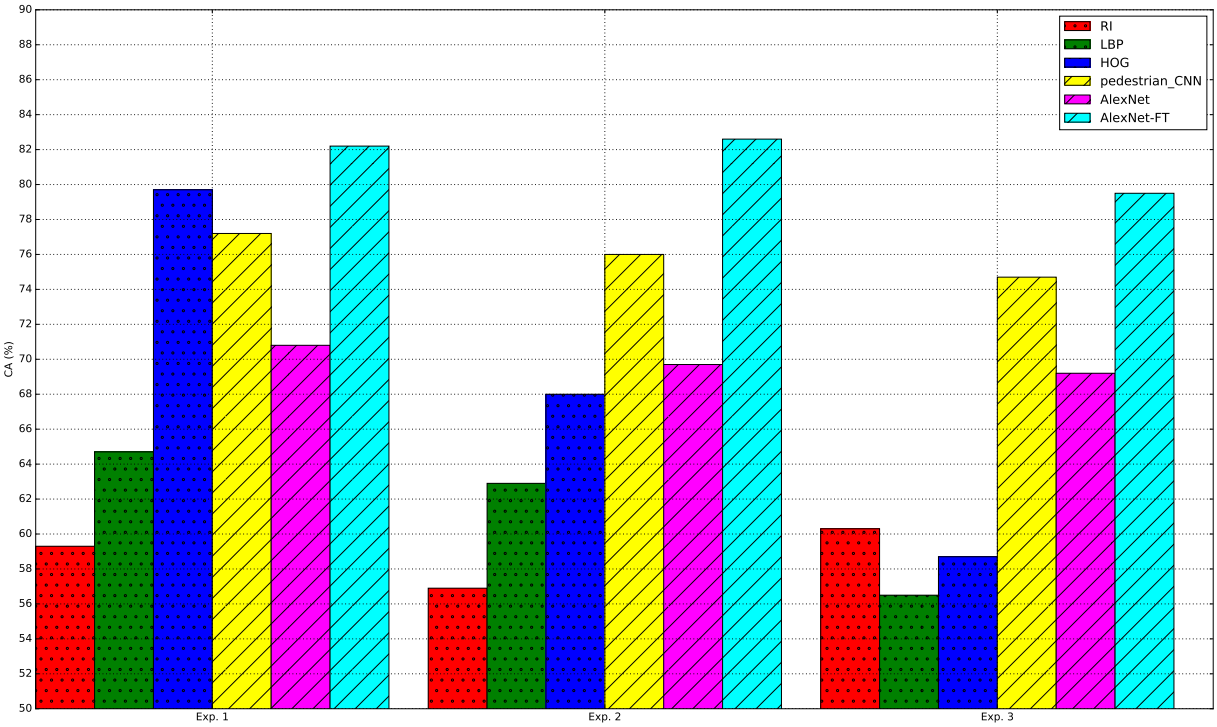



Figure 4.2.5 – Learned vs. hand-crafted features: average results. Exp. 1: homogeneous training data; same-dataset evaluation scenario. Exp. 2: heterogeneous training data; same-dataset evaluation scenario. Exp. 3: heterogeneous training data; cross-dataset evaluation scenario. The details are provided in Subsection 4.2.3.

For convenience, the bar plot in Figure 4.2.5 summarizes the average gender CAs from Tables 4.2.5, 4.2.6 and 4.2.7. All in all, the key findings of this section are the following:

1. HOG is the most suitable hand-crafted features for gender recognition from body images. This finding conforms with the results of many previous studies [Cao+08; BMM11; GS+16].
2. Learned features better adapt to heterogeneous training data and much better generalize to completely unseen test data than hand-crafted features.
3. Fine-tuning for the target domain significantly improves the effectiveness of the transfer learning (this finding is used and confirmed in Chapter 5).
4. A very compact *pedestrian_CNN* trained from scratch on less than 10K images outperforms all hand-crafted features and demonstrates a comparable accuracy with the much deeper *AlexNet-FT*.

Good generalization of the learned features to heterogeneous and unseen data can be explained by the fact that contrary to hand-crafted features, they focus on the semantics of the problem and not on low-level image details (like edges, corners etc.) This general understanding of the problem is learned during the training phase (which is absent for the hand-crafted features) and it allows the learned features to grasp the concept of gender through the large variety of body poses, backgrounds etc. Remarkably, even the original *AlexNet* features better generalize to unseen body images than HOG features. It demonstrates that high-level semantic concepts from the *ImageNet* dataset are transferable to body images.

## 4.3 Study 2: CNN Architecture for Training from Scratch

Among other things, the experiments in Section 4.2 have demonstrated the effectiveness of transfer learning which allows reusing of the CNN-features learned on large multi-class datasets (such as *ImageNet* [Rus+15] or *Places* [Zho+14]) for other problems.

But what if we want to train a CNN from scratch so that the corresponding learned features are better fit to the particular application? The example of *pedestrian_CNN* from the previous Section 4.2 demonstrates that a very compact CNN trained from scratch can obtain comparable performances with a fine-tuned *AlexNet*. However, this example is biased by the fact that few training images are available for gender recognition from bodies. But in case if training data is abundant, does it necessarily mean that the deeper CNN architecture would outperform the shallower one?

In order to answer to this question, in this section, we consider the problem of gender recognition from face images which has been introduced in Chapter 3. Unlike pedestrian images, today, there are big public datasets of face images annotated with gender information which can be used for training. We propose a simple algorithm for CNN architecture optimization which allows us to experimentally show that when trained from scratch, the gender CAs are saturated with a quite shallow CNN. Results of this section highlight the importance of the target problem for the effectiveness of deep learning.

### 4.3.1 Algorithm to Optimize CNN Architecture

In order to design an optimal CNN architecture for gender recognition from face images, we propose a simple but effective algorithm (*cf.* Algorithm 4.1). It takes a complex *start_CNN* architecture at its input and successively optimize it to obtain more compact *optimized_CNN* at the output without decreasing the resulting gender CAs.

More precisely, once *start_CNN* is trained, we evaluate its score on the validation dataset and fix it as a reference accuracy $CA_0$. The subsequent optimization of *start_CNN* should simplify the architecture without deteriorating the reference gender recognition accuracy. Therefore, all intermediate architectures in Algorithm 4.1 are compared against $CA_0$. For the sake of a fair evaluation, every intermediate CNN is trained 3 times which results in 3 different CAs per CNN architecture. When $CA_{current}$ of an intermediate architecture is compared with $CA_0$ (lines 9, 16 and 23 of Algorithm 4.1) both mean values $\mu_{current}$ and standard deviations $\sigma_{current}$ (estimated over 3 evaluations) are taken into account. More precisely, we consider that *start_CNN* and an intermediate *current_CNN* have a similar gender recognition accuracy (*i.e.*, the criterion from lines 9, 16 and 23 of Algorithm 4.1 is NOT fulfilled: $criterion(CA_0, CA_{current}) = False$), when *current_CNN* achieves the reference accuracy with respect to its standard deviation: $CA_0 \leqslant \mu_{current} + \sigma_{current}$ (in other words, we consider one model to be better than another one only if the margin between the two CAs is at least as big as the corresponding standard deviation).

The optimization is composed of three main steps which can be summarized as (1) reducing the CNN's depth; (2) reducing the CNN's width; and (3) reducing the number of fully-connected weights. Below, we detail each of these steps motivating our choices.

The first step of Algorithm 4.1 (lines 3-9) optimizes the number of convolutional blocks in the CNN architecture. Beginning with *start_CNN*, we iteratively remove the convolutional blocks until the gender recognition accuracy deteriorates significantly (in the sense described above). Due to the fact that con-

```
   input  : Initial CNN architecture start_CNN; training and validation datasets
   output: Optimized CNN architecture optimized_CNN
 1 train (start_CNN);
 2 CA_0 := evaluate_CA (start_CNN);
   /* Step 1. Optimizing number of convolutional blocks and retina
      size.                                                          */
 3 current_CNN := start_CNN ;
 4 repeat
 5 |   current_optimal_CNN := current_CNN ;
 6 |   current_CNN := remove_ConvBlock (current_CNN);
 7 |   train (current_CNN);
 8 |   CA_current := evaluate_CA (current_CNN);
 9 until criterion (CA_0,CA_current) or current_CNN .#ConvBlocks < 2;
   /* Step 2. Optimizing number of convolutional feature maps.  */
10 current_CNN := current_optimal_CNN ;
11 repeat
12 |   current_optimal_CNN := current_CNN ;
13 |   current_CNN := half_CNNWidth (current_CNN);
14 |   train (current_CNN);
15 |   CA_current := evaluate_CA (current_CNN);
16 until criterion (CA_0,CA_current) or current_CNN .#ConvFeatureMaps < 2;
   /* Step 3. Optimizing number of neurons in the fully-connected
      layer.                                                         */
17 current_CNN := current_optimal_CNN ;
18 repeat
19 |   current_optimal_CNN := current_CNN ;
20 |   current_CNN := half_FCNeurons (current_CNN);
21 |   train (current_CNN);
22 |   CA_current := evaluate_CA (current_CNN);
23 until criterion (CA_0,CA_current);
24 optimized_CNN := current_optimal_CNN ;
```

**Algorithm 4.1:** Optimization of the CNN architecture. The algorithm assumes that initial *start_CNN* architecture is composed of several convolutional blocks each followed by a factor-2 subsampling and a final fully-connected layer. For simplicity, the number of convolutional feature maps (*#ConvFeatureMaps*) in each convolutional layer and the number of neurons in the fully-connected layer (*#FCNeurons*) is expected to be a power of 2.

volutional blocks are ended by a factor-2 subsampling, when removing each of them, we also half the height and width of retina. This allows preserving the size of feature maps at the last convolutional block.

The optimal architecture after the first step is given at the input of the second one (lines 10-16 of Algorithm 4.1) which objective is to reduce the number of feature maps in each convolutional layer (in other words, the architecture's width). Number of feature maps in all convolutional layers of the initial *start_CNN* is a power of 2, so during every iteration of the second step of Algorithm 4.1, we half the feature maps quantity.

Similarly, during the last third step of the algorithm, we iteratively half the number of neurons in the

| ***start_CNN*** | Step 1 | | | Step 2 | Step 3 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *A* | ***B*** | *C* | *D* | *E* | *F* | *G* | *H* | ***I*** | *J* | *K* | *L* | *M* |
| Input: 128x128 | Input: 64x64 | Input: 32x32 | Input: 16x16 | Input: 32x32 | Input: 32x32 | | | | | | | | |
| Conv: 32@3x3<br>Conv: 32@3x3 | Conv: 32@3x3<br>Conv: 32@3x3 | Conv: 32@3x3<br>Conv: 32@3x3 | Conv: 64@3x3<br>Conv: 64@3x3 | Conv: 16@3x3<br>Conv: 16@3x3 | Conv: 32@3x3<br>Conv: 32@3x3 | | | | | | | | |
| MaxPool: 2x2 | Max-Pool: 2x2 | Max-Pool: 2x2 | Max-Pool: 2x2 | Max-Pool: 2x2 | MaxPool: 2x2 | | | | | | | | |
| Conv: 32@3x3<br>Conv: 32@3x3 | Conv: 32@3x3<br>Conv: 32@3x3 | Conv: 64@3x3<br>Conv: 64@3x3 | — | Conv: 32@3x3<br>Conv: 32@3x3 | Conv: 64@3x3<br>Conv: 64@3x3 | | | | | | | | |
| MaxPool: 2x2 | Max-Pool: 2x2 | Max-Pool: 2x2 | — | Max-Pool: 2x2 | MaxPool: 2x2 | | | | | | | | |
| Conv: 64@3x3<br>Conv: 64@3x3 | Conv: 64@3x3<br>Conv: 64@3x3 | — | — | — | — | | | | | | | | |
| MaxPool: 2x2 | Max-Pool: 2x2 | — | — | — | — | | | | | | | | |
| Conv: 64@3x3<br>Conv: 64@3x3 | — | — | — | — | — | | | | | | | | |
| MaxPool: 2x2 | — | — | — | — | — | | | | | | | | |
| FC: 512 | | | | | FC: 256 | FC: 128 | FC: 64 | FC: 32 | FC: 16 | FC: 8 | FC: 4 | FC: 2 | — |
| Softmax: 2 | | | | | | | | | | | | | |

Table 4.3.1 – Optimization of the *start_CNN* architecture by Algorithm 4.1. "Conv: N@MxM" denotes a convolutional layer with N kernels of size MxM. "MaxPool: MxM" denotes downsampling by a factor of M using Max-Pooling. "FC: N" denotes a fully-connected layer with N neurons.

fully-connected layer. When the third step is completed, the resulting CNN architecture is considered as the algorithm's output *optimized_CNN*.

It is interesting to notice that an approach very similar to Algorithm 4.1 has been later independently proposed by Perez de San Roman et al. [SR+17] for optimizing the CNN architecture in the context of the analysis of egocentric videos.

### 4.3.2 Experiments

In this subsection, we optimize *start_CNN* according to Algorithm 4.1 presented above. Once an optimal CNN architecture is found, we evaluate its performance on a separate test dataset (cross-dataset evaluation protocol) and measure the impact of the training dataset's size on gender recognition accuracy.

The design of the starting architecture (to be optimized) is inspired by that of the CNN proposed by Simonyan and Zisserman [SZ15] (the architecture "*B*" from their work). Following their work, *start_CNN* is composed of several convolutional blocks each containing two successive convolutional

layers with ReLU activations and max-pooling subsampling operations. The kernels of all convolutional layers have a spatial dimension of 3x3 pixels. However, *start_CNN* has several differences from its initial prototype in [SZ15]. In *start_CNN*, the input image resolution is of 128x128 pixels instead of 224x224 pixels. We use a lower resolution because initial resolutions of face images in the training dataset vary approximately from 60x60 to 120x120 pixels, and it does not make sense to significantly upsample input faces. Taking into account the smaller inputs, *start_CNN* contains 4 instead of 5 convolutional blocks. Finally, due to the fact that the gender recognition problem is easier than *ImageNet* classification (2 target classes instead of 1000 classes), we have reduced the number of kernels in the convolutional layers and used only one fully-connected layer. The final architecture of *start_CNN* is detailed in the first column of Table 4.3.1.

Training of all CNNs in this subsection has been carried out by optimizing the binary cross-entropy objective function using the mini-batch Nesterov's accelerated gradient descent ([Nes83]). In order to prevent the CNNs from overfitting, we have employed the "dropout" regularization ([Sri+14]) on the activations of convolutional layers and the fully-connected layer. We have made the ratio of the "dropout" to be dependent on the particular size of the convolutional or the fully-connected layer varying it from 0 (*i.e.* no "dropout") to 0.5. The training has been stopped once the validation accuracy stops improving. In practice, it corresponds to the moment when the training accuracy is between 98.0% and 98.1% (depending on the particular CNN architecture). Training has taken about 30 epochs with slight variations depending on the particular CNN architecture, which corresponds to about 27 hours of training for the *start_CNN* and 2.5 hours of training for the resulting *optimized_CNN* on a contemporary GPU.

#### 4.3.2.1 Datasets

The experiments presented in this section have been carried out on two publicly available face datasets: *CASIA WebFace* [Yi+14] and Labeled Faces in the Wild (*LFW*) [Hua+07]. The first one is used for training and validation whereas the second one is used for testing. Figures 4.3.1-(a) and 4.3.1-(b) present some example images of the two respective datasets. While collecting the *CASIA WebFace* dataset, its authors made sure that there are no subject intersections between *CASIA WebFace* and *LFW* [Yi+14].

***CASIA WebFace* dataset** *CASIA WebFace* dataset was collected for face recognition purposes by [Yi+14]. The dataset contains photos of actors and actresses born between 1940 and 2014 from the IMDb website [3]. Images of the *CASIA WebFace* dataset include random variations of poses, illuminations, facial expressions and image resolutions. In total, there are 494,414 face images of 10,575 subjects (53.5% of men faces and 46.5% of women faces). Images of the *CASIA WebFace* dataset are split between training and validation in proportion 90% to 10%. There are no subject intersections between training and validation parts.

Authors of *CASIA WebFace* provide names of 10,575 subjects but not their genders. We have annotated genders using the metadata provided by IMDb and also by manual annotation.

3. http://www.imdb.com/ Intenet Movie Database (IMDb) is an online database of information related to films, television programs and video games.

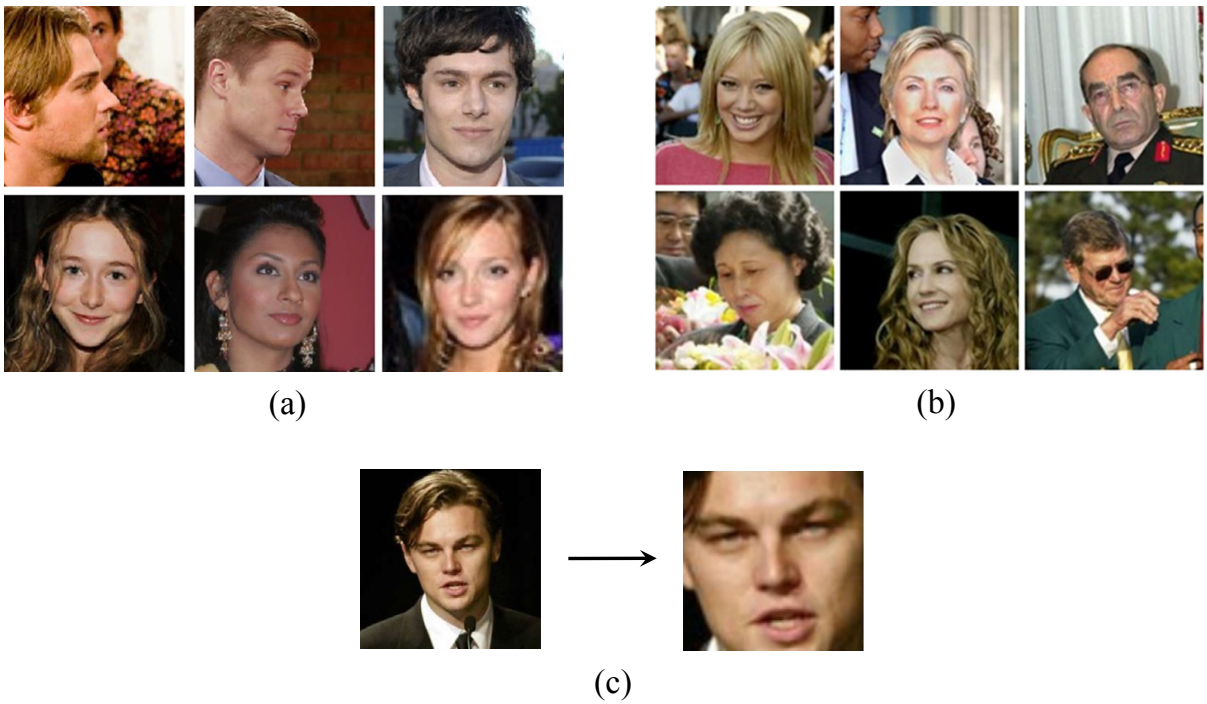


Figure 4.3.1 – Face datasets used in Section 4.3. (a) Examples of images from *CASIA WebFace*; (b) examples of images from *LFW*; (c) illustration of face detection and resizing.

***LFW* dataset** Being collected by [Hua+07], the *LFW* dataset has become a standard benchmark for gender recognition from faces in an unconstrained environment. It consists of 13,233 photos of 5,749 celebrities (76.9% of men faces and 23.1% of women faces). Contrary to *CASIA WebFace*, *LFW* does not only contain photos of actors and actresses but it also contains photos of politicians and other celebrities. [4]

**Face Detection and Normalization** Images of both *CASIA WebFace* and *LFW* are face-centred and have an initial resolution of 250x250 pixels. The two datasets have been processed in the same way: the faces are firstly extracted with an internal face detector (based on [Zha+07]), and then they are rescaled to a certain square size (the particular size depends on the input dimensions of the CNN). This process is illustrated in Figure 4.3.1-(c). Before being processed by a CNN, input RGB face images are normalized by subtracting a mean image and dividing by an image of standard deviations.

When trained on *CASIA WebFace*, *start_CNN* obtains 97.5% and 96.9% of CA on the validation part of *CASIA WebFace* and on *LFW*, respectively. These scores are among the state-of-the-art gender recognition results (*cf.* Chapter 5).

#### 4.3.2.2 Optimization of *start_CNN*

Below, we report the results of optimizing *start_CNN* according to Algorithm 4.1 by subsequently performing the three steps of the algorithm.

**Step 1. Optimizing the retina size and the number of convolutional layers.** In the first step of Algorithm 4.1, we minimize the number of convolutional blocks and the associated retina size. By

4. Gender annotations for the *LFW* dataset are available at http://face.cs.kit.edu/431.php

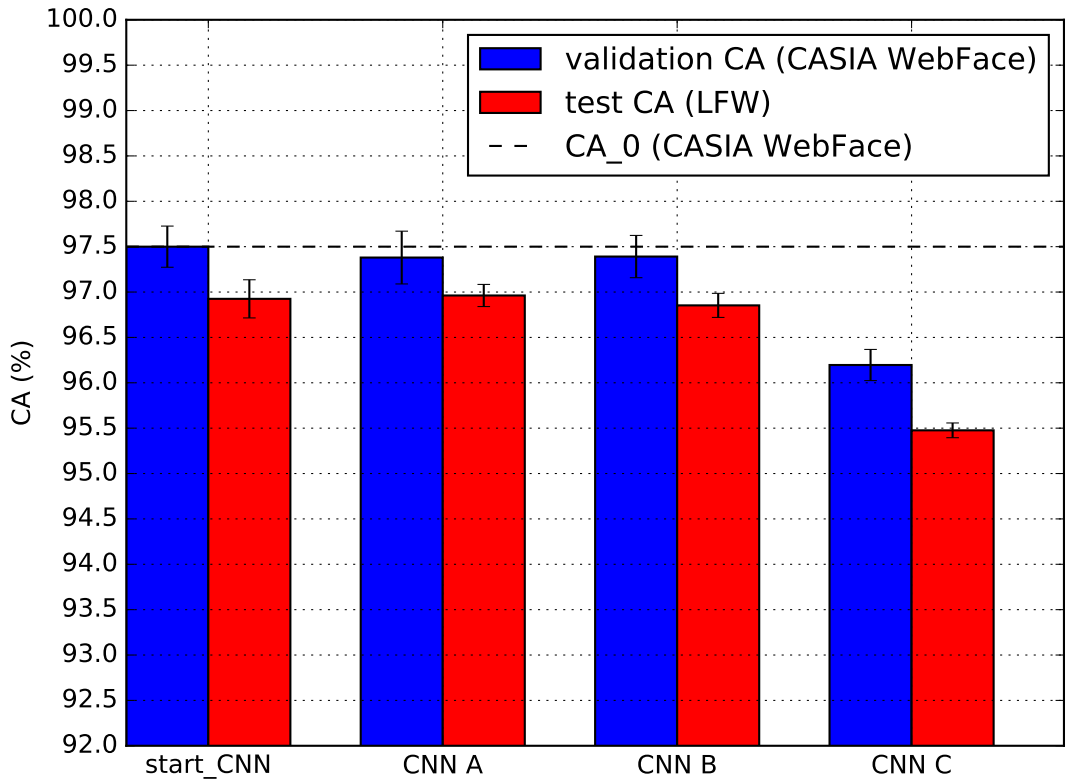



Figure 4.3.2 – Optimization of *start_CNN* according to Algorithm 4.1 — step 1. Optimizing the retina size and the number of convolutional layers: the CNN architecture *B* is selected after step 1.

progressively removing 3 out of 4 convolutional blocks of *start_CNN*, we respectively obtain CNN architectures *A*, *B* and *C* (*cf.* Table 4.3.1-(step 1)). In order to vary the number of convolutional layers while approximatively preserving the global number of weights in the CNN, we keep constant the number and size of the feature maps at the last convolutional layer during the first step (as more than 90% of weights of *start_CNN* are concentrated between the last convolutional and the fully-connected layers).

Figure 4.3.2 compares gender CAs of *start_CNN*, CNN *A*, CNN *B* and CNN *C*. Mean CAs are depicted by bars while corresponding standard deviations are given by error segments (both statistics are estimated using 3 CNN instances trained from scratch). The reference $CA_0 = 97.5\%$ corresponds to the accuracy of *start_CNN* on the validation dataset and is fixed for all three steps of Algorithm 4.1. In order to illustrate how the validation accuracy on the *CASIA WebFace* dataset is related to the test accuracy on the *LFW* dataset, we also present the test accuracies of all compared CNNs in Figure 4.3.2. However, we highlight that the results on the test dataset are not used in the architecture selection.

There is no significant difference between the CAs of *start_CNN*, CNN *A* and CNN *B*. All of them show validation scores which are very close to the reference $CA_0$ (with respect to the criterion defined in Subsection 4.3.1). The accuracy of CNN *C* is significantly lower than accuracies of the first three networks: decrease of about 1.5 points on the validation dataset. Therefore, as the objective is to reduce the complexity of the architecture while preserving the gender recognition accuracy, CNN *B* is selected after this first optimization step.

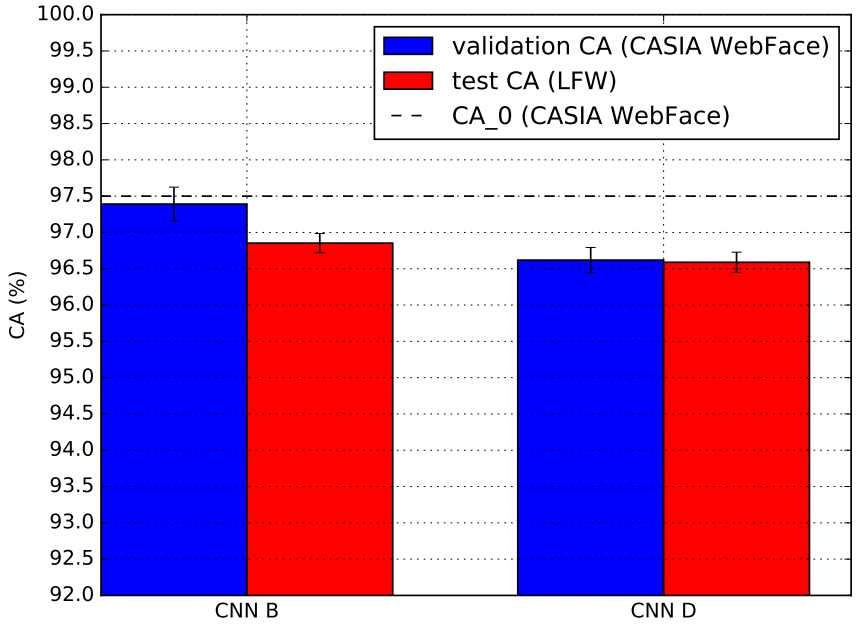



Figure 4.3.3 – Optimization of *start_CNN* according to Algorithm 4.1 — step 2. Optimizing the number of feature maps (the CNN's width): the CNN architecture *B* is selected after step 2.

**Step 2. Optimizing the number of feature maps (the CNN's width).** Following Algorithm 4.1, we half the number of feature maps in all convolutional layers of the CNN architecture *B* selected after the first optimization step obtaining the CNN architecture *D* (*cf.* Table 4.3.1-(step 2)). The two CNN architectures *B* and *D* are compared in Figure 4.3.3. The CA of CNN *D* is clearly below the reference $CA_0$, which implies that the architecture *B* is selected again after the second optimization step.

**Step 3. Optimizing the number of neurons in the fully-connected layer.** Starting from 512 neurons in the CNN architecture *B*, we have reduced the number of neurons by a factor of 2 until only 2 neurons are left in the CNN architecture *L* (*cf.* Table 4.3.1-(step 3)). We also evaluate the performances of the fully-convolutional CNN architecture *M* which does not contain fully-connected layers at all. The results are summarized in Figure 4.3.4.

This time, the difference between compared CNNs is less significant than in the first two optimization steps indicating that the size of the fully-connected layer is less influential on the resulting gender recognition accuracy than the number and the width of the convolutional layers. Nevertheless, we observe that CNNs *B*, *E* — *I* reach the reference threshold of 97.5% of CA, while the performances of CNNs *J* — *M* are below the threshold on the validation dataset. Hence, the CNN architecture *I* is selected after the last optimization step and is considered as the final output of the optimization algorithm: *optimized_CNN* := CNN *I*.

#### 4.3.2.3 Impact of the Training Dataset Size

In order to assess the impact of the training dataset's size on the resulting gender recognition CAs, we have trained several instances of *optimized_CNN* varying the number of training images. The corres-

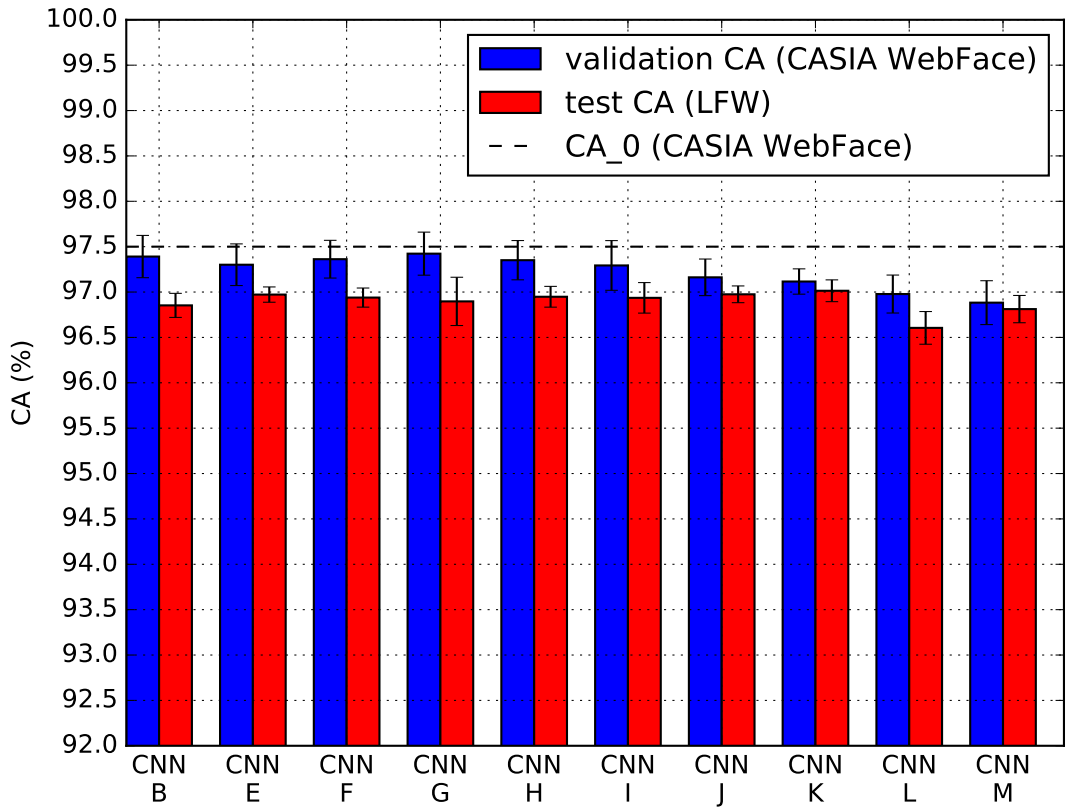



Figure 4.3.4 – Optimization of *start_CNN* according to Algorithm 4.1 — step 3. Optimizing the number of neurons in the fully-connected layer: the CNN architecture *I* is selected after step 3.

ponding subsets of images have been selected randomly from the initial training dataset. As in previous experiments of this section, for each considered dataset size, we have trained 3 instances of *optimized_CNN*. Results of the comparison are summarized in Figure 4.3.5.

The accuracy of *optimized_CNN* trained only on a half of available images does not distinguish significantly from the accuracy of *optimized_CNN* trained on all data. Further reducing of the training dataset down to 25% and 10% of its initial size leads to a loss of performance. However, these losses are relatively small. Thus, *optimized_CNN* which is trained on only 10% of the available data (*i.e.* on about 45K images) performs reasonably well: 96.0% of correct gender predictions on *LFW* (it is better than, for example, the model by [Sha12] which used to be the state-of-the-art on *LFW* in 2012).

#### 4.3.2.4 Gender Recognition Accuracy on *LFW*

The gender CA of *optimized_CNN* on the test *LFW* dataset is of 96.9% which is the state-of-the-art result among CNNs trained from scratch. When combining three instances of *optimized_CNN* in a single ensemble model, the CA is marginally improved reaching 97.3% (the model fusion is done on the predictions level). We have not observed considerable ameliorations of this result by combining more than three CNNs in one ensemble.

The full comparison of *optimized_CNN* (and other gender recognition models developed by us which

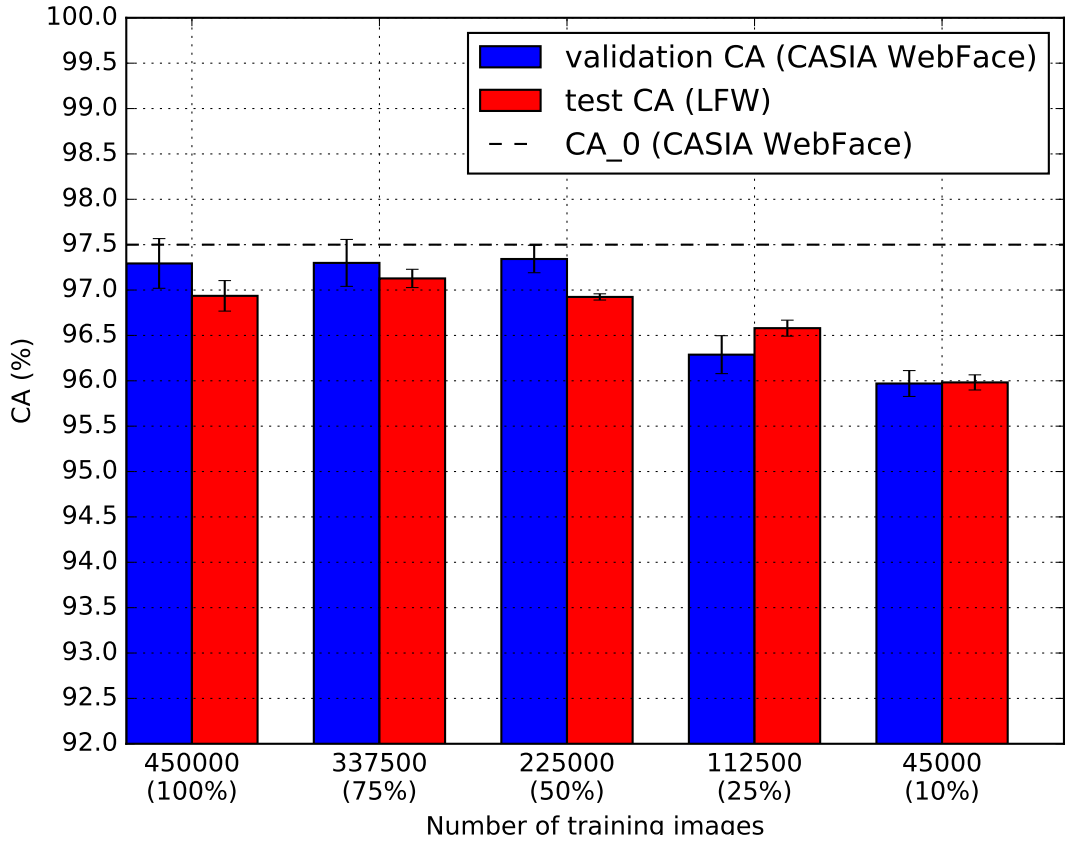



Figure 4.3.5 – Impact of the number of training images on gender CA of *optimized_CNN*.

| Reference | Features | CA on *LFW* |
|---|---|---|
| [JC15] | Hand-crafted | 96.9% |
| *optimized_CNN* | Learned | 96.9% |
| Ensemble of 3 *optimized_CNN* | | 97.3% |
| [CSLNRB16] | Learned and Hand-crafted | 98.0% |

Table 4.3.2 – Evaluation of the optimized CNN *I* on the test *LFW* dataset.

are presented further in the manuscript) with the state-of-the-art is provided in Chapter 5. However, Table 4.3.2 compares gender CAs of *optimized_CNN* with two other models which, in our opinion, are very illustrative. The first model by Jia and Cristianini [JC15] is the state-of-the-art among the approaches which apply only hand-crafted features for face gender recognition, and the second one by Castrillon-Santana et al. [CSLNRB16] combines learned (by a shallow CNN) and hand-crafted features. Our *optimized_CNN* and the model by Jia and Cristianini [JC15] obtains the same CAs of 96.9%. On the contrary, the model by Castrillon-Santana et al. [CSLNRB16] outperforms our *optimized_CNN* by about 1 point which is a very significant improvement for face gender recognition.

Therefore, contrary to the pedestrian gender recognition problem considered in Section 4.2, in case of gender recognition from faces, the hand-crafted features (in particular, LBP) are as good as the features learned from scratch. Moreover, combining the two leads to the increase in performances. We believe that there are two explanations to the observed results:

1. Body images have more variety than face ones in terms of poses, environments, occlusions etc. Hence, the advantages of learned features (their abilities to better adapt to heterogeneous data and to better generalize to unknown data demonstrated in Section 4.2) are less visible in case of gender recognition from face images.
2. Gender recognition training does not result in learning as rich and expressive features as those learned by training for more general and complicated tasks (for example, *ImageNet* classification). This point is detailed in Chapter 5, where we demonstrate that face recognition pretraining drastically improves the gender recognition accuracy.

### 4.3.3 Summary of the Second Preliminary Study

In this section, we have optimized a CNN architecture for a particular problem in the frame of training from scratch. For our experiments, we have chosen the problem of gender recognition from face images which on the one hand, is not trivial (*cf.* Chapter 3) and on the other hand, is easier than other discriminative problems considered in this manuscript: gender recognition from pedestrian images, age estimation and face recognition. For optimization of the CNN architecture, we propose a simple but efficient Algorithm 4.1. Our main conclusions are the following:

1. In a CNN, the number and the width of convolutional layers has more impact on the resulting performances than the size of the fully-connected layer. This finding conforms with several recent studies [LCY14; Sze+15] (and is used in Chapter 5).
2. When trained from scratch, a very compact gender recognition *optimized_CNN* is enough to saturate the gender recognition accuracy (we confirm this finding in Chapter 5).
3. When trained from scratch, gender recognition accuracy is not improved by using more than 250K training images.
4. The obtained test *LFW* CA of 96.9% is on par with the CA obtained by Jia and Cristianini [JC15] with LBP features and a SVM-like algorithm.

On the one hand, the designed *optimized_CNN* reaches decent gender recognition performances which are among the state-of-the-art, and the network is compact enough to be used in real-time even on embedded devices with very limited resources. But on the other hand, the fact that a shallow *optimized_CNN* appears to be optimal reveals that the gender recognition problem is not demanding enough and cannot take the full advantage of deep learning. In Chapter 5, we elaborate more on this issue and remedy it by pretraining on a more challenging problem.

## 4.4 Conclusion

There is a common confusion that all the power of deep learning relies just on two pillars: (1) deep neural architectures and (2) huge datasets. While both of them are very important, in this chapter, we show that the two are neither essential nor a guarantee of the top performances.

Indeed, in Section 4.2, we demonstrate that the learned features better adapt to heterogeneous or unseen data than the hand-crafted ones. In other words, the learned features has proved to generalize better than hand-crafted ones despite the tiny training dataset of less than 10K images and the fact that one

of the employed CNN architectures (*pedestrian_CNN*) is very shallow (only two convolutional layers). Moreover, we prove that the problem of the lack of training data can be effectively resolved by transfer learning even when initial and target domains are quite different (*ImageNet* classification and gender recognition from pedestrian images, respectively).

In addition to that, in Section 4.3, it is shown that when trained from scratch, the performances of face gender recognition CNNs are saturated with about 250K training images. More importantly, a very compact *optimized_CNN* architecture is enough to obtain the maximum gender recognition CA, the one which can be obtained with only classical hand-crafted features. This result indicates that the problem of gender recognition from face images is not sufficiently challenging to learn expressive features from scratch which are able to considerably improve the hand-crafted baseline.

The last observation leads us to the main take-away message of this chapter: the effectiveness of learned features (and deep learning, in general) depends not only on the used neural architecture and the size of the training dataset, but also on the difficulty of the problem on which they are trained. Thus, in Chapter 5, we show that face recognition and age estimation problems require bigger CNN architectures and more training data than gender recognition problem, and we demonstrate that a complex task pretraining can drastically improve the accuracy of a face gender recognition CNN.

Chapter 5

# Gender/Age Prediction from Face Images

**Contents**



## 5.1 Introduction

In the present chapter, we detail one of the two central contributions of this manuscript: design of the state-of-the-art CNNs for gender and age prediction. The interest of the research in gender recognition and age estimation from face images is catalysed by the constantly growing demand in the fields of targeted advertising, multimedia analysis and security. We refer the authors to Chapters 1 and 3 for the

list of motivations and the summary of the most notable non-CNN studies on the topics. Below, we focus exclusively on technical aspects of CNN training for the considered problems.

Preliminary studies in Chapter 4 have shown that training of CNNs can be very tricky. Indeed, neither a state-of-the-art CNN architecture nor a huge training dataset on their own has appeared to be a guarantee of top performances. Hence, in Section 5.2 of this chapter, we begin with outlining the optimal practices for gender recognition and age estimation CNNs. To this end, we firstly analyse the existing CNN-based approaches for gender and age prediction, and underline the CNN design and training parameters which mostly differ between them. Then, we methodically compare the selected parameters looking for configurations which facilitate the training of gender and age prediction CNNs.

The optimal strategies of the CNN design and training which are identified in Section 5.2 are further used in Section 5.3 to obtain our best performing models for gender recognition and age estimation. In practice, we take advantage of the synergy between the found training strategies and the state-of-the-art CNN architectures, namely: *VGG-16* [SZ15] and *ResNet* [He+16]. We show that the designed models outperform previous gender recognition and biological age estimation approaches on three mostly used benchmark datasets. Moreover, our best biological age estimation CNN favourably compares with motivated human participants of a popular French TV show "Guess My Age" (*cf.* Paragraph 5.3.3.3).

Finally, as explained in Chapter 3, in the state-of-the-art, biological and apparent age estimation problems are distinguished. In order to demonstrate that the found CNN training and design principles hold for apparent age estimation as well, we have participated and won the first place in the international competition on apparent age estimation [Esc+16] (*cf.* Section 5.4).

## 5.2 CNN Design and Training Strategy

The objective of this section is to find the optimal design and training parameters for gender recognition and age estimation CNNs.

To this end, we firstly refer to the existing works on CNN gender and age prediction in Subsection 5.2.1 highlighting and analysing the differences between them. As a result, we come up with five CNN training and design parameters which are further studied below. These parameters are the following: (1) the age encoding and the loss function for the age CNNs, (2) the alignments of the face images which are given at the input of the CNNs, (3) the depth of the employed CNN architectures, (4) the use of pretraining, and (5) the training strategy: mono-task vs. multi-task. After that, in Subsection 5.2.2, we present the mentioned CNN parameters in details highlighting their importance and motivating our choices, and in Subsection 5.2.3, we experimentally compare various gender and age CNNs which are trained by modifying the studied parameters. The conclusions of this section are used to train the deep top performing gender and age CNNs in the following Section 5.3.

### 5.2.1 Previous Studies on Gender and Age Prediction with CNNs

Due to the overwhelming popularity of deep learning for computer vision tasks, a significant number of recent studies have applied deep CNNs for gender recognition and age estimation. In this subsection, we organize these works highlighting the key differences between them. Non-CNN studies on gender and age prediction have been presented earlier in this manuscript (in Chapter 3).

**CNN Architecture and Pretraining** One of the most evident differences between various CNN models is the choice of the network architecture. CNNs can be roughly split into shallow networks (*i.e.* up to 5-6 convolutional layers) and deep networks with more convolutional layers. We have observed that in general, the studies [CSLNRB16; YLL14; WGK15; Yan+15a; LH15; Ekm16] which train gender/age CNNs from scratch use shallow architectures, while the works employing deeper architectures (like *AlexNet* or *VGG-16/19*) fine-tune already pretrained CNNs [RTVG16; Liu+15a; Zhu+15; Liu+17; OAE16]. Indeed, for all well-known deep CNN architectures, there are publicly available CNN instances already pretrained on the *ImageNet* dataset which gives the possibility of employing transfer learning (in a similar way as it has been illustrated in Section 4.2 of Chapter 4).

Moreover, two pretraining types (the general task [1] one and the face recognition one) were used for the demographics estimation. Their fitness for the target problems was studied by Ozbulak et al. [OAE16]. However, the results of Ozbulak et al. are difficult to interpret given the fact that two types of pretraining are compared on two different architectures: *AlexNet* and *VGG-16*.

For objective evaluation of the impact of the CNN architecture and pretraining on the resulting gender and age accuracies, we separate these two aspects in the experiments of the present section. In particular, we demonstrate that face recognition pretraining is effective regardless of the CNN architecture while the importance of the latter varies between gender recognition and age estimation problems.

**Format of Face Images** Perhaps surprisingly, there is no widely adopted "standard" of what is understood by face images in previous studies on gender recognition and age estimation from face images. Thus, the vast majority of the CNN-based approaches use square-sized face images, but there is no consensus of what part of the face is given to CNNs. Indeed, due to the fact that faces are rather oval than square shaped, there is a dilemma between focusing only on central part of the face (cropping the top of the forehead and the bottom of the chin) as in [Han+15], or providing the CNNs with full head photos (which probably contain some face unrelated information on the sides) as in [LH15].

In this section, we show that gender and age prediction CNNs gain in performances when they are fed with images containing full heads of SoI even if these photos contain some background information.

**Loss Functions for Age CNNs** Loss functions and age encoding strategies are another source of variation between different age estimation CNNs. The vast majority of CNN-based age models were trained with either pure classification [LH15; OAE16; RTVG16] or with pure metric regression objectives [YLL14; Liu+15a; Zhu+15].

The two approaches have opposite advantages and downsides. Thus, the classification training is more robust to outliers and to incorrect annotations, while the regression training better reflects the continuous aspect of the human age. Therefore, a number of studies have attempted to design complex loss functions composed of both classification and regression parts [DLL16; Liu+17]. Similarly, ordinal ranking employed by [Yan+15a; Niu+16] can also be seen as an intermediate approach between classification and regression. As explained in Chapter 3, in the ordinal ranking formulation performs, a real-valued age estimation is calculated as a linear combination of the results of $(K-1)$ binary age classification.

1. Here and below in this chapter, by the "general task pretraining", we understand the classification pretraining on the *ImageNet* dataset of 1000 classes.

A common downside of the mentioned approaches is their complexity: the respective loss function are composed of several parts (which often balance each other in a trade-off) increasing the difficulty of the CNN optimization. Instead, in the present section, we adopt the idea of employing distributed age encodings which was firstly proposed in [GYZ13]. Distributed age encodings reflect both the continuity and the uncertainty of age annotations, and, as shown below, allow a natural formulation of age estimation as a continuous (soft) classification problem.

**Mono-task vs. Multi-task Learning** Finally, several studies [YLL14; Yan+15a] compared mono-task learning for gender recognition and age estimation versus simultaneous learning for both tasks. The results of the two studies seem contradictory: Yi el al. [YLL14] reported no difference between mono-task and multi-task learnings, while Yang et al. [Yan+15a] obtained an improvement in age estimation accuracy from the multi-task learning.

Contrary to the mentioned works, in the present section, we compare the two learning approaches on a test dataset which has been explicitly balanced between genders and between different age categories. As a result, we find that the effectiveness of multi-task learning is of the same nature as the one of face recognition pretraining.

### 5.2.2 Studied Parameters

| Parameter | Tested Values | |
|---|---|---|
| | Gender CNN | Age CNN |
| Target Age Encoding | N/A | 0/1-CAE |
| | | RVAE |
| | | LDAE |
| Face Crop | "face-only" | |
| | "face+40%" | |
| CNN Depth | 2 conv. layers | |
| | 4 conv. layers | |
| | 6 conv. layers | |
| | 8 conv. layers | |
| Pretraining / Multi-task Learning | No pretraining, mono-task | |
| | FR pretraining, mono-task | |
| | No pretraining, multi-task | |
| | FR pretraining, multi-task | |

Table 5.2.1 – CNN design and training parameters for gender recognition and age estimation CNNs which are evaluated in Section 5.2. FR = Face Recognition.

Table 5.2.1 summarizes the CNN design and training parameters which are evaluated in the present section. Below, we subsequently define each of them highlighting their importance for gender recognition and age estimation CNNs.

#### 5.2.2.1 Target Age Encoding and Loss Function

Target encoding defines how the target labels (in our case, genders and ages) are represented in a neural network. Both the information which is given (or not) to the neural network during the training

and the choice of the loss function for optimization depend on target encoding.

Gender recognition is a binary classification problem which does not leave much liberty for the choice of the target encoding and the loss function to optimize. Binary classification problems are solved by neural networks with one or two neurons at the output layer. In the first case, the logistic regression loss function is employed for optimization and in the second case, the cross-entropy one. Cross-entropy loss is mathematically equivalent to logistic one in case of binary classification, so there is no need for experimental comparison of the two losses. If not said otherwise, we train gender recognition CNNs with two neurons at the output layer and the cross-entropy loss (in the same way as in the preliminary studies of Chapter 4).

Contrary to gender recognition, the age estimation problem can be approached in many different ways: classification with coarse categories, per-year classification, regression or even ranking (*cf.* Subsection 5.2.1). Each case imposes particular age encoding and loss function. In this section, we compare three strategies which proved to be the most effective during the first edition of ChaLearn Apparent Age Estimation Challenge [Esc+15]: pure per-year classification (employed by the 1st place winner [RTVG16]), pure regression (employed by the runner-ups [Liu+15a; Zhu+15]) and soft classification (employed by the participants who got the 4th place [Yan+15b]). It is important to highlight that the results of the ChaLearn Challenge cannot be regarded as a fair comparison between the mentioned age encoding strategies because many other factors influence the final performances of age estimation methods (each team used different CNN architectures, pretraining types, training datasets etc.)

| Encoding | Loss function |
|---|---|
| 0/1-CAE | $L_{CAE} = -\frac{1}{N}\sum_{k=1}^{N}\sum_{i=1}^{100} t_i^{(k)} \log p_i^{(k)}$ |
| RVAE | $L_{RVAE} = \frac{1}{N}\sum_{k=1}^{N}(t^{(k)} - p^{(k)})^2$ |
| LDAE | $L_{LDAE} = -\frac{1}{N}\sum_{k=1}^{N}\sum_{i=1}^{100}(t_i^{(k)} \log p_i^{(k)} + (1-t_i^{(k)}) \log (1-p_i^{(k)}))$ |

Table 5.2.2 – Age encodings and corresponding loss functions. $N$ denotes the number of images in a mini-batch, $t$ denotes the targets and $p$ denotes the predictions of CNNs. 0/1-CAE = 0/1-Classification Age Encoding. RVAE = Real-Value Age Encoding. LDAE = Label Distribution Age Encoding.

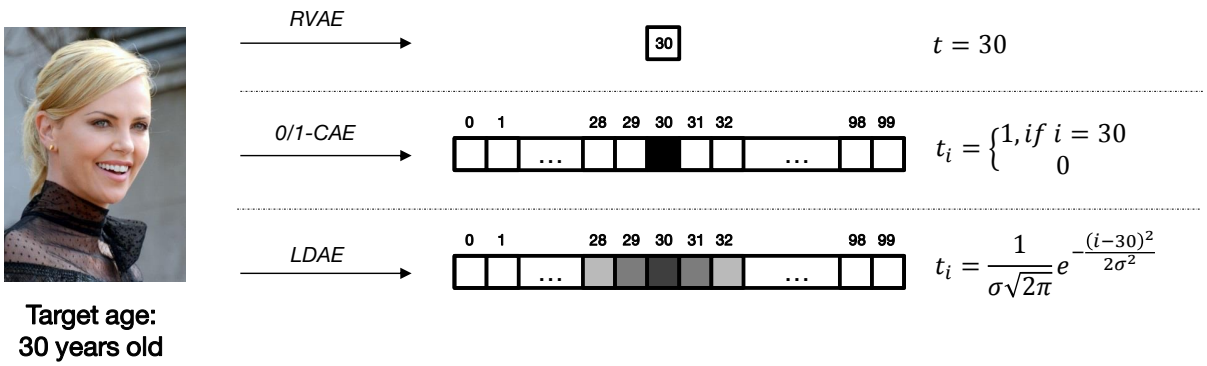



Figure 5.2.1 – Example of age encodings. $t$ denotes the resulting encoding. $\sigma$ is a hyper-parameter of Label Distribution Age Encoding (LDAE). In this work, we use $\sigma = 2.5$ (by experimenting with various $\sigma \in [1,4]$, we have not experienced a significant impact of the $\sigma$ value on the resulting performance).

Below, we detail each of the three encodings.

In **pure per-year classification**, each age (with a precision up to one year) is treated as a separate class which implies that the age label is encoded as a one-hot 1D-vector. The size of this vector corresponds to the number of classes (in this work, we use 100 classes for ages between 0 and 99 years old). We further refer to this encoding as *0/1-Classification Age Encoding (0/1-CAE)*.

**Pure regression** has real numbers as targets, therefore real age values are used as labels in this case. This straightforward age encoding is referred as *Real-Value Age Encoding (RVAE)* in our work.

Finally, **soft classification** can be seen as an intermediate case between the discrete classification and continuous regression. As in pure classification, ages are encoded by vectors of the dimension which corresponds to the number of classes. However, instead of being binary, the values in the vector are encoded with Gaussian distribution centred at the target age. This allows to encode a notion of neighbourhood between different age classes (which is present in RVAE but does not exist in 0/1-CAE): for example, LDAE encodes the information that the age of 20 years old is closer to the age of 21 years old than to the age of 80 years old. Following the work where the encoding was introduced [GYZ13], we refer to it as *Label Distribution Age Encoding (LDAE)*.

Table 5.2.2 presents the discussed encodings as well as the corresponding loss functions, and Figure 5.2.1 provides an example of how they are used to encode an age of an example face image. An experimental evaluation of the compared age encodings is provided in Paragraph 5.2.3.2.

#### 5.2.2.2 Face Crop

Face images which are given at the input of CNNs define the face part which is received by a CNN and which is further used to predict gender and age. For clarity, here and below, we refer to such images as *face crops*. Thus, in Subsection 5.2.1, we highlight that previous studies often significantly vary in their interpretations of what is understood by face crops.

In Chapter 3, it is explained that a typical pipeline of face crop extraction consists of (1) face detection, and (2) face alignment stages. The second stage, face alignment, is optional, and in its turn, it consists of (a) face landmark detection, and (b) alignment itself (*cf.* Chapter 3 for details). In the preliminary study presented in Section 4.3 of Chapter 4, we have not performed face alignment for the sake of simplicity. On the contrary, in the present section, our goal is to find good practices for design and training of the gender recognition and age estimation CNNs. To this end, we look for an optimal face alignment which would allow the resulting face crops to contain the most useful information for gender and age prediction. Indeed, a particular face alignment defines the positions of a certain set of landmarks, and therefore, it allows *controlling* which face parts (*e.g.* front, chin, ears or hair) are included in face crops and which are not.

Thus, in the present section, we compare two types of face crops, namely: “face-only” and “face+40%” (*cf.* Figure 5.2.2), which correspond to two manually predefined sets of 27 landmark points. We have designed both crops to be square-sized in order to facilitate the subsequent treatment by the state-of-the-art CNN architectures (such as *VGG-16* [SZ15] or *ResNet* [He+16]) which expect square inputs. At the same time, an average human face is oval and not square. Therefore, we can either focus only on the central part of a face by discarding the extremities (as in “face-only” crop), or allow some background information on the sides, but preserve all face information (as in “face+40%” crop). In other words, the choice between the “face-only” and “face+40%” crops represents a trade-off between (1) feeding a CNN

(a) (b) (c)

Figure 5.2.2 – (Better viewed in color). Face crops for gender recognition and age estimation which are compared in Section 5.2. (a) Initial image. (b) "face-only" crop. (c) "face+40%" crop.

with only useful but (probably) incomplete information, or (2) feeding a CNN with complete information but (probably) with some side noise, respectively.

In order to optimally resolve this trade-off, in Paragraph 5.2.3.2, we experimentally compare "face-only" and "face+40%" crops between each other. However, it should be noted that when rescaled to the same size (*cf.* Figures 5.2.2-(b) and 5.2.2-(c)), the two face crops have different resolutions of face details. For example, the distance between the eyes (measured in pixels) is two times bigger in "face-only" crop than in "face+40%" one. Therefore, for the sake of a fair comparison, in the experiments of Paragraph 5.2.3.2, we use CNNs with identical architectures, but different retina sizes: 32x32 in case of "face-only" and 64x64 (*i.e.* upscaled by a factor of 2) in case of "face+40%" crops.

#### 5.2.2.3 CNN Depth

As already outlined on multiple occasions in Chapters 2 and 4, the depth of a neural network (*i.e.* number of its hidden layers) has a fundamental importance in deep learning as it allows to learn a hierarchy of image descriptors for a particular problem starting from the elementary features in the early hidden layers until the high-level problem-dependent features in the last ones [Ben+09; ZF14]. Informally, the more complicated is a problem, the deeper CNN architecture is required to address it.

Several recent works [LCY14; Sze+15] as well as the experiments in Section 4.3 of Chapter 4 have shown that fully-convolutional CNNs (*i.e.* CNNs composed of only convolutional layers with no fully-connected ones) perform almost identically to classical CNNs while having much less trainable parameters. It suggests that the discriminative power of a CNN depends rather on its convolutional layers than on its fully-connected ones.

Therefore, in the present chapter, we evaluate only the impact of the number of *convolutional* layers on the quality of gender/age CNNs. In particular, by increasing the CNN depth from 2 and up 8 convolutional layers, we compare its relative impacts on the corresponding gender and age prediction accuracies in Paragraph 5.2.3.2.

#### 5.2.2.4 Pretraining and Multi-Task Learning

Despite pretraining and multi-task learning may seem as two completely independent techniques at the first sight, both of them can be considered as particular cases of transfer learning (already used in

chapter 4). Indeed, the idea of transfer learning is that the knowledge learned from one problem can be reused for the other one. It is both reflected in pretraining and multi-task learning.

In case of pretraining, CNN is initialized by training on a separate complex problem for which there is a lot of training data. The rich internal CNN representations which are learned during pretraining facilitate the further CNN training (fine-tuning) for a problem of interest.

Thus, unlike Section 4.2 of Chapter 4 (where we have used a general task pretraining), in this chapter, we have selected face recognition as a pretraining task due to the following two intuitions. Firstly, contrary to gender recognition and age estimation problems, face recognition allows training very deep CNNs from scratch as in [Tai+14; PVZ15; SWT15] which proves that this problem is difficult enough to serve as a strong CNN regularizer during the training. Secondly, being a face-related task, face recognition is close to our target problems. Indeed, gender is a part of a person's identity, therefore gender recognition can be seen as an elementary sub-problem of face recognition. Though age is clearly independent of a person's identity, it was shown that the face representation learned by a CNN which is trained for face recognition implicitly encodes elementary age information [Liu+15b].

A multi-task CNN is trained to resolve several problems (in our case, gender recognition and age estimation) at the same time. This way, the CNN learns to extract more information from input images than in case of mono-task training which also results in richer internal CNN representations.

In Paragraph 5.2.3.2, face recognition pretraining and multi-task learning are evaluated both separately and simultaneously in the frame of the two studied problems.

### 5.2.3 Experiments

We firstly define the experimental protocol which is used for evaluation of all tested CNN parameters in this section. The protocol consists of the set of the CNN architectures with varying number of convolutional layers as well as of the training and test datasets.

#### 5.2.3.1 Experimental Protocol

***fast_CNN* Architecture** For our experiments, we have designed a set of compact CNN architectures of varying depths: *fast_CNN_2*, *fast_CNN_4*, *fast_CNN_6* and *fast_CNN_8* with 2, 4, 6 and 8 convolutional layers, respectively. These CNN architectures are presented in Table 5.2.3. All of them are used to evaluate the impact of the CNN depth on gender recognition and age estimation accuracies, while the experiments on target age encoding, face crops and transfer learning are performed with the middle-size architecture *fast_CNN_4*, which is further referred as *fast_CNN* for simplicity.

We have opted for quite compact CNN architectures (comparing to the state-of-the-art ones like *VGG-16* [SZ15], GoogLeNet [Sze+15] and ResNet [He+16]). Indeed, the goal of this section is the objective comparison of the presented above CNN parameters rather than the design of the best performing gender/age CNNs. The latter is done in Section 5.3, where we train very deep state-of-the-art CNNs using pretraining based on the conclusions of the present section.

As it is the case of our *start_CNN* from Section 4.3 of Chapter 4, *fast_CNN* follows the same basic design principles as *VGG-16* CNN [SZ15]. In particular, (1) all convolutional layers are composed of square feature maps with kernels of size 3x3 pixels, and (2) max-pooling layers reduce both heights and widths of feature maps in 2 times. In order to facilitate convergence and to prevent overfitting, we employ

| *fast_CNN_2* | *fast_CNN_4* | *fast_CNN_6* | *fast_CNN_8* |
|---|---|---|---|
| Input retinal size depends on the used face crops: 32x32 for "face-only", 64x64 for "face+40%" | | | |
| Conv1_1: 32@3x3 | Conv1_1: 32@3x3 | Conv1_1: 32@3x3 | Conv1_1: 32@3x3 |
| — | Conv1_2: 32@3x3 | Conv1_2: 32@3x3 | Conv1_2: 32@3x3 |
| — | — | Conv1_3: 32@3x3 | Conv1_3: 32@3x3 |
| — | — | — | Conv1_4: 32@3x3 |
| MaxPool: 2x2 | MaxPool: 2x2 | MaxPool: 2x2 | MaxPool: 2x2 |
| Conv2_1: 32@3x3 | Conv2_1: 32@3x3 | Conv2_1: 32@3x3 | Conv2_1: 32@3x3 |
| — | Conv2_2: 32@3x3 | Conv2_2: 32@3x3 | Conv2_2: 32@3x3 |
| — | — | Conv2_3: 32@3x3 | Conv2_3: 32@3x3 |
| — | — | — | Conv2_4: 32@3x3 |
| MaxPool: 2x2 | MaxPool: 2x2 | MaxPool: 2x2 | MaxPool: 2x2 |
| FC: 512 | FC: 512 | FC: 512 | FC: 512 |
| Experiment-specific output layer | | | |

Table 5.2.3 – CNN architectures which are used in experiments of Section 5.2. "Conv: N@MxM" denotes a convolutional layer with N kernels of size MxM. "MaxPool: MxM" means that input maps are downsampled by a factor of M using Max-Pooling. "FC: N" denotes a fully-connected layer with N neurons.

a batch normalization module [IS15] before ReLU activations and a 0.5-dropout module [Sri+14] on the fully connected layer. As explained in Paragraph 5.2.2.2, *fast_CNN* is fed with either 32x32 or 64x64 RGB-images for "face-only" and "face+40%" crops, respectively. Contrary to Section 4.3 of Chapter 4, *fast_CNN* does not require prior normalization of face crops given the fact that batch normalization is employed. The design of the output (fully-connected) layer as well as the corresponding loss function depend on the particular problem (gender recognition or age estimation) and on the age encoding type (in case of age estimation).

**Training Dataset: *IMDB-Wiki_cleaned*** In order to train gender recognition and age estimation CNNs on the same data, we need a dataset with both gender and age annotations. The currently biggest public dataset with these properties was collected by Rothe et al. [RTVG16] and called *IMDB-Wiki* following the two sources of the face images. Indeed, the dataset consists of 523,051 images collected from IMDb (460,723 images) and Wikipedia (62,328 images). Due to the fact that each image contains a celebrity (whose identity, gender and birth date are known) and a timestamp, the authors managed to automatically annotate all images in the *IMDB-Wiki* dataset with genders and ages.

However, for the majority of images from *IMDB-Wiki*, the provided annotations are not directly usable. The problem comes from the fact that a lot of original images contain more than one person. Assuming that all faces in the image are detected, it is not obvious how to automatically select the face to which the given annotation corresponds to. To circumvent this problem, we have pursued the following two approaches:

1. We have used those images for which the "Head Hunter" face detector [Mat+14] has detected only one face. In this case, we can be almost sure (the approximatively estimated error rate is less than 1%) that the detected face corresponds to the provided age annotation. This approach has resulted in 182,019 images.
2. We have developed a simple web interface to manually annotate the remaining images. Given an

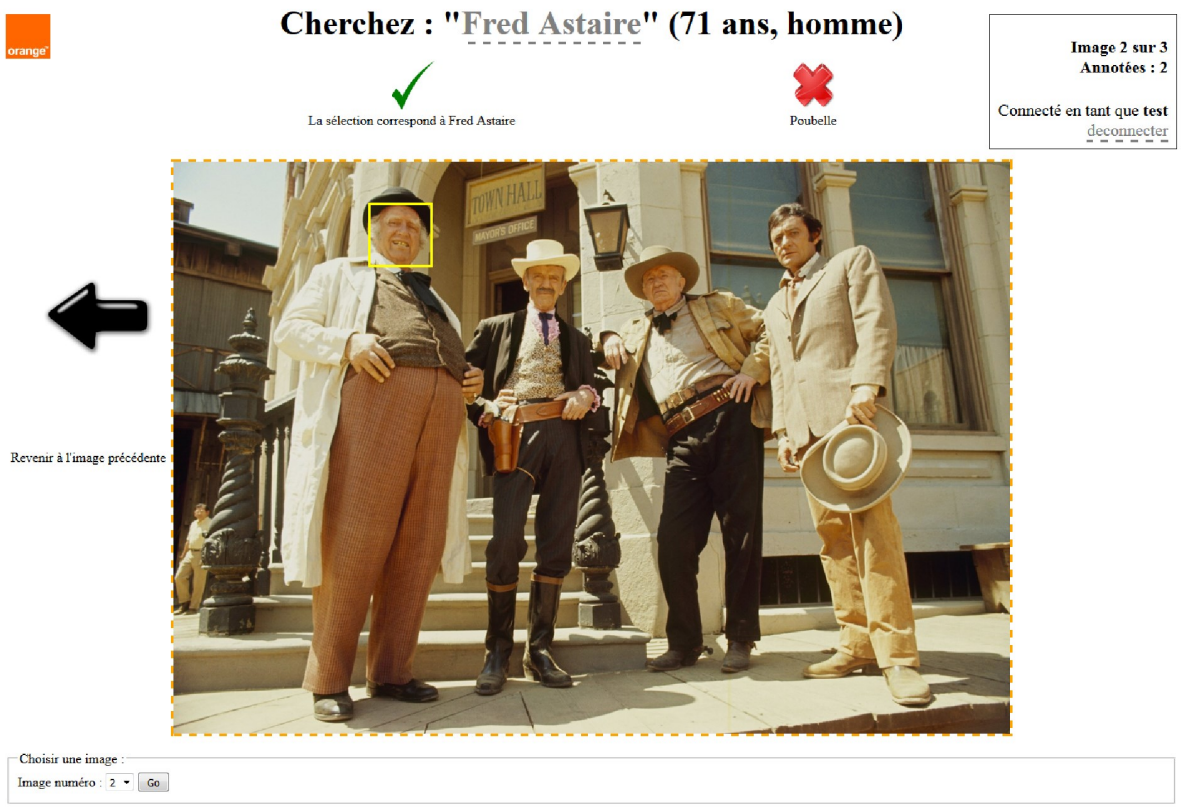



Figure 5.2.3 – (Better viewed in color). Screenshot of the web interface which has been developped for "cleaning" the annotations of the *IMDB-Wiki* dataset [RTVG16]. The name of the celebrity to look for in the photo as well as the corresponding annotations are indicated at the top of the screen, while the initial face proposal is denoted by the yellow rectangle. A user can either confirm the initial face proposal, or manually select another face among those presented in the photo.

input image and a corresponding annotation (the person identity, gender and age), a user has to simply select a face in the image to which the given annotation corresponds to. The screenshot of the web interface is presented in Figure 5.2.3. By crowdsourcing the annotation process via the described interface, we have managed to annotate 68,548 images (26 persons participated in the annotation campaign which lasted for 4 days).

Thus, in total, 250,367 images from the *IMDB-Wiki* dataset have been used in our experiments. In order to avoid ambiguity with the whole *IMDB-Wiki* dataset, below, we refer to this subset of 250,367 images of *IMDB-Wiki* as the *IMDB-Wiki_cleaned*.

| Dataset | Men | Women |
|---|---|---|
| *IMDB-Wiki_cleaned* | 56.9% | 43.1% |
| *PBGA* | 50.0% | 50.0% |
| *LFW* | 82.0% | 18.0% |
| *MORPH-II* | 84.7% | 15.3% |

Table 5.2.4 – Men / women ratio for all datasets which are used for training and/or evaluation of gender CNNs in the present chapter.

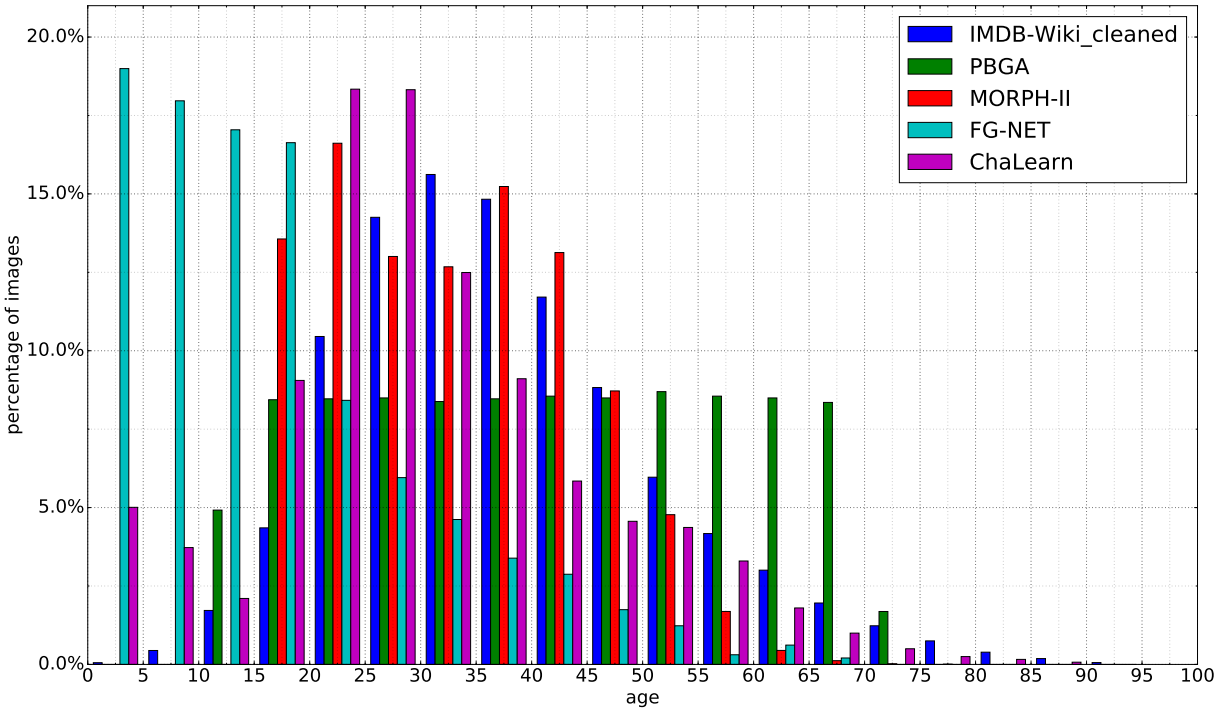



Figure 5.2.4 – (Better viewed in color). Histograms of the age distributions for all datasets which are used for training and/or evaluation of age CNNs in the present chapter. Each bin corresponds to an interval of five years.

**Test Dataset: Private Balanced Gender Age (PBGA)** The common problem of public benchmark datasets (like *LFW*, *MORPH-II* and *FG-NET* used in Section 5.3 for comparison with existing state-of-the-art approaches) is the fact that they are not well-balanced. For example, the ratio of men and women both in *LFW* and *MORPH-II* is about 80% to 20%. Similarly, about 50% of images in *FG-NET* belong to children while *MORPH-II* dataset contains almost 0 images of people over 60 and below 18 years old.

The performances measured on these benchmarks are prone to be biased. This is not critical for comparing the final best gender and age estimators with other state-of-the-art models (anyway, almost all gender recognition and age estimation studies evaluate their algorithms on one of the three listed benchmarks).

However, in this section, where our goal is to make important design and training choices for gender and age CNNs, we want to minimize the possible bias due to the evaluation dataset. To this end, we use a private internal dataset of non-celebrities. For each age in the interval between 12 years old and 70 years old, the dataset contains 30 images of men and 30 images of women. Thus, 3540 images in total. The resulting dataset perfectly suits the evaluation purposes as (1) it is ideally balanced between genders and ages, and (2) it is composed of non-celebrities which avoids all possible intersections with *IMDB-Wiki_cleaned* which is used for training. Below, we refer to this dataset as Private Balanced Gender Age or simply *PBGA* dataset.

In Table 5.2.4 and Figure 5.2.4, we compare the distributions of genders and ages in all datasets used in the present chapter highlighting that our *PBGA* dataset is the only one to be balanced. All results reported on the *PBGA* dataset in this section are calculated according to the cross-dataset protocol (*i.e.* without fine-tuning on *PBGA*).

#### 5.2.3.2 Experimental Results

**Target Age Encoding** Table 5.2.5 compares AE accuracies of *fast_CNN*s trained with different target age encodings presented in Paragraph 5.2.2.1. In Table 5.2.5 and further in this manuscript, age estimation CNNs are compared according to their MAEs (*cf.* the definition in Chapter 3).

| Age Encoding | Age Prediction Type | Age MAE |
|---|---|---|
| 0/1-CAE | ArgMax | 7.00 |
| | Expected Value | 6.42 |
| RVAE | N/A | 7.19 |
| LDAE | ArgMax | 6.58 |
| | **Expected Value** | **6.05** |

Table 5.2.5 – Comparison of target age encodings. Age estimation MAEs are reported on the *PBGA* dataset. Experiments are performed using the *fast_CNN* architecture.

For 0/1-CAE- and LDAE-based CNNs, we explore two possibilities to predict an age given 100 activations of the output layer. On the one hand, one can select the class (*i.e.* the age) which corresponds to the neuron with the highest activation — we denote this approach as "ArgMax" in Table 5.2.5. On the other hand, the age can be estimated as the expected value of all output activations: $age = \sum_{i=1}^{100} i * p_i$, where $p_i$ is the activation of the $i$-th output neuron (here, we assume that $\sum_{i=1}^{100} p_i = 1$).

The results in Table 5.2.5 have at least two conclusions. Firstly, we observe that age estimation by expected values significantly outperforms the one by "ArgMax" both for 0/1-CAE and for LDAE. This result conforms with the similar findings by Rothe et al. [RTVG16]. Secondly, the results demonstrate the general superiority of the CNN trained with LDAE over CNNs trained with 0/1-CAE and RVAE. Indeed, LDAE combines the strong points of the two other encodings: the similarity of the neighbouring ages (as in RVAE) and the robustness of age estimation (as in 0/1-CAE). Based on the obtained results, in the rest of the chapter, we use LDAE encoding and the expected value approach for age estimation.

| Face crop | Gender | | Age MAE |
|---|---|---|---|
| | CA | AUC | |
| "face-only" | 89.2% | 0.9777 | 6.54 |
| **"face+40%"** | **92.8%** | **0.9867** | **6.05** |

Table 5.2.6 – Comparison of "face-only" and "face+40%" crops. Experiments are performed using the *fast_CNN* architecture. The retina size of *fast_CNN* is set to 32x32 for "face-only" crop and to 64x64 for "face+40%" one. Results are reported on the *PBGA* dataset.

**Face Crop** The comparison of performances of the gender recognition and age estimation *fast_CNN*s which are trained with either "face-only" or "face+40%" face crops is presented in Table 5.2.6. As explained in Paragraph 5.2.2.2, in order to maintain the same resolution of the face details at the input of *fast_CNN*, the retina of the "face-only" *fast_CNN* is set to 32x32, while the one of the "face+40%" *fast_CNN* is set to 64x64. As in Chapter 4, CNNs are evaluated both according to their CAs and AUCs for the sake of more balanced evaluation between two classes (men and women).

Results in Table 5.2.6 leave no doubt on the choice of the face crops for gender and age CNNs. Indeed, *fast_CNN*s which are trained with "face+40%" crops significantly outperform the ones which are

trained with "face-only" crops both on gender recognition and age estimation problems. The performed experiments demonstrate that the top of the forehead and the chin (which are present in "face+40%" crops but not in "face-only" ones) provide some additional information for CNNs which is helpful for the studied problems (the ablation study in Paragraph 5.3.3.2 further confirm this result). At the same time, *fast_CNN*s are not confused by the background face-unrelated noise on the sides of "face+40%" crops which shows that CNNs learn to focus on the task-related information during the training.

Based on the obtained results, in all further experiments of the present chapter, we employ "face+40%" face crops both for gender recognition and age estimation CNNs.

**CNN Depth** Below, we compare four CNN architectures of different depths: *fast_CNN_n*, where $n \in \{2,4,6,8\}$ is the number of convolutional layers, for gender recognition and age estimation tasks.

| CNN | Gender | | Age MAE |
|---|---|---|---|
| | CA | AUC | |
| *fast_CNN_2* | 92.2% | 0.9833 | 6.65 |
| *fast_CNN_4* | 92.8% | **0.9867** | 6.05 |
| *fast_CNN_6* | **92.9%** | 0.9862 | 5.95 |
| *fast_CNN_8* | 92.3% | 0.9859 | **5.89** |

Table 5.2.7 – Impact of the CNN's depth on gender recognition and age estimation. Results are reported on the *PBGA* dataset.

The results presented in Table 5.2.7 highlight the difference between gender recognition and age estimation. Indeed, in case of gender recognition (columns 2-3 of Table 5.2.7), we observe that the best performances are already obtained with only four convolutional layers. Increasing the depth up to six layers has almost no impact on gender recognition results, while *fast_CNN_8* of eight convolutional layers performs even worse than shallower networks overfitting on the training dataset. At the same time, the column 4 of Table 5.2.7 clearly indicates a positive correlation between the depth of age CNNs and their performances. *fast_CNN_4* outperforms *fast_CNN_2* by almost 10% while *fast_CNN_6* and *fast_CNN_8* subsequently improve the age estimation by more than 1% each.

It is important to highlight that in this paragraph, our goal is *not* finding an optimal number of convolutional layers for *fast_CNN*, but evaluating the relationship between the CNN depth and the resulting gender recognition and age estimation performances. Thus, we observe that there exists a positive correlation between the CNN depth and the age estimation accuracy, while the gender recognition precision is not directly dependent on the number of convolutional layers.

More generally, the results in Table 5.2.7 illustrate that age estimation is a more complex and demanding problem than gender recognition. Indeed, the performed experiments show that contrary to age estimation, gender recognition training does not provide CNNs with the information which is discriminative enough to take the full advantage of the CNN's depth.

**Pretraining and Multi-Task Learning** Below, we firstly provide details on how we perform in practice the face recognition pretraining and multi-task learning for *fast_CNN*, and then we report the respective experimental results.

Training of a multi-task *fast_CNN* is very similar to the training of a mono-task one. In particularly, we also employ LDAE to encode age information. The only difference is that multi-task *fast_CNN* has 101 output neurons instead of 100 as 1 bit of gender information is concatenated to LDAE of size 100.

*fast-CNN* for face recognition is trained on a subset of the recent *Ms-Celeb-V1* dataset [Guo+16] (which was deliberately collected by its authors for face recognition purposes) containing the faces of 7,395 most popular persons (in terms of the number of images) from the original dataset. For simplicity, we approach the face recognition task as an identity classification problem with 7,395 classes. Therefore, the output layer of the face recognition *fast-CNN* contains 7,395 neurons with the softmax activation function. The CNN is optimized with the cross-entropy loss function in the similar way, as the gender recognition *fast_CNN*. The resulting model obtains a very decent accuracy of 92.1% when evaluated according to the *LFW* face verification protocol [LM+16] (for example, this accuracy is close to the one of the very popular OpenFace software [ALS16], which is further used in Chapter 6). Once face recognition pretraining is done, the last layer containing 7,395 neurons is substituted by a new (randomly initialized) layer for further fine-tuning for either gender recognition or age estimation, or for both tasks simultaneously.

Table 5.2.8 evaluates the impacts of the face recognition pretraining and multi-task learning on the performances of gender and age *fast_CNN*s. Thus, both face recognition pretraining and simultaneous learning for the two tasks increase the gender and age prediction accuracies with respect to the mono-task *fast_CNN* which is trained from scratch (lines (1, 3) and (1, 2) of Table 5.2.8, respectively).

| Training Type | Pretraining | Gender | | Age MAE |
|---|---|---|---|---|
| | | CA | AUC | |
| Mono-task | None | 92.8% | 0.9867 | 6.05 |
| Multi-task | None | 93.9% | 0.9891 | 5.96 |
| Mono-task | FR | **95.0%** | **0.9917** | **5.96** |
| Multi-task | FR | 94.5% | 0.9874 | 5.96 |

Table 5.2.8 – Effect of transfer learning (face recognition pretraining and multi-task learning) for gender recognition and age estimation CNNs. Results are reported on the *PBGA* dataset using *fast_CNN*. FR = Face Recognition.

The relative improvement of transfer learning on gender *fast_CNN* is more important than that on age *fast_CNN*. This perfectly makes sense as gender recognition training itself is not challenging enough to take the full advantage of deep CNNs (*cf.* the results of the CNN depth experiments). Hence, face recognition pretraining and multi-task learning work as regularizers during the gender recognition training making *fast_CNN* to learn richer and more expressive internal CNN representations. At the same time, age estimation is a more complicated problem than gender recognition which rather requires more sophisticated deep CNN architectures than an explicit help of transfer learning (though the latter also remains useful for age *fast_CNN* as shown in Table 5.2.8).

Moreover, while the two transfer learning approaches have exactly the same impact on age *fast_CNN* (MAE reduction from 6.05 to 5.96), face recognition pretraining is more effective than multi-task learning for gender *fast_CNN*. Indeed, as already mentioned above, gender recognition can be considered as a sub-problem of face recognition because gender is a part of a person's identity. Thus, the internal CNN representations of input faces which are learned during face recognition pretraining contain information

which can be directly used to predict gender.

Finally, the lines (3, 4) of Table 5.2.8 demonstrate that face recognition pretraining and multi-task learning for gender recognition and age estimation are not complimentary. Combining the two approaches together does not improve age MAEs and even leads to a slight decrease of gender CAs. This result indicates that the CNN regularization arising from the multi-task learning has already been obtained during the face recognition pretraining. So we can conclude that face recognition pretraining encompasses the positive effects of the multi-task learning for gender recognition and age estimation being a more general regularization approach.

### 5.2.4 Summary of the Optimal Design and Training Choices

The presented section has been devoted to the selection of optimal practices for design and training of gender recognition and age estimation CNNs. To this end, we have methodically identified and studied the following five parameters: (1) target age encoding strategies for age CNNs; (2) face crops which are given at the input of CNNs; (3) depth of the used CNN architectures; (4) usage of face recognition pretraining; and finally (5) possibility of multi-task learning for the two problems simultaneously.

The results of the section can be summarized as following:

1. LDAE should be employed as the age encoding strategy.
2. More noisy but wider "face+40%" face crops are better for gender and age prediction CNNs than less noisy but more narrow "face-only" face crops.
3. Age estimation is a more complex problem than gender recognition, and both gender recognition and age estimation trainings can be improved with the help of transfer learning.
4. Face recognition pretraining is particularly effective for gender recognition.
5. Face recognition pretraining encompasses multi-task learning meaning that the two transfer learning strategies should not be used together.

The stated conclusions are used in the following section for training our top performing state-of-the-art gender recognition and age estimation CNNs.

## 5.3 Top Performing CNNs for Gender and Age Prediction

In this section, we design the top performing gender and age prediction CNNs. The idea is to employ some of contemporary deep CNN architectures which have proven to be the most effective for other problems (such as *ImageNet* classification) and to train them for gender recognition and age estimation according to the conclusions of Section 5.2. In particular, we adopt two recent CNN architectures:*VGG-16* [SZ15] and *ResNet-50* [He+16] of 16 and 50 layers, respectively. *VGG-16* is a natural choice because the design of *fast_CNN*, which has been used in Section 5.2, is inspired from *VGG-16*, so this architecture can be considered as a very deep extension of *fast_CNN*. At the same time, *ResNet*s of different depths are currently one of the state-of-the-art CNN architectures. As shown in [CPC16], *ResNet-50* is a very good trade-off between the running time and the resulting performances.

More precisely, we adopt the following strategy for training both CNNs (*VGG-16* and *ResNet-50*) for gender recognition and age estimation:

1. Gender and age CNNs are firstly pretrained for face recognition.
2. Gender and age CNNs are trained separately (mono-task learning).
3. LDAE is used to encode ages for age estimation CNNs.

### 5.3.1 Design of the Top Performing CNNs

In order to design the best performing gender recognition and age estimation models, we employ *VGG-16* and *ResNet-50* which have been pretrained for face recognition. The two CNNs obtain respectively 97.2% and 99.3% of face verification accuracy when evaluated according to the the standard *LFW* protocol [2].

As already observed in Section 5.2, face recognition pretraining has a direct influence on gender recognition because the latter can be considered as a particular sub-problem of the former. This is further confirmed by the results in Table 5.3.1. Indeed, being more accurate for face recognition, *ResNet-50* also outperforms *VGG-16* for gender recognition by 1.6 CA points.

| CNN | Pretraining | Gender | | Age MAE |
|---|---|---|---|---|
| | | CA | AUC | |
| *VGG-16* | None | 93.7% | 0.9883 | 4.91 |
| *VGG-16* | GT | 96.8% | 0.9958 | 4.50 |
| *VGG-16* | FR | 97.1% | 0.9967 | **4.26** |
| *ResNet-50* | FR | **98.7%** | **0.9991** | 4.33 |

Table 5.3.1 – Deep CNNs for gender recognition and age estimation. Results are reported on the *PBGA* dataset. FR = Face Recognition. GT = General Task.

On the contrary, age estimation and face recognition are two independent problems, and while face recognition pretraining has a very important regularization role to facilitate age CNN training, the particular face recognition accuracy is not a decisive aspect for age estimation as in the case of gender recognition. Thus, as presented in Table 5.3.1, the age estimation accuracies of *ResNet-50* and *VGG-16* are almost the same. Actually, the fact that a much deeper *ResNet-50* does not improve *VGG-16* for age estimation reveals the limits of the *IMDB-Wiki_cleaned* dataset which is used for the training. Indeed, *ResNet-50* CNN model is so complex that it overfits on 250K of training images just after about 5 training epochs (while *VGG-16* does not overfit even after 50 full epochs). That said, we believe that *ResNet-50* would outperform *VGG-16* on age estimation if more training images with age annotations were available.

Summarizing, we select *ResNet-50* CNN as our best model for gender recognition, and *VGG-16* CNN as our best model for age estimation (the last two lines of Table 5.3.1).

**Importance of Face Recognition Pretraining** As a side remark, it is interesting to measure the particular impact of face recognition pretraining with respect to general task pretraining. Indeed, in Paragraph 5.2.2.4, we only intuitively motivate the choice of face recognition as a pretraining task. In order to quantitatively confirm this intuition, we have also trained *VGG-16* CNNs for gender recognition and age

2. The details of the *LFW* face verification protocol are provided here: http://vis-www.cs.umass.edu/lfw/.

estimation (1) from scratch, and (2) by fine-tuning from the *ImageNet* version of *VGG-16* [SZ15]. The resulting performances are presented in the lines 1 and 2 of Table 5.3.1.

As one may observe by comparing these lines, general task pretraining also improves the quality of gender/age prediction with respect to a *VGG-16* CNN which is trained from scratch. This is particularly visible for the gender CNN which has experienced an amelioration of CA from 93.7% up to 96.8% (confirming one of the conclusions of Subsection 5.2.1 about ineffectiveness of the CNN training from scratch for gender recognition). However, the lines 2 and 3 of Table 5.3.1 clearly indicate that face recognition pretraining is more effective than general task one for both studied problems.

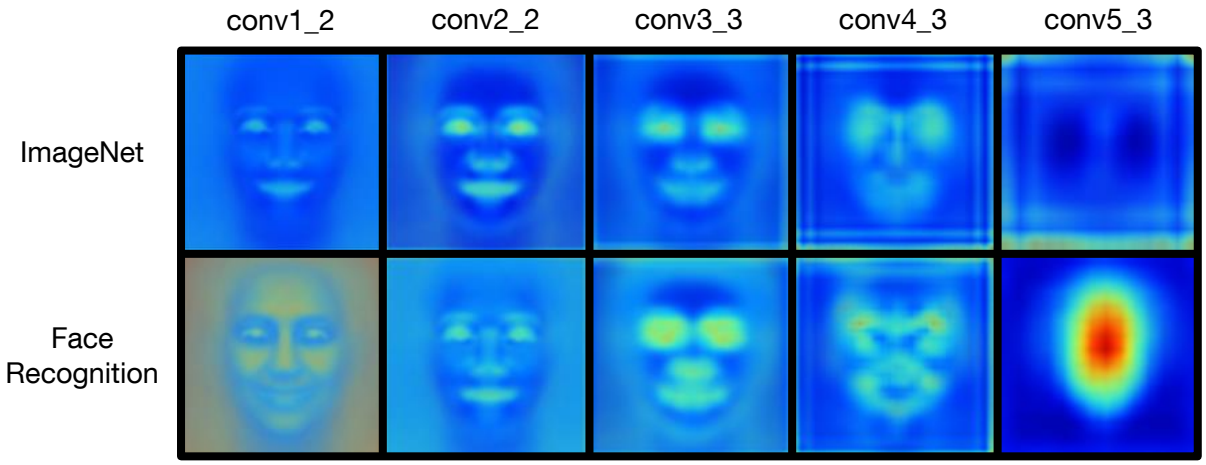



Figure 5.3.1 – (Better viewed in color). Heat maps of mean activations of convolutional layers in two *VGG-16* CNNs: the one trained for general task classification on *ImageNet* (top), and the one trained for face recognition (bottom).

Moreover, the advantage of face recognition pretraining can be also perceived qualitatively. To this end, we visualize the mean activations of the intermediate convolutional layers of general task and face recognition *VGG-16* CNNs when face crops are given at the inputs of the two networks in Figure 5.3.1. More precisely, *VGG-16* is composed of five blocks of 2-3 convolutional layers in each of them, and in Figure 5.3.1, we present the mean activations at the last convolutional layers of each of these block. In general, early convolutional layers of a deep CNN are activated by elementary parts of input images: like edges, corners etc. Thus, activations in the early layers *conv1_2* and *conv2_2* of the face recognition and general task *VGG-16* CNNs are similar, and they focus on the most salient face parts (*i.e.* eyes, mouth, and nose). Face recognition *VGG-16* further consolidates these activations in the deeper layers *conv3_3*, *conv4_3* and *conv5_3* targeting its attention on the face region. Therefore, the last convolutional layer of the face recognition CNN is a high-level face descriptor which can be potentially used for gender recognition and age estimation. On the contrary, the mean activations of the last *conv5_3* layer of the general task *VGG-16* are uniformly dispersed all over the map demonstrating that the general task *VGG-16* is not trained to focus on human faces (there are few human faces among *ImageNet* images). Thus, face recognition pretraining allows a CNN to better extract high-level information from face images than general task pretraining making the former more suitable for face-related problems such as gender and age prediction.

### 5.3.2 Benchmark Evaluation

In Subsection 5.3.1, we have designed the top performing deep CNNs: *ResNet-50* for gender recognition and *VGG-16* for age estimation. In this subsection we evaluate these two CNNs on three popular benchmark datasets: *LFW* (for gender recognition), *FG-NET* (for age estimation) and *MORPH-II* (for both tasks).

#### 5.3.2.1 Benchmark Datasets

Below, we present the benchmark datasets and the corresponding evaluation protocols. The distribution of genders and ages in all three used datasets is illustrated in Figure 5.2.4.

**LFW (gender recognition)** The Labelled Faces in the Wild (*LFW*) dataset which is today the standard benchmark for face and gender recognition systems has already been presented and used in Chapter 4. Below, we employ it for the comparison of our best gender recognition CNN with the state-of-the-art gender recognition models. Most of the recent studies reporting gender recognition results on *LFW* do not fine-tune their models on the target dataset. Therefore, we also follow this cross-dataset protocol for *LFW*.

**MORPH-II (gender recognition and age estimation)** The *MORPH-II* dataset [RJT06] is the biggest public dataset of non-celebrities with both gender and age annotations. The dataset which was collected by American law enforcement services contains more than 50K face images.

Guo et al. [GM10] proposed an evaluation protocol on *MORPH-II* which was later adopted by a large part of the research community. The protocol is the following: the *MORPH-II* dataset is split into three non-overlapping parts $S_1$, $S_2$ and $S_3$ with predefined proportions on gender and ethnicity distributions in each of the parts. Gender recognition and age estimation models are firstly trained on $S_1$ and tested on $S_2 \cup S_3$, and secondly trained on $S_2$ and tested on $S_1 \cup S_3$. Mean CA and MAE over these two experiments are reported as the final ones. We follow this protocol to evaluate both our best gender and age CNNs.

**FG-NET (age estimation)** *FG-NET*[3] is a tiny dataset containing 975 face images of 82 persons with age annotations. Despite its small size, *FG-NET* is still broadly used in age estimation research. The Leave One Person Out (LOPO) (*i.e.* 82-fold Cross-Validation) protocol has been widely adopted for evaluating age estimation models on *FG-NET*. We follow this protocol to compare our age CNN with the state-of-the-art.

#### 5.3.2.2 Evaluation Results

For convenience, Tables 5.3.2 and 5.3.3 regroup the scores of our best gender recognition *ResNet-50* and age estimation *VGG-16*, respectively comparing them with the state-of-the-art. For the sake of exhaustiveness, Table 5.3.2 also reports the score of our *optimized_CNN* from Section 4.3 of Chapter 4 on the *LFW* dataset.

3. Download link: www.cse.msu.edu/rgroups/biometrics/Publications/Databases/FGNETAgeEstimation.zip

| Reference | Year | Used Approach | CA | |
|---|---|---|---|---|
| | | | *LFW* | *MORPH-II* |
| [GM10] | 2010 | BIF + OLPP | N/A | 97.8% |
| [GM11] | 2011 | BIF + kPLS | N/A | 98.2% |
| [Sha12] | 2012 | LBP + SVM | 94.8% | N/A |
| [TP13] | 2013 | Multiscale LBP + SVM | 95.6% | N/A |
| [GM14] | 2014 | BIF + kCCA | N/A | 98.4% |
| [YLL14] | 2014 | Multi-scale CNN | N/A | 97.9% |
| [Yan+15a] | 2015 | CNN | N/A | 97.9% |
| [Han+15] | 2015 | BIF + hierarchical SVM | N/A | 97.6% |
| | | Human Estimators | N/A | 96.9% |
| [DBM15] | 2015 | FIS + SVM/RBF | 93.4% | N/A |
| [JC15] | 2015 | LBP + C-Pegagos | 96.9% | N/A |
| [MAP16] | 2016 | Local CNN | 94.5% | N/A |
| *This work (Chapter 4)* | *2016* | *Ensemble of optimized_CNNs* | *97.3%* | *N/A* |
| [CSLNRB16] | 2016 | LBP/HOG/CNN + SVM | 98.0% | N/A |
| [Moe+17] | 2017 | SLCDL + CRC | 96.4% | N/A |
| **This work** | **2017** | **ResNet-50 CNN** | **99.3%** | **99.4%** |

Table 5.3.2 – Comparison of our best gender recognition CNN with the state-of-the-art works on *LFW* and *MORPH-II* datasets.

| Reference | Year | Used Approach | MAE | |
|---|---|---|---|---|
| | | | *FG-NET* | *MORPH-II* |
| [Zho+05] | 2005 | Boosting + Regression | 7.48 | N/A |
| [GZSM07] | 2007 | AGES | 6.77 | 8.83 |
| [Guo+08] | 2008 | OLPP + regression | 5.07 | N/A |
| [Luu+09] | 2009 | AAM + SVR | 4.37 | N/A |
| [ZY10] | 2010 | MTWGP | 4.83 | 6.28 |
| [GM10] | 2010 | BIF + OLPP | N/A | 4.33 |
| [Luu+11] | 2011 | CAM + SVR | 4.12 | N/A |
| [CCH11] | 2011 | OHRANK | 4.85 | 5.69 |
| [GM11] | 2011 | BIF + kPLS | N/A | 4.18 |
| [GM14] | 2014 | BIF + kCCA | N/A | 3.92 |
| [YLL14] | 2014 | Multi-scale CNN | N/A | 3.63 |
| [Han+15] | 2015 | BIF + hierarchical SVM | 4.8 | 3.8 |
| | | Human Estimators | 4.7 | 6.3 |
| [WGK15] | 2015 | Unsupervised CNN | 4.11 | 3.81 |
| [Yan+15a] | 2015 | Ranking CNN | N/A | 3.48 |
| [LYK15] | 2015 | Hierarchical grouping and fusion | 2.81-3.55 | 2.97-3.63 |
| [Niu+16] | 2016 | Ordinal CNN | N/A | 3.27 |
| [RTVG16] | 2016 | ImageNet VGG-16 CNN + regression | 3.09 | 2.68* |
| [Liu+17] | 2017 | Group-aware CNN | 3.93 | 3.25 |
| **This work** | **2017** | **VGG-16 CNN + LDAE** | **2.84** | **2.99/2.35*** |

Table 5.3.3 – Comparison of our best age estimation CNN with the state-of-the-art works on *FG-NET* and *MORPH-II* datasets. (*) different protocol (80% of dataset for training, 20% for test).

The majority of the works from Tables 5.3.2 and 5.3.3 are discussed in Chapter 3, but for all reported results we provide short descriptions of the employed methods in the dedicated column. To the best

of our knowledge, the current best results for gender recognition were obtained by [CSLNRB16] and [GM14] on *LFW* and *MORPH-II*, respectively. Castrillon et al. [CSLNRB16] combined hand-crafted features (LBP and HOG) with the features from a compact CNN (comparable by size to *fast_CNN*) and used an SVM classifier above. Guo et al. [GM14] used BIF features (which are somewhat similar to the features from early layers of deep CNNs) and a kernel-based Canonical Correlation Analysis for simultaneous estimation of gender and age.

We improve the results of these two works from 98.0% to 99.3% and from 98.4% to 99.4%, respectively. For both datasets, the improvements are statistically significant with $p < 0.01$ according to the proportions test. We believe that the key reason for the success of our model is the usage of face recognition as pretraining which has allowed us to effectively train a much deeper CNN than those which were employed by previous CNN-based approaches for gender recognition.


Figure 5.3.2 – (Better viewed in color). Examples of gender recognition (on *LFW*) and of age estimation (on *MORPH-II*) by our best models. Both successful and failed cases are presented. For GR, the maximum softmax activation is provided.

The state-of-the-art age estimation models were described in the recent works [LYK15], [Liu+17] and [RTVG16]. The study from [LYK15] is extremely interesting. Indeed, despite the fact that the authors employed a fusion of very basic hand-crafted features with a standard SVR, they managed to obtain excellent age estimation results by a meticulous selection of a hierarchical structure of their model (*i.e.* by firstly predicting an age group and then estimating the precise age inside the group) and by proposing several feature fusion algorithms. However, the choice of an optimal combination of features to fuse depends on the dataset, therefore it is difficult to evaluate the real age estimation scores from their work (thus, in Table 5.3.3, we provide intervals from their paper rather than a single score). Liu et al. [Liu+17] used a hierarchical age grouping to train an age CNN reporting the currently best score on *MORPH-II* following the well-established protocol from [GM10]. Rothe at al. [RTVG16] did not follow this protocol so their score on *MORPH-II* cannot be compared to others in Table 5.3.3 (for the sake of fair comparison, we evaluate our age CNN both according to the protocols from [GM10] and [RTVG16]). Rothe et al. [RTVG16] also obtained the best MAE of 3.09 on the *FG-NET* dataset. The approach of Rothe et al. is very similar to ours: the same *VGG-16* CNN architecture and *IMDB-Wiki* training dataset. However, the principal difference between our solutions is the fact that we use LDAE instead of RVAE

encoding and the face recognition pretraining instead of the general task pretraining. The results in Table 5.3.3 confirm the validity of the training choices made in Section 5.2.

Finally, some successful and failed examples of gender recognition and age estimation by our best CNNs on the benchmark datasets are presented in Figure 5.3.2. As one can observe, the failed cases correspond to either peculiar facial expressions (the top line) or unusual aging patterns (the bottom line).

### 5.3.3 Qualitative Analysis

Subsection 5.3.2 shows that our *ResNet-50* and *VGG-16* CNNs obtain the state-of-the-art performances on the most popular gender and age benchmarks, respectively. Below, we additionally perform several qualitative assessments of the designed models, namely: evaluation of their sensitivities to the resolutions and occlusions in input face images, and comparison of our age CNN with human participants on a popular French TV show.

#### 5.3.3.1 Face Crop Resolution

It is intuitively obvious that the higher is the resolution of an input face crop the easier it is to correctly estimate the corresponding gender and age. But to what extent the accuracies of our best gender and age CNNs depend on the input resolution?

In order to answer to this question, it is firstly important to separate the notions of the size of the face crops which are given at the input of CNNs, and the resolution of these crops. Indeed, the former is defined uniquely by the retina of the used CNN model: for example, *fast_CNN* which is used in Section 5.2 requires face crops of size 64x64, while our best *ResNet-50* and *VGG-16* CNN require face crops of size 224x224. At the same time, the crop resolution depends both on the initial images from which the face crops are extracted, and on the model retina. More precisely, if the target retina size is bigger (smaller) than the size of the face crop region in the original photo, then this region is downsampled (upsampled) to the target retina size using bilinear interpolation. Therefore, the maximum possible resolution of face crops is limited by the size of the model retina meaning that in case of our best CNNs, it is of 224x224 pixels.


Figure 5.3.3 – (Better viewed in color). Example of face crops of the same size (224x224), but of different resolutions varying from 224x224 down to 16x16. In order to upscale a lower resolution face crop to 224x224, the nearest-neighbour interpolation is used.

The maximum resolution of a face crop $n_1$x$n_1$ can be easily reduced to $n_2$x$n_2$ ($n_1 > n_2$) by firstly downsampling it to $n_2$x$n_2$, and then uspamling it back to $n_1$x$n_1$ using nearest-neighbour interpolation (in Subsection 4.2.4 of Chapter 4, we refer to this operation as "Pixelization PPF"). Thus, in order to measure the sensitivity of the designed *ResNet-50* and *VGG-16* CNNs to the maximum face crop resolution, we

vary the latter from 224x224 to 16x16, and report the respective gender and age scores on the *PBGA* dataset. Examples of face crops of the same sizes (224x224, which is the retina size of *ResNet-50* and *VGG-16*) but with varying resolutions are illustrated in Figure 5.3.3.

| Maximum Resolution (pixels) | Gender | | Age MAE |
|---|---|---|---|
| | CA | AUC | |
| 224x224 | 98.7% | 0.9991 | 4.26 |
| 196x196 | 98.7% | 0.9990 | 4.31 |
| 128x128 | 98.6% | 0.9987 | 4.48 |
| 96x96 | 98.6% | 0.9988 | 4.46 |
| 64x64 | 98.5% | 0.9977 | 4.79 |
| 32x32 | 94.0% | 0.9858 | 8.06 |
| 16x16 | 50.9% | 0.6532 | 13.70 |

Table 5.3.4 – Sensitivity of our best performing gender and age CNNs to face crop resolution. The maximum resolution of face crops is varied from 224x224 pixels down to 16x16 pixels. Results are reported on the *PBGA* dataset.

The results of the performed resolution experiment are summarized in Table 5.3.4. As one can observe, the dependency of the CNN accuracies on the maximum resolution of face crops is highly non-linear both for gender recognition and for age estimation. The degradations of performances of the two networks are marginal even when the maximum resolution is reduced down to 96x96 (as a side remark, it is interesting to notice that the same observation was made by Schroff et al. [SKP15] in the context of the face recognition problem). The further reduction of the resolution down to 64x64 pixels results in more significant losses of precision (especially, in case of age estimation), but the scores remain comparable to the original ones (*i.e.* those with the maximum resolution of 224x224). Finally, there is a clear gap between the CNN performances on face crops with resolutions of 32x32 and 64x64 pixels. The obtained results informally confirm the choice of 64x64 as a retina size for *fast_CNN* in the experiments of Section 5.2.

#### 5.3.3.2 Sensitivity to Occlusions

In this paragraph, we perform a simple ablation analysis to estimate the relative importance of various regions of human faces for our gender recognition and age estimation CNNs. The idea is to mask these regions in face crops and to evaluate the resulting impacts on gender recognition and age estimation accuracies. The amount of impact on performances indicates the importance of each tested region for the respective tasks.

We uniformly split input images into 63 regions as shown in Figure 5.3.4 (top): 49 tiny square regions to evaluate the impact of small local parts of the face, 7 horizontal stripes and 7 vertical stripes to evaluate the respective symmetries. It is important to highlight that the described test regions are selected uniformly in face crops and not in connection with certain face landmarks and/or semantic face parts. This way, we do not introduce any human a priori on which face regions should be tested and which ones should not. For each of 63 tested regions, we blur the corresponding part in all face crops of the *PBGA* dataset using a Gaussian filter with $\sigma = 7$ which makes the considered part completely concealed. After processing the blurred images with our best gender and age CNNs, we evaluate the performance losses

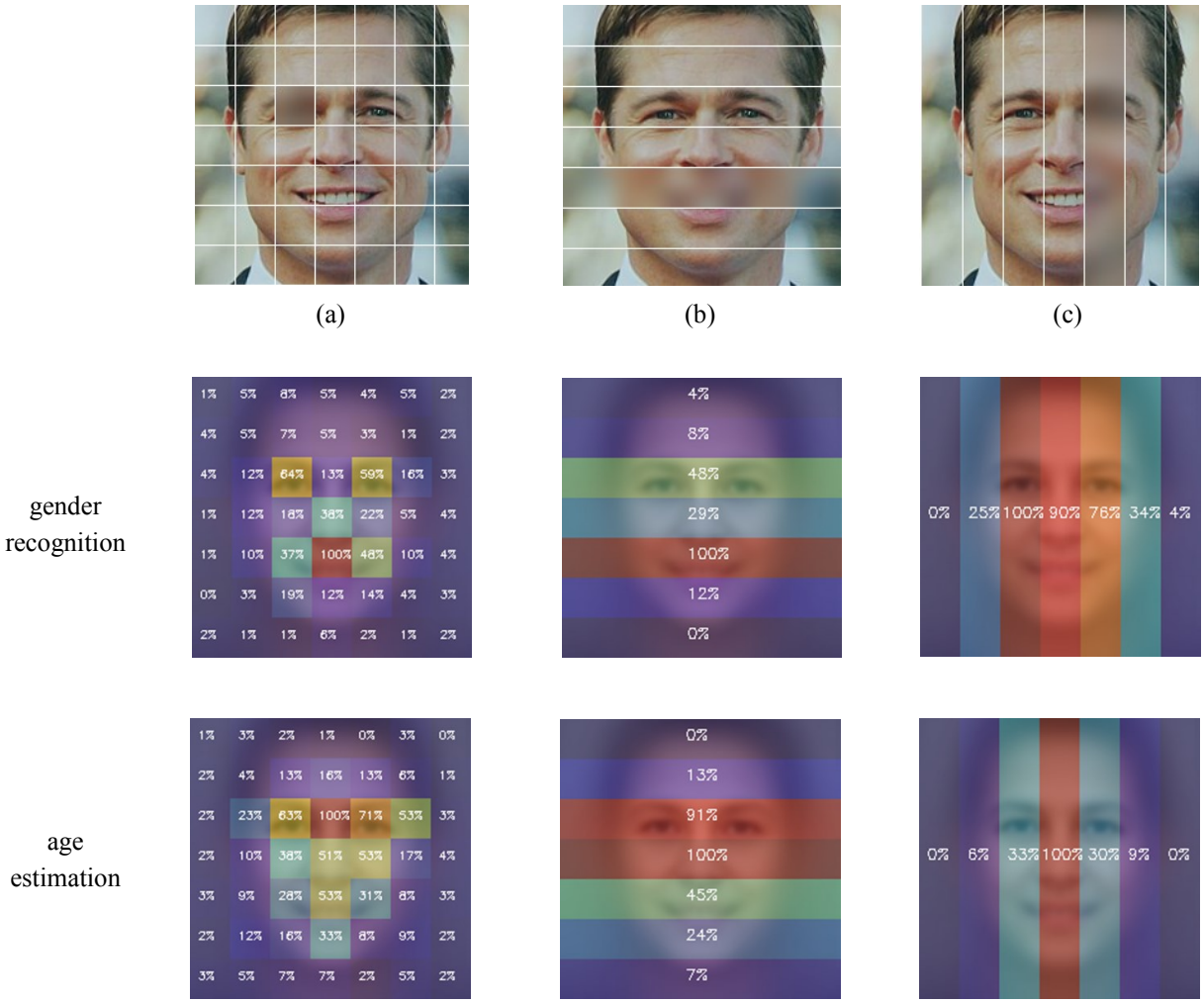



Figure 5.3.4 – (Better viewed in color). (Top) examples of the used occlusions: (a) 1 of 49 square areas of 32x32 pixels, (b) 1 of 7 horizontal stripes of the height of 32 pixels, and (c) 1 of 7 vertical stripes of the width of 32 pixels. White lines are presented only for the sake of illustration and are not the part of the occlusions. (Middle) sensitivity of our gender recognition *ResNet-50* to the occlusions. (Bottom) sensitivity of our age estimation *VGG-16* to the occlusions. Percentages and heat maps indicate the relative losses in performances after blurring the corresponding image parts.

with respect to the original face crops. The percentages of degradations in CAs (for gender recognition) and in MAEs (for age estimation) which are caused by blurring are written in white in Figure 5.3.4 (middle and bottom) and are also illustrated with the help of heat maps. For convenience, we provide the mean face image on the background of the heat maps in Figure 5.3.4 (middle and bottom).

If we analyse the heat maps of small square regions (the 1st column of Figure 5.3.4 (middle and bottom)), we observe that globally, both networks are sensitive to the salient regions of the face: eyes, eyebrows, nose and mouth. It makes sense because as it is illustrated in Figure 5.3.1, the salient face regions produce the majority of activations in the first layers of CNNs. Moreover, the gender CNN is more sensitive to the center of the mouth and to the periocular region conforming with previous studies [Hu+11; JX+16], while the age CNN more equally depends on all salient face parts. Finally, Figure 5.3.4 also demonstrates that the two CNNs quite precisely follow the horizontal symmetry of faces.

#### 5.3.3.3 "Guess My Age" TV Show [4]

Being humans, we rarely experience difficulties in gender recognition from face images. At the same time, precise age estimation can be very challenging even for human perception. Thus, the French TV channel "C8" launched a TV show named "Guess My Age: Saurez-vous devinez mon âge ?" which has rapidly gained popularity among spectators [5]. The TV show is a game where a pair of participants can win money depending on how well they estimate ages of strangers.

"Guess My Age" offers a good opportunity to compare our best *VGG-16* CNN versus humans on age estimation task. Indeed, contrary to human estimators taking part in a crowdourcing campaign, the participants of the TV show have a strong financial motivation to do their best in order to correctly predict age. Moreover, when the game participants estimate ages, they see the full bodies of strangers which allows rectifying their decisions based on the strangers' clothes and accessories (the strangers are explicitly asked to be dressed in the same way as they are in everyday life). Due to the fact, that our age estimation CNN takes only face crops at its input, the game participants are a priori in more advantageous conditions. Therefore, the age estimation results of the TV game participants represent a very challenging baseline for our age model.

Figure 5.3.5 – (Better viewed in color). Examples of screenshots from the TV show "Guess My Age" which have been used for face crop extraction and subsequent age evaluation.

In order to perform the most objective comparison, we have downloaded 12 broadcasts of the first season, and from each show, we have manually extracted screenshots containing strangers whose ages should be estimated. In total, we have collected photos of 61 different persons aged between 19 and 82 years old, and for each person, 5 screenshots have been taken (examples of screenshots are provided in Figure 5.3.5). The resulting face crops have been processed by our best *VGG-16* CNN, and the age estimations have been averaged among 5 screenshots.

Table 5.3.5 compares the resulting accuracies of the age CNN with those of humans according to two metrics, namely: MAE and the number of better estimations. The latter represents the number of times our model better predicts an age of a certain person than the TV game participants, and vice-versa. Despite the fact that the game participants have been in more favourable conditions (*cf.* explanations above), our model estimates the age more accurately according to both metrics.

4. The experiments presented in this paragraph have been performed by Alexandre Fradet, a master student at Eurecom, in the frame of his semester project (autumn 2016). The project was supervised by Grigory Antipov and Jean-Luc Dugelay.

5. The show was elected as the "best TV game of the season" by TV Notes 2017.

| | Human Participants | *VGG-16* CNN |
|---|---|---|
| MAE | 5.3 | **4.8** |
| # of better estimations | 24 | **37** |

Table 5.3.5 – Comparison of age estimation by humans participants of a TV show "Guess My Age" and by our best *VGG-16* CNN using on some screenshots from the show. "# of better estimations" is the number of times our model (the human participants) better predicts an age of a certain person than the human participants (our model).

The obtained results clearly demonstrate that our *VGG-16* CNN is at least as accurate as the participants of the TV show. However, the small number of available test images does not allow us to conclude with enough statistical significance that our model is *better* than an average human participant. Indeed, a p-value of a two-sided Wilcoxon signed rank test with a null hypothesis that $(x-y)$ (where $x$ is a vector of 61 absolute human age estimation errors, and $y$ is the similar list for our model) comes from a distribution with a zero median is only 0.2. In future, we plan to collect more screenshots from the second season of the TV game (which just ended) in order to perform a more complete study.

### 5.3.4 Top Performing CNNs: Summary

In the presented section, we have obtained the central results of this chapter: the state-of-the-art *ResNet-50* CNN for gender recognition and *VGG-16* CNN for age estimation. The superiority of the performances by the designed models over the ones reported in previous studies have been shown on three most popular benchmark datasets.

Moreover, the qualitative analysis of the obtained gender and age CNNs has allowed to experimentally confirm a number of intuitive conjectures which have been made before in the manuscript. Thus, the comparison of two *VGG-16* CNNs, which have been pretrained for *ImageNet* classification and face recognition problems, have demonstrated the advantages of the latter for the face related problems confirming the intuitive motivation for selecting face recognition as a pretraining task in Section 5.2. The experiments in Paragraph 5.3.3.1 have validated the choice of the retina size (64x64) for our *fast_CNN* architecture from Section 5.2.

Finally, the evaluation of our best age CNN on the screenshots from "Guess My Age" TV show has allowed to compare the age estimation by the designed model and by human vision. Thus, the obtained results have confidently shown that our model performs at least as good as highly motivated human estimators.

## 5.4 ChaLearn Competition on Apparent Age Estimation

In Section 5.3, we have designed the best performing age estimation CNN according to the most popular public benchmarks. The reported age estimation results concern biological age estimation.

At the same time, apparent age estimation (*cf.* Chapter 3 for definitions of biological and apparent ages) has recently attracted a lot of attention due to to the first public Apparent Age Estimation Competition (AAEC) organized by ChaLearn in 2015 [Esc+15]. The organizers collected a dataset of face images and developed a web interface where people could annotate these images with apparent ages.

More than 100 teams participated in the competition and the five best approaches were based on deep CNNs [Esc+15].

The interest to the first edition of the AAEC was so high, that in 2016, ChaLearn decided to organize its second edition [Esc+16]. Being motivated to verify whether our conclusions on the optimal design and training strategy of age CNNs from Section 5.2 (notably, the usage of face recognition pretraining and LDAE age encoding) hold for apparent age estimation as well, we participated in this second edition by adapting our best biological age estimation *VGG-16* CNN for apparent age. As a result, we won the first place of the second edition of ChaLearn AAEC significantly outperforming other participants.

In this section, we describe the competition and detail our winning solution. In particular, the section is organized as following: we firstly present the competition organization and protocol in Subsection 5.4.1, then we outline our winning solution highlighting the importance of its subparts in Subsections 5.4.2 and 5.4.3, and finally we report and analyse the competition results in Subsection 5.4.4.

### 5.4.1 AAEC Protocol

With the help of a Facebook game-like application and Amazon Mechanical Turk [6] crowdsourcing campaign, the competition organizers collected the biggest public dataset annotated with apparent ages. We will further refer to it as *ChaLearn* dataset. It contains 7591 images (4113 images for training, 1500 for validation and 1978 for test). The apparent age annotations for training and validation images have been made public, while only the organizers have an access to the annotations of the test images.

Each image of the competition dataset is annotated with a mean age $\mu$ and a corresponding standard deviation $\sigma$ (these statistics are calculated based on at least 10 human votes per image). The metric which was chosen by the competition organizers to evaluate apparent age estimation models is quite different from MAE which is used for biological age estimation. The competition metric $\varepsilon$ is defined as the size of the tail of the normal distribution with the mean $\mu$ and the standard deviation $\sigma$ with respect to the predicted age $\hat{x}$:

$$\varepsilon = 1 - e^{-\frac{(\hat{x}-\mu)^2}{2\sigma^2}} \tag{5.4.1}$$

Therefore, the apparent age estimation errors on examples with a small standard deviation (*i.e.* on examples on which human votes are close to each other) are penalized stronger than the same errors on examples with a high standard deviation (*i.e.* on examples on which human votes disagree).

The competition itself is organised in two stages. During the first development stage (which lasts for about three months), the participants have access to training images with annotations and validation images without annotations. The participants can train their models using the training images and evaluate the respective performances on the validation ones via a dedicated web interface (there are no limit on the number of evaluations). It is important to mention, that the organizers strongly encourage the usage of external training data.

During the second (evaluation) stage, the organizers publish the apparent age annotations for the validation images as well as the test images which must be used the final evaluation. The second stage lasts for only a couple of days, and the competitors are required to upload their age estimations of the test images to the competition website (only one attempt is authorized) as well as the source codes of the

6. https://www.mturk.com/mturk/welcome

developped solutions. The official results are announced in about a week after the end of the competition (as soon as the organizers verify the validity of the provided source codes).

### 5.4.2 Proposed Solution

As outlined above, our final solution at the ChaLearn AAEC is largely based on our best performing *VGG-16* CNN from Section 5.3 which is designed for biological age estimation. Basically, the key idea is to fine-tune our biological age CNN for apparent age estimation using the competition training data. A similar approach was adopted by the winners of the first edition of the AAEC [RTVG16]. Nevertheless, our solution has one important particularity: we design a separate CNN for apparent age estimation of children. Below, we detail our approach motivating the key design choices.

#### 5.4.2.1 Face Crop Extraction

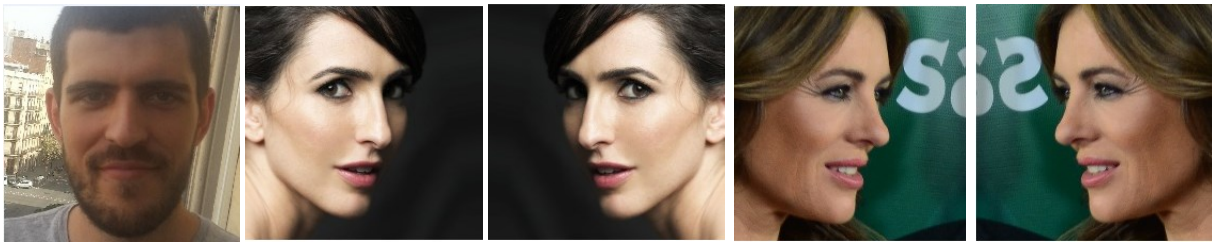
ChaLearn AAEC is an "end-to-end" contest meaning that given at input raw real-life images (which are mostly extracted from social networks), the participants have to output corresponding apparent age estimations. Image preprocessing is considered as a part of the challenge, and the participants are free to apply any algorithms with the condition to provide the respective sources afterwards.

As for biological age estimation, we have performed face crop extraction prior to the CNN processing, which consists in face detection, face landmark extraction and face alignment (*cf.* Paragraph 5.2.2.2). However, unlike all other experiments in this manuscript (which use face crops extracted with private face detection and face landmark detection solutions), for the AAEC, we have employed open-source algorithms which are presented below (this way, our final solution is completely reproducible, as required by the competition rules).

**Face detection** We have used the open source "Head Hunter" face detector [Mat+14]. In particularly, we have employed the fast implementation by [PVZ15]. In order to detect faces regardless of the image orientation, we rotate each input image at all angles in the range [-90°, 90°] with the step of 5°. We then select the rotated version of the input image which gives the strongest output of the face detector for the face alignment step. If no face is detected in all rotated versions of the input image, the initial image is upscaled and the presented algorithm is repeated until a face is detected. 2 upscaling operations have been enough to detect at least one face in all images of the competition test dataset.

**Face alignment** We have chosen a popular face landmark extraction solution which is proposed in [Uri+15]. The solution of [Uri+15] is based on a multi-view facial landmark extraction approach. There are five landmark extraction models: a frontal model, two profile models and two half-profile models. Each of these models is tuned to work on one of the corresponding facial poses (*cf.* Figure 5.4.1).

The face alignment follows the face detection and requires running of all 5 landmark models on the detected face. Each model reports a confidence score which shows how well the corresponding landmarks are detected in the given face. We then select the model with the highest confidence score and perform an affine transformation from the detected landmarks to the predefined optimal positions of these landmarks with respect to the detected facial pose.



(a) (b) (c) (d) (e)

Figure 5.4.1 – Examples of facial poses: (a) frontal, (b) -half-profile, (c) half-profile, (d) -profile, (e) profile.

#### 5.4.2.2 Separate Age Estimation CNN for Children

If we compare the age distributions of the *IMDB-Wiki_cleaned* dataset, which has been used for training of our biological age *VGG-16* CNN, and of the *ChaLearn* dataset, which is used in the competition, we will immediately notice that the two have very different proportions of children images. Indeed, less than 1% of *IMDB-Wiki_cleaned* images belong to the age category 0-12, while the competition dataset contain about 10% of such images (*cf.* Figure 5.2.4).

Moreover, we have noticed that according to the competition dataset annotations, the average standard deviation of human votes for images of children (between 0 and 12 years old) is about 1, while the average standard deviation for all other images is about 5 (*cf.* Figure 5.4.2). In other words, according to the competition data, humans estimate an age of a child almost 5 times more precisely than an age of an adult. At the same time, as outlined in Subsection 5.4.1, the competition metric $\varepsilon$ (*cf.* Equation 5.4.1) is defined in a way that the errors on the test images for which the standard deviation $\sigma$ is low (*i.e.* for which various human estimations agree between each other) are penalized the most.

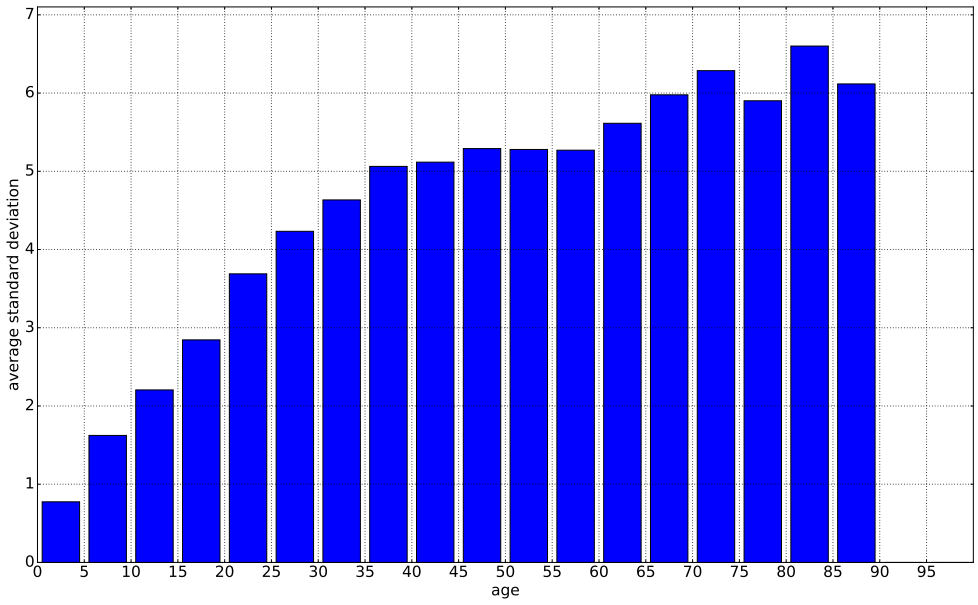



Figure 5.4.2 – Histogram of standard deviations of apprarent age estimation depending on the age categories. Calculated based on human annotations of the *ChaLearn* dataset. Each bin corresponds to an interval of five years.

The observations above lead to a key issue which needs to be addressed. Indeed, on the one hand, precise age estimation of children images is vital for the ChaLearn AAEC, and on the other hand, due to the lack of the training data, our biological *VGG-16* CNN is prone to underperform on children images. In order to resolve this problem, we have trained a separate "children" *VGG-16* CNN which is dedicated exclusively to age estimation of children between 0 and 12 years old in addition to a "general" age *VGG-16* CNN (trained in Section 5.3).

To this end, we have collected a private dataset of about 6K children images in the 0-12 age category (biological ages) using the Internet search engines. In particular, we have used the following 2 approaches for the data collection:

1. Queries in the search engines of type "1 year old baby" or "girl 7 years old" in different languages. This approach requires a manual verification of the search engine results.
2. Usage of own private class photos (from the primary and secondary schools). These photos are very useful, because each photo usually contain frontal faces of 20-30 children of the same age.

Once the children dataset has been collected, we have trained the "children" *VGG-16* CNN by fine-tuning from the "general" network on the images of children. Unlike the "general" *VGG-16*, the last layer of the "children" CNN contains 13 neurons as the network outputs only ages between 0 and 12 years old.

#### 5.4.2.3 Fine-tuning for Apparent Age Estimation

Having two CNNs (the "general" and the "children" ones) for biological age estimation, our next step is fine-tuning them for apparent age estimation on the competition *ChaLearn* dataset. The two networks are fine-tuned separately: the "general" *VGG-16* is fine-tuned on all available *ChaLearn* images, while the "children" one only on those images for which (apparent) age annotations are between 0 and 12 years old. Moreover, following the winners of the first edition of the ChaLearn AAEC [RTVG16], we have fine-tuned several instances of "general" and "children" CNNs (*cf.* Figure 5.4.3).

In particular, in case of the "general" CNN, we have combined all training and validation images from the *ChaLearn* (5613 images in total) and fine-tuned 11 "general" CNNs for apparent age estimation using 11-fold cross-validation where the size of each of 11 training datasets is 5113 images and the size of each of 11 non-overlapping validation datasets is 500 images. In case of the "children" CNNs, we have combined all children images from the training and validation parts of the competition dataset (there are 543 of them). Due to the small number of available images, we fine-tune the "children" CNNs for apparent age estimation without any validation saving the CNN weights at 3 predefined points which have been chosen by experimenting on the validation dataset. As a result, we obtain 3 "children" CNNs for apparent age estimation.

Due to the relatively small size of the *ChaLearn* dataset, we have used "5-times data augmentation" when fine-tuning "general" and "children" CNNs for apparent age estimation. Apart from the original images, we have used their mirrored versions, randomly rotated versions (the absolute rotation angle is no more than 5°), randomly shifted versions (the absolute shift length is no more than 5% of the image size) and randomly scaled versions (the scaled size is between 95% and 105% of the original size).

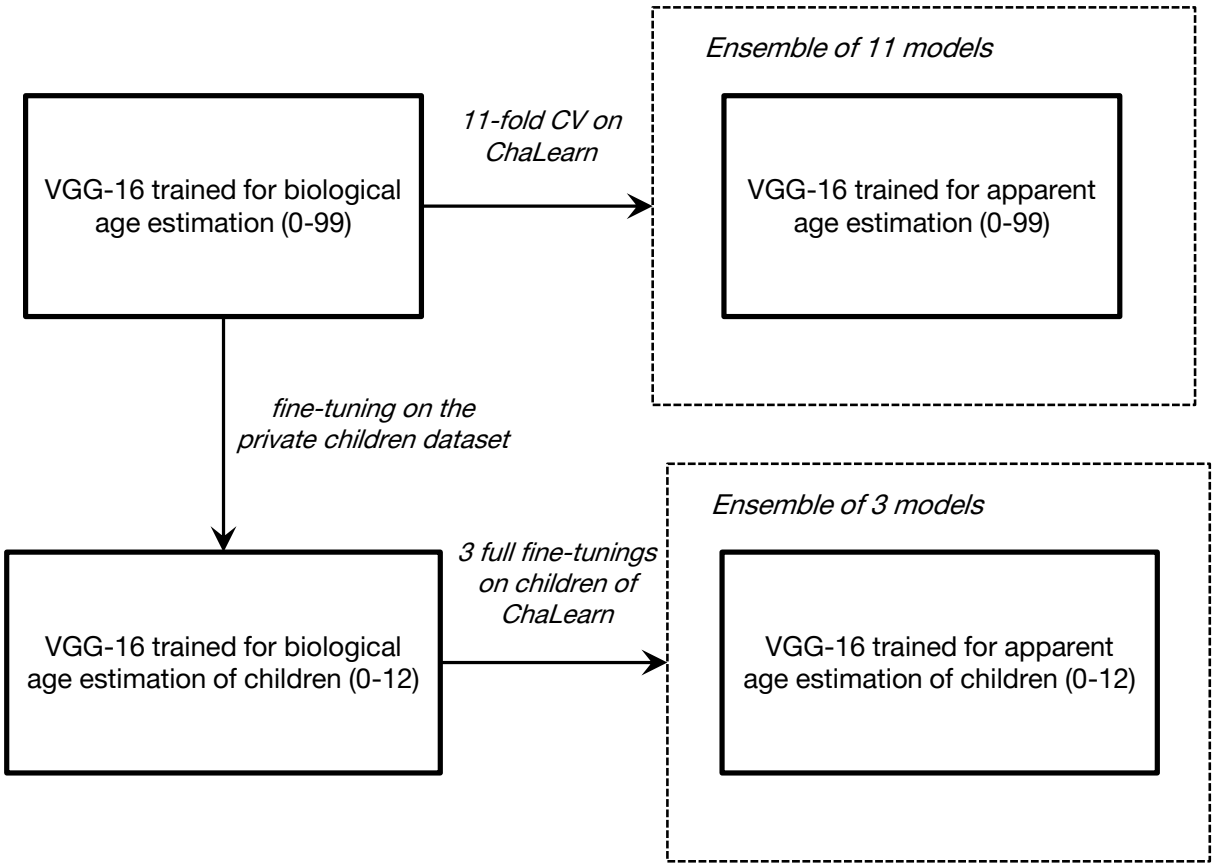



Figure 5.4.3 – Pipeline of fine-tuning of 11 "general" and 3 "children" *VGG-16* CNNs for ChaLearn AAEC.

#### 5.4.2.4 Full Test Pipeline

Summarizing, the whole pipeline of our final solution at test stage is presented in Figure 5.4.4. A test image is firstly processed by a face detector which defines a face bounding box and rotates the image accordingly. Then the detected face is aligned and the resulting image is resized to 224x224 pixels. From the obtained image, we generate its 7 modified versions: the mirrored one, the ones rotated at ±5°, the ones shifted by 5% on the left/right and the ones scaled in/out by 5%. This is done in order to compensate a negative impact from minor face alignment errors (which are inevitable given the difficulty of the competition dataset). In total, there are 8 images including the original one.

All these images are processed by 11 "general" CNNs. We take the values of 100 output neurons after each of 88(= $8 \times 11$) CNN forward passes, average them and normalize them to sum up to 1. Thus, we obtain an averaged vector $p$ of 100 values representing probabilities of belonging to ages between 0 and 99 years old. The final "general" age prediction is calculated as an expected value of these probabilities: $general\_age = \sum_{i=0}^{99} i * p_i$. If the predicted "general" age is superior to 12, it is considered as the final apparent age estimation and the algorithm stops. In the opposite case, we process the same 8 images as before by 3 "children" CNNs. We take the values of 13 output neurons after each of 24(= $8 \times 3$) CNN forward passes, average them and normalize them to sum up to 1. Thus, we obtain a vector $p$ of 13 values representing probabilities of belonging to ages between 0 and 12 years old. The final "children"

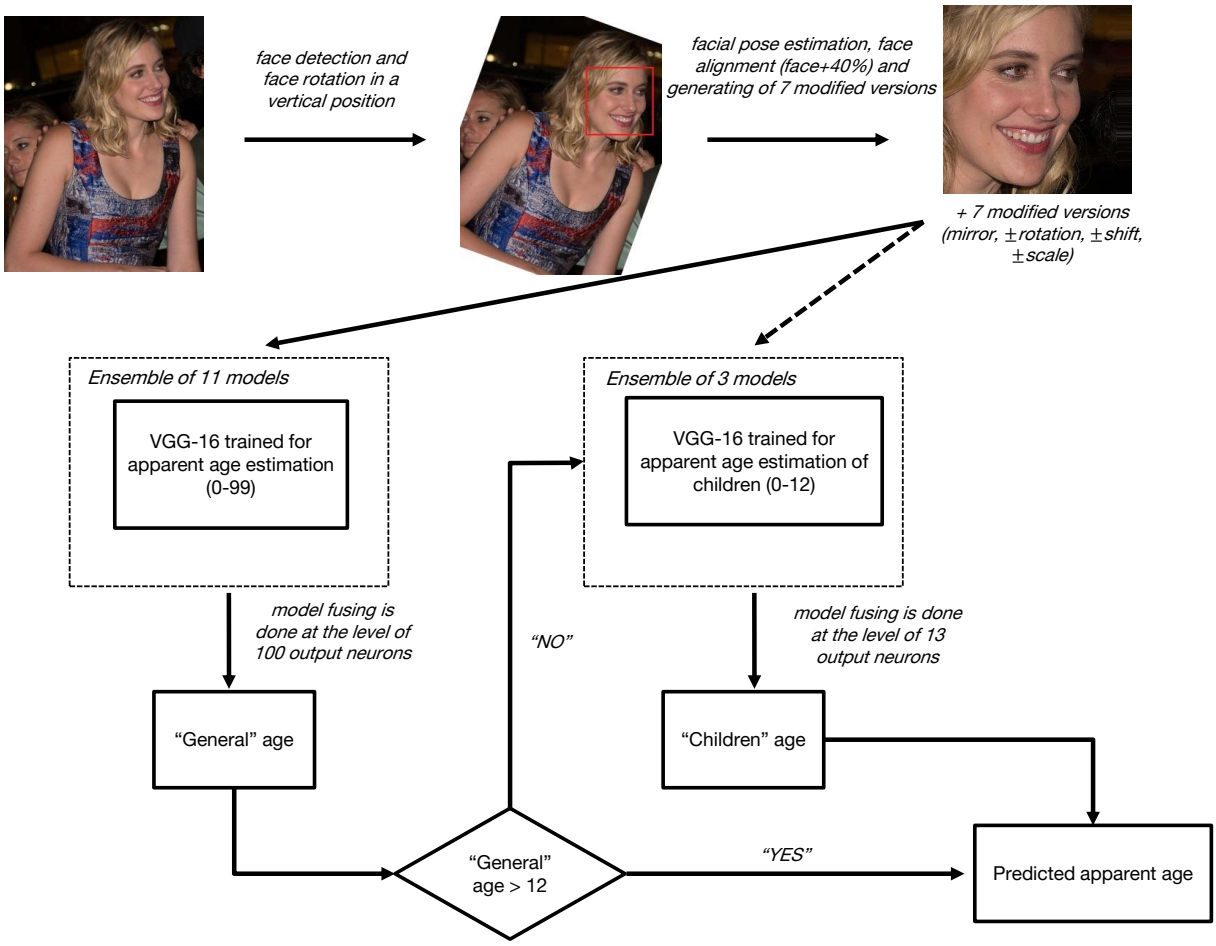



Figure 5.4.4 – Pipeline of our final solution for ChaLearn AAEC at test stage.

age prediction is calculated as an expected value of these probabilities: $children\_age = \sum_{i=0}^{12} i * p_i$. The predicted "children" age is considered as the final apparent age estimation.

### 5.4.3 Experiments

| Training / Fine-tuning | | Data augmentation | | "Children" CNN | $\varepsilon$-score |
|---|---|---|---|---|---|
| Biological | Apparent | Fine-tuning | Test | | |
| **Yes** | No | No | No | No | 0.3927 |
| **Yes** | **Yes** | No | No | No | 0.2986 |
| **Yes** | **Yes** | **Yes** | No | No | 0.2825 |
| **Yes** | **Yes** | **Yes** | **Yes** | No | 0.2782 |
| **Yes** | **Yes** | **Yes** | **Yes** | **Yes** | **0.2609** |

Table 5.4.1 – Impact of the key design choices of our solution at ChaLearn AAEC. Results are reported on the validation part of the *ChaLearn* dataset.

In this subsection, we present the experimental confirmation (using the validation part of *ChaLearn* for evaluation) of the effectiveness of the key design choices detailed in Subsection 5.4.2 for improving the performance of our solution. The experimental results are regrouped in Table 5.4.1.

In the first line of Table 5.4.1, we present the initial $\varepsilon$-score if we directly employ our biological *VGG-16* from Section 5.3 for apparent age estimation without fine-tuning on the *ChaLearn* images. This score (0.3927) is to be compared with the score in line 2 (0.2986) which represents the performance of the "general" *VGG-16* CNN after fine-tuning on the training part of the *ChaLearn* dataset. The large gap of almost 0.1 of $\varepsilon$-score (*i.e.* 24%) between these 2 results clearly demonstrates the difference between apparent and biological age estimations as well as the importance of fine-tuning on the competition data.

The data augmentation during the fine-tuning for apparent age estimation (line 3 of Table 5.4.1) has proved to be very effective gaining us about 0.015 of $\varepsilon$-score (*i.e.* 5%) with respect to fine-tuning without data augmentation. The data augmentation during the test stage (as explained in Paragraph 5.4.2.4) has been effective as well: a gain of about 0.005 in terms of $\varepsilon$-score *i.e.* 2% (line 4 of Table 5.4.1).

Finally, the last line of Table 5.4.1 proves the importance of the accurate age estimation of children. Adding a separate model for this age category has improved our validation score by about 0.017 of $\varepsilon$-points (*i.e.* 6%).

### 5.4.4 AAEC Results

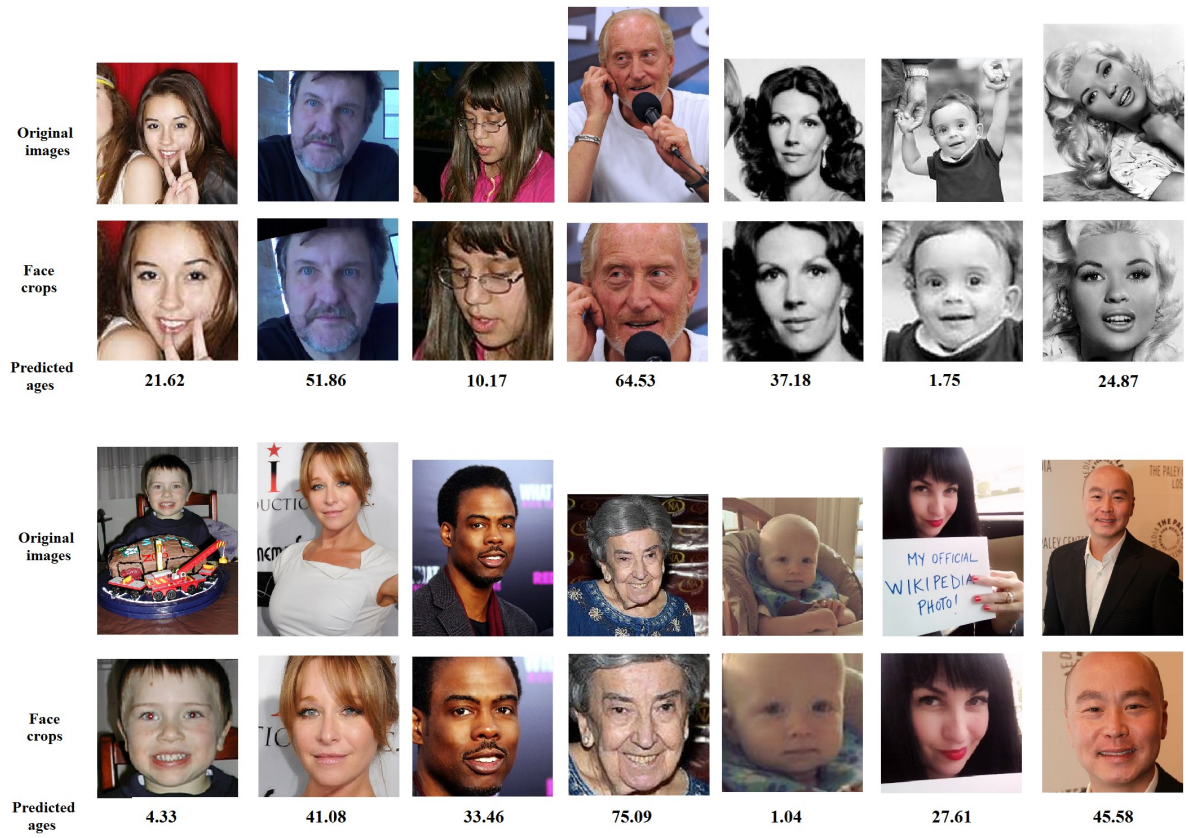



Figure 5.4.5 – (Better viewed in color). Examples of apparent age estimation of test images from the *ChaLearn* dataset. Ground truth is unknown, so only estimations by our solution are provided.

The final results of the second edition of the ChaLearn LAP AAEC are presented in Table 5.4.2.

Our team (**OrangeLabs**) has won the first place largely outperforming all other participants. Our

| Position | Team | $\varepsilon$-score | Reference |
|---|---|---|---|
| **1** | **OrangeLabs** | **0.2411** | This work |
| 2 | palm_seu | 0.3214 | [Huo+16] |
| 3 | cmp+ETH | 0.3361 | [Uři+16b] |
| 4 | WYU_CVL | 0.3405 | — |
| 5 | ITU_SiMiT | 0.3668 | [CMAKE16] |
| 6 | Bogazici | 0.3740 | [Gur+16] |
| 7 | MIPAL_SNU | 0.4569 | — |
| 8 | DeepAge | 0.4573 | — |

Table 5.4.2 – Final results of the second edition of the ChaLearn AAEC [Esc+16].

final score on the test dataset ($\varepsilon = 0.2411$) improved our best result obtained on the validation dataset ($\varepsilon = 0.2609$) by about 0.02 of $\varepsilon$-points (*i.e.* 8%). As the winners of the previous year's competition [RTVG16], we have experienced a significant gain of performance due to merging of multiple models which have been trained using cross-validation.

In Figure 5.4.5, we present some examples of apparent age estimation by our solution on images from the competition test dataset.

After comparing our solution with the ones of other participants, we believe that our solution has two decisive aspects which distinguish it from the alternatives. The first one is a very good starting point (*i.e.* our biological age *VGG-16* CNN designed in Section 5.3) for the apparent age estimation fine-tuning, while the second one is the usage of a separate "children" CNN. Indeed, all 8 best teams which are mentioned in Table 5.4.2 have used CNN-based approaches pretraining their models on external biological age datasets (often *IMDB-Wiki* as ourselves), and have employed the data augmentation (which is the standard "competition trick").

Thus, the obtained results further confirm the choice of the training strategy for our biological age *VGG-16* CNN (in particularly, LDAE age encoding and face recognition pretraining).

## 5.5 Conclusion

The presented chapter has dealt with one of the primary problems of this thesis: design of the top performing CNNs for gender recognition and age estimation.

We have started in Subsection 5.2 by identifying and comparing the key CNN design and training parameters which impact the effectiveness of the CNN optimization for the studied problems. As a result, we have made some important observations which can be summarized as following:

1. Age estimation is a more challenging problem than gender recognition. It requires bigger training datasets and deeper CNN architectures.

2. Face recognition pretraining helps subsequent training for gender recognition and age estimation. Face recognition pretraining has been shown more effective than general task one for both studied problems, and especially for gender recognition.

3. LDAE age encoding is more effective than commonly employed 0/1-CAE or RVAE encodings for training of age estimation CNNs.

The listed conclusions of Subsection 5.2 have been utilized in the following Subsection 5.3 to design our best *ResNet-50* CNN for gender recognition and *VGG-16* CNN for biological age estimation. These models have reached the state-of-the-art accuracies on three most popular public benchmarks for gender and age prediction, namely: *LFW*, *MORPH-II* and *FG-NET*. In particular, our *VGG-16* CNN performs at least as good as an average human for biological age estimation (*cf.* Paragraph 5.3.3.3).

Finally, our winning entry at the ChaLearn AAEC (which has been described in Section 5.4) has proved that the designed biological age estimation model can be effortlessly adapted for apparent age estimation as well.

Chapter 6

# Gender/Age Synthesis and Editing in Face Images

**Contents**



## 6.1 Introduction

In this last contribution chapter, we address the second primary objective of the present manuscript which consists in designing deep models for synthesis and editing of human faces with the required gender and age attributes.

Such models are often used as means of data augmentation and face normalization prior to automatic face analysis. The complete list of practical motivations as well as the potential domains of applications for face synthesis and editing have been presented in Chapters 1 and 3, but the general scientific interest for studying these problems can be summarized by a famous aphorism of Richard Feynman which states "what I cannot create, I do not understand".

More formally, as introduced in Chapter 1, gender and age prediction allows annotating human faces, while face synthesis and editing allows generating new faces with the required annotations. In this sense, the contributions presented below are complementary to the ones of Chapter 5.

In particular, we open this chapter by firstly discussing how generative deep models can be used for synthesis and editing of images in general, and then by focusing on the existing face-related approaches highlighting their limitations in Section 6.2. After that, the problems of face synthesis and editing are addressed in a sequential manner.

Indeed, in Section 6.3, we start by training a conditional generative model of face images, which is able to produce synthetic faces with the required demographic parameters. The resulting generative model becomes the core of our novel method for aging/rejuvenation and gender swapping of an input face, which is the primary contribution of the chapter. More precisely, we describe the basic part of our face editing method in the second part of Section 6.3 (*cf.* Subsection 6.3.3), while in Section 6.4, we propose an amelioration of our approach which eventually allows its application for improving an off-the-shelf face recognition software in a cross-age face verification scenario. The key advantages of our face editing method, distinguishing it from existing alternatives, are (1) high visual fidelity of the resulting face images, (2) quasi-perfect preservation of the original person's identity after face editing, and (3) finally, its universality which means that our method can be easily adapted to modify any face attributes (and not only gender and age).

## 6.2 Face Editing with Conditional Generative Models

In Section 3.3 of Chapter 3, we have discussed different classical methods of face editing observing a common limitation which is shared by many of them. As a matter of fact, the observed methods are fit uniquely for editing of a single face attribute (*e.g.* gender or age), and their application for other face attributes is impossible without redesigning of the whole algorithm. Even more, the majority of the aging/rejuvenation models from Chapter 3 can do only one out of two tasks at a time (*i.e.* either aging or rejuvenation), and in order to do the other one, a separate model must be trained. These constraints are counter intuitive and limit the possibilities for the practical application of such methods.

Instead, in this chapter, we are looking for a *universal* face editing method, which can be equally easy applied for editing of various face attributes. The recent development of deep conditional generative models provides a promising tool to achieve this goal. Indeed, as explained in Chapter 2, conditional generative models learn to imitate the joint distribution $p(x,y)$ of human faces $x$ and face attributes $y$ which enables them to intrinsically model all face variations together.

There are two classes of deep conditional generative models, namely conditional Variational AutoEncoders (cVAEs) and conditional Generative Adversarial Networks (cGANs), which allow (1) sampling from the joint distribution $p(x,y)$ of faces $x$ and face attributes (conditions) $y$, and (2) full control over the synthesized face images via selection of the latent vectors $z$ from the latent space $N^z$ with a known distribution. In particular, a pair of a latent vector $z$ and a face condition $y$ (in our case, $y$ encodes the required gender and age) completely defines the face which will be synthesized by a cVAE or a cGAN.

An important property of these two classes of conditional generative models is that both of them learn to implicitly disentangle the face information encoded by the latent vectors $z$ and by the conditions

$y$ [Yan+16; Lar+16; Per+16] For example, if $y$ encodes gender and age information (as it is the case in the present chapter), then $z$ encodes everything apart from gender and age (this is explained in more details in Subsection 6.3.2).

This property makes cVAE and cGAN particularly suitable for face editing. Indeed, imagine that an input face $x_{(y^0)}$ with initial face attributes $y^0$ is approximated by the synthetic face $\bar{x}_{(y^0)} = G(z^*, y^0)$, where $G$ is either a generator of a cGAN or a decoder of a cVAE (*cf.* Section 2.4 of Chapter 2) and $z^*$ is a particular latent vector estimated with respect to the given input. Then the same latent vector $z^*$ can be used in a pair with the target face attributes $y^1$ to synthesize the final result of face editing: $\bar{x}_{(y^1)} = G(z^*, y^1)$. In other words, face editing with conditional generative models is trivial once an optimal latent vector $z^*$ (enabling to synthetically reconstruct the initial face $x_{(y^0)}$) is found.

cVAEs have an explicit mechanism for estimating $z^*$ given an input face $x$ with facial attributes $y$. Indeed, being an autoencoder, cVAE consists of an encoder $E(x) : N^x \to N^z$ and a decoder $G(z, y) : N^z \times N^y \to N^x$, where $N^x$ is the space of face images, $N^z$ is the latent space and $N^y$ is the space of conditions. During the training, the encoder learns to approximate the inverse mapping of the decoder: $x \approx G(E(x), y)$. Therefore, $z^*$ can be simply found by processing of the input face by the encoder: $z^* = E(x)$. However, as explained in Chapter 2, cVAEs are known to produce blurry images, which often results in poor preservation of the particularities of the original faces. Thus, the loss of the most discriminative facial details is clearly visible in several examples of face reconstruction with cVAE presented in Figure 6.2.1-(a). This issue of a poor quality of the synthetic faces is the main reason why, despite the theoretical soundness of cVAEs, in this chapter, we have chosen to employ cGANs in order to address the problems of face synthesis and editing.

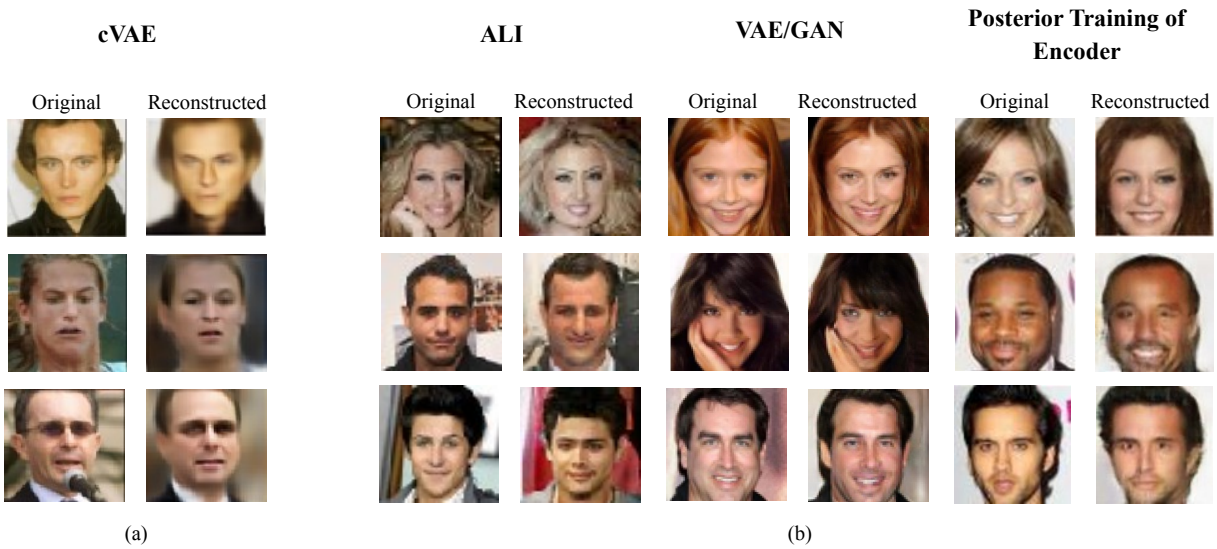



Figure 6.2.1 – (Better viewed in color). Input face reconstruction with conditional generative models. (a) cVAE [Yan+16]; (b) several cGAN-based approaches: ALI [Dum+17], VAE/GAN [Lar+16] and using the encoder $E$ which is trained posterior to the cGAN training [Per+16]. Examples are extracted from the original articles.

The synthetic images produced by cGANs are indeed much more realistic and sharp than those produced by cVAEs [Goo16]. However, the original cGAN framework does not contain an encoder $E$ able to directly estimate a latent vector $z^*$ which complicates its application for face editing.

Therefore, a number of studies has been devoted to circumvent this problem. For example, Dumoulin et al. [Dum+17] proposed the ALI (Adversarially Learned Inference) model extending the GAN framework with a second generator. More precisely, ALI [Dum+17] is composed of two generators: the one which synthesizes images from latent vectors and conditions $G_x : N^z \times N^y \rightarrow N^x$ and the other one which maps in the opposite direction: $G_z : N^x \rightarrow N^z \times N^y$. The two generators are trained simultaneously with a single discriminator $D(z,y,x)$ which tries to distinguish the triples $(z,y,x)$ issued from $G_x$ from those issued from $G_z$. Once ALI is trained, the generator $G_z$ can be used for inferring the latent vector $z^*$ for an input face $x$.

In a similar way, Larsen et al. [Lar+16] designed a VAE/GAN model regrouping cGAN and cVAE in one framework. In VAE/GAN, a cGAN and a cVAE share the same generator (called "decoder" in the cVAE context) which allows to train the two models together. The authors of VAE/GAN [Lar+16] were one of the first to apply a cGAN-based model for face editing. In particular, they reported results on editing several binary facial attributes (such as color of hair, presence of smile, gender, etc.)

Both ALI and VAE/GAN integrate the inference of the latent vector $z^*$ directly in the training process. However, the studies of Zhu et al. [Zhu+16] and Perarnau et al. [Per+16] demonstrated that an encoder $E$ for a cGAN can be also trained posterior to the training of a cGAN. This significantly simplifies the convergence of cGANs with respect to ALI and VAE/GAN, and often results in synthetic faces of superior quality [Per+16]. Thus, in Figure 6.2.1-(b), we provide some examples of the face reconstruction by ALI, VAE/GAN and posteriorly trained encoder, which are extracted from the respective articles.

As one can tell from Figure 6.2.1, the faces reconstructed with cGAN-based methods (Figure 6.2.1-(b)) have more details and are visually more plausible than those which are reconstructed with cVAE (Figure 6.2.1-(a)). At the same time, the cGAN-based reconstructions are also far from being perfect. In particular, despite the high visual similarity between the original and the cGAN-reconstructed faces, a human observer directly perceives that the person identities in the original and reconstructed faces are not the same.

This is an extremely important issue, because a person's identity is something which we definitely want to preserve when editing face attributes. Below, we address the stated issue designing a novel face editing method which is able to quasi-perfectly preserve the original person's identity.

## 6.3 Gender and Age Conditioned Generative Adversarial Network

In this section, we design a first cGAN which is able to synthesize human faces of high visual fidelity with the required gender and age. Here and further in this work, we refer to our model as GA-cGAN (which stands for Gender/Age-conditioned Generative Adversarial Network).

Section 6.2 explains that face editing with a cGAN is trivial if there is a mechanism to infer a latent vector $z^*$ allowing the generator of the cGAN to reconstruct an input face. Despite a number of works [Lar+16; Per+16] applied cGANs to editing of various face attributes, to the best of our knowledge, no existing study proposes a cGAN-based model which performs artificial aging/rejuvenation. In order to make the latter possible with our GA-cGAN, in the present section, we propose a novel approach for the inference of $z^*$, which is focused on preserving the original person's identity.

### 6.3.1 Design and Training of GA-cGAN

| Generator $G$ | Discriminator $D$ | Encoder $E$ | Face Recognition $FR$ |
|---|---|---|---|
| 4x4(2x2)@512, BN, ReLU | 4x4(2x2)@64, LReLU | 5x5(2x2)@32, BN, ReLU | 3x3(1x1)@32, BN, ↓, ELU |
| 4x4(2x2)@256, BN, ReLU | 4x4(2x2)@128, BN, LReLU | 5x5(2x2)@64, BN, ReLU | 3x3(1x1)@64, BN, ↓, ELU |
| 4x4(2x2)@128, BN, ReLU | 4x4(2x2)@256, BN, LReLU | 5x5(2x2)@128, BN, ReLU | 3x3(1x1)@128, BN, ↓, ELU |
| 4x4(2x2)@64, BN, ReLU | 4x4(2x2)@512, BN, LReLU | 5x5(2x2)@256, BN, ReLU | 3x3(1x1)@256, BN, ↓, ELU |
| 4x4(2x2)@3, Tanh | 4x4(1x1)@1, Sigmoid | FC 4096, BN, ReLU | **3x3(1x1)@512, BN, ↓, ELU** |
| — | — | FC 100 | FC 4096, BN, ELU |
| — | — | — | FC 4096, BN, ELU |
| — | — | — | FC 128, Normalize |

Table 6.3.1 – CNN architectures used in Chapter 6: the generator $G$ and the discriminator $D$ are parts of GA-cGAN, while the encoder $E$ and the face recognition CNN $FR$ are used for the latent vectors inference. kxk(sxs)@M denotes a convolutional layer (or deconvolutional layer for $G$) of M feature maps with kernels of size k and stride s; FC N denotes a fully-connected layer of N neurons; BN denotes batch normalization; ↓ denotes 2x2 MaxPooling.

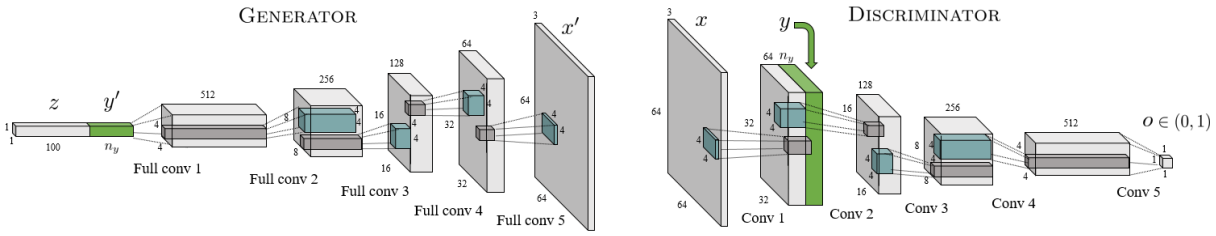



Figure 6.3.1 – (Better viewed in color). (Extracted from [Per+16]). Optimal injection of conditional information into the DCGAN framework. Conditions are provided at the input of the generator $G$ and at the first layer of the discriminator $D$.

As explained in Chapter 2, GAN training is known to be very unstable and difficult to control. Therefore, we follow the widely adopted practices to facilitate the convergence of GANs. Thus, we employ the DCGAN CNN architectures for the generator $G$ and the discriminator $D$ which were proposed by Radford et al. [RMC16]. The two architectures are detailed in the first two columns of Table 6.3.1. As in the original work [RMC16], the generator $G$ of our GA-cGAN is fed with latent 100-dimensional vectors $z \in N^z$, $N^z = \mathbb{R}^{100}$ which are sampled from the standard normal distribution $z \sim N(0, I)$, and produces 3-channel RGB images of size 64$x$64.

In order to encode a person's age, we have identified six age categories: "0-18", "19-29", "30-39", "40-49", "50-59" and "60+" years old. They have been selected so that there are at least 5K images belonging to each category in the training dataset (which will be presented in Paragraph 6.3.4.1). Thus, the age conditions $y_a$ of GA-cGAN are 6-dimensional one-hot vectors. The gender conditions $y_g$ are simply encoded with binary values. Therefore, the complete conditional vectors $y$ of GA-cGAN are 7-dimensional vectors which are composed by concatenating $y_g$ and $y_a$: $y = (y_g, y_a)$.

During the cGAN training, there are two types of conditional information (*cf.* the cGAN definition in Chapter 2): (1) the real gender and age annotations $y$ which are sampled from the training dataset with the respective training images $x$: $(x, y) \sim p_{data}$, and (2) the random conditions $\tilde{y}$ which are sampled along with the latent vectors $z$ to generate synthetic face images. We sample the random gender and age conditions $y_g$ and $y_a$ from the discrete uniform distribution for the sake of balanced training of all gender/age categories.

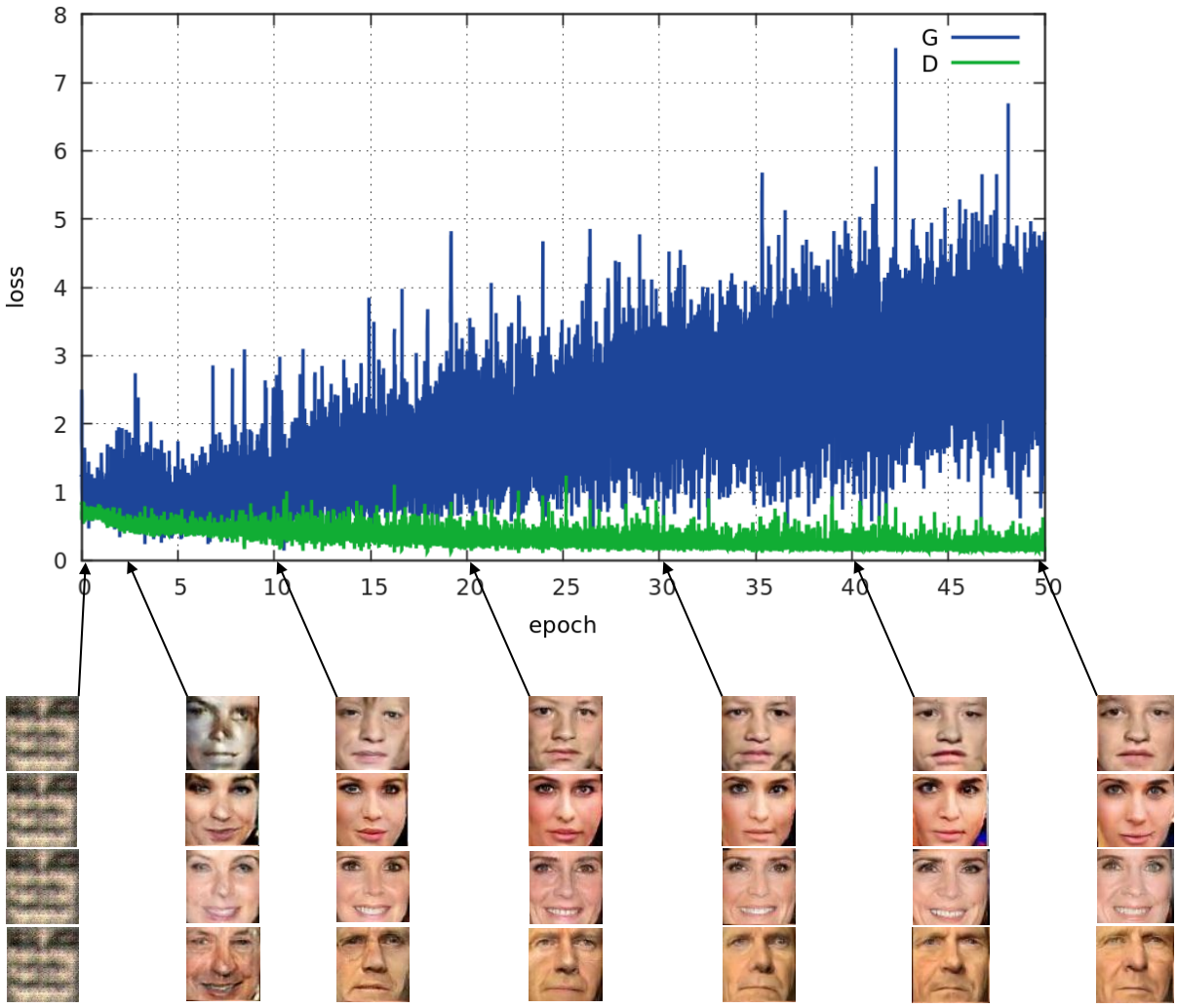



Figure 6.3.2 – (Better viewed in color). Training progress of our GA-cGAN. Top: the loss curves for the generator $G$ and the discriminator $D$. Bottom: examples of the synthetic faces generated by $G$ at different stages of training (inputs $(z, y)$ are fixed).

The original DCGAN from [RMC16] is designed for non-conditional GAN training. Therefore, in order to extend DCGAN and to optimally introduce the gender and age information both in the generator and in the discriminator, we use the conclusions of Perarnau et al. [Per+16]. In particular, we concatenate the conditional vectors $y$ to the latent vectors $z$ at the input of $G$, and we inject $y$ in the form of an additional 2-dimensional map at the first convolutional layer of $D$ (*cf.* Figure 6.3.1).

Our GA-cGAN is trained on "face-only" cropped images contrary to "face+40%" crops which are employed in Chapter 5. The reason for that is two-fold: firstly, the more narrow crops compensate the limited resolution of the synthetic images (as the retina of 64x64 pixels is imposed by the DCGAN architecture) by "zooming into" the face region, and secondly, they improve the quality of face reconstruction of GA-cGAN by better preserving the person identity traits (*cf.* Subsection 6.3.4 for the detailed comparison).

GA-cGAN has been trained with the Adam optimizer [KB14] ($\beta_1 = 0.5$, $\beta_2 = 0.999$, $\varepsilon = 10^{-8}$) with a learning rate of 0.002 and a mini-batch size of 64 during 50 epochs (these particular values of the training hyperparameters have been selected according to the recommendations in [Per+16]). We have used the

matching-aware discriminator method for cGAN training [Ree+16] in order to accelerate the learning of the conditional distribution, which has proven to be very effective. The training curves for $G$ and $D$ are presented in Figure 6.3.2. One may observe that over the time, the average loss of the generator slightly grows, while the one of the discriminator progressively goes down, which is quite usual behaviour for the cGAN training (the particular GAN training dynamics can take a variety of shapes depending on the dataset, hyperparameters and even the random initialization). In order to illustrate the progress of the generator over the training, we also provide some random samples which were generated by $G$ from the same latent vectors $z$ and conditions $y$ at different epochs in Figure 6.3.2.

### 6.3.2 Synthetic Face Manifold

Once GA-cGAN training is finished, the generator $G$ can generate plausible synthetic faces following a distribution similar to the one of natural faces. Varying the latent vectors $z$ and the conditions $y$ at the input of $G$ results in different synthetic faces $\bar{x}$ at its output. Importantly, the mapping learned by the generator is continuous meaning that small variations in latent vectors $z$ and conditions $y$ result in small variations in the generated faces $\bar{x}$. Thus, an ensemble of all possible synthetic faces produced by $G$ with various $z \in N^z$ and $y \in N^y$ taken together form a *synthetic manifold* $\bar{N}^x$ (the term was coined in [Zhu+16]).

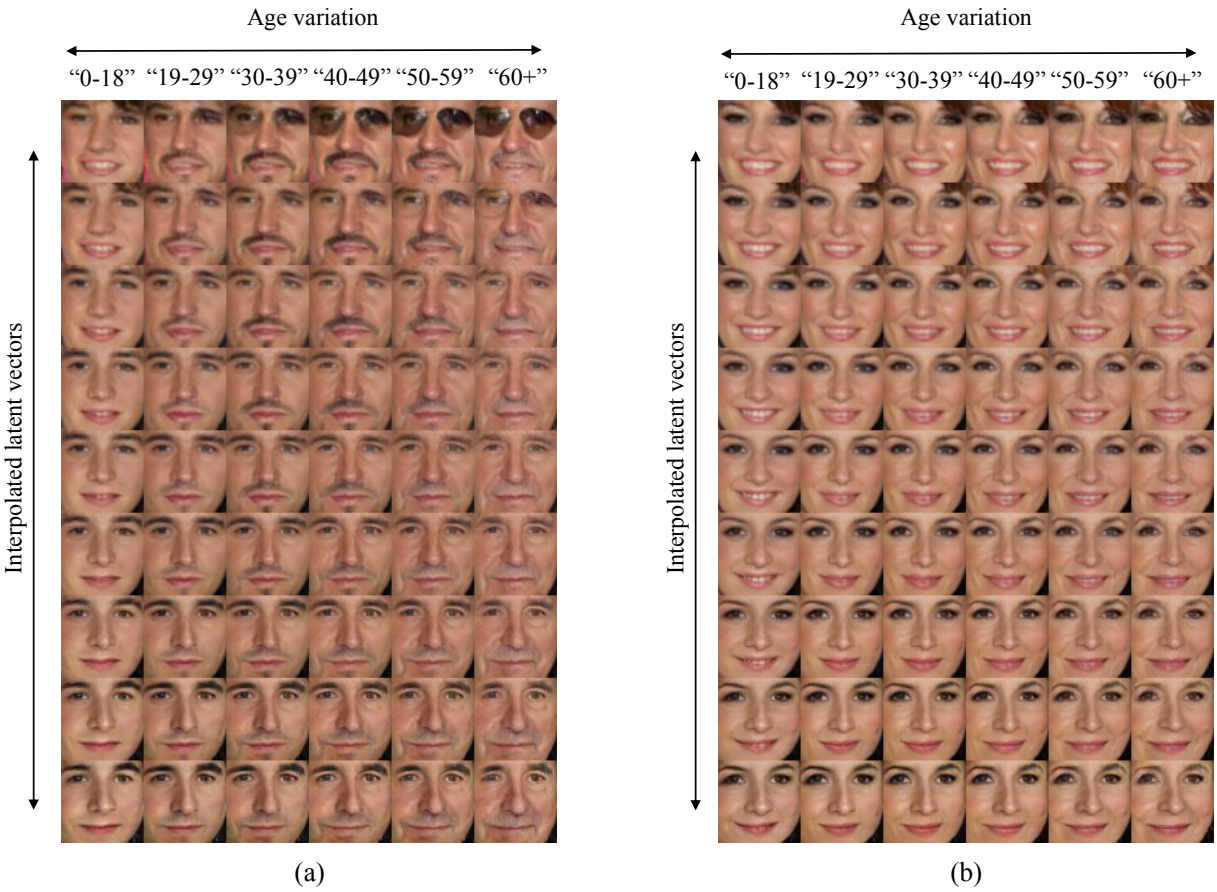



Figure 6.3.3 – (Better viewed in color). Exploration of the synthetic manifold $\bar{N}^x$ learned by our GA-cGAN. Each face has been synthesized by the generator $G$ with particular inputs: $(z, (y_g, y_a))$. The rows illustrate progression in the latent space $N^z$, the columns represent six age conditions $y_a$ while the binary gender condition $y_g$ is switched between (a) and (b).

In Figure 6.3.3, we present a tiny part of the manifold $\bar{N}^x$. The top and the bottom rows of both Figures 6.3.3-(a) and 6.3.3-(b) correspond to two random points from the latent space $N^z = \mathbb{R}^{100}$: $z^1$ and $z^2$ sampled from $N(0,I)$, while the seven rows in between correspond to seven equally spaced interpolated latent vectors. The columns in Figures 6.3.3-(a) and 6.3.3-(b) represent all possible sets of gender/age conditions at the input of $G$. In particular, the synthetic faces in Figure 6.3.3-(a) have been generated with $y_g$ set to "man" while those in Figure 6.3.3-(b) with $y_g$ set to "woman". At the same time, faces in each column of two Figures have been generated with a particular age condition $y_a$.

Figure 6.3.3 is a good illustration of the ability of cGANs to disentangle the face information encoded by $z$ and by $y$ (which is mentioned in Section 6.2). Thus, one may observe that in our GA-cGAN, $y_g$ and $y_a$ encode gender and age, while $z$ encodes other face attributes such as facial pose, expression etc. Interestingly, the person's identity traits are split between $z$ and $y$: gender is encoded by $y_g$ while the remaining identity information is encoded by $z$. Indeed, comparing the synthetic faces at the same (row, column) positions in Figures 6.3.3-(a) and (b) reveals that the respective pairs may have belonged to siblings or cousins.

### 6.3.3 Face Reconstruction via Manifold Projection

In Section 6.2, we highlight that the key part of face editing with conditional generative models in general and with cGANs in particular is the reconstruction of an input face $x_{(y^0)}$ by the synthetic one $\bar{x}_{(y^0)} = G(z^*, y^0)$. We assume that the initial face attributes $y^0$ of the input face (*i.e.* initial gender and age in case of our GA-cGAN) are either known in advance or can be estimated (for example, by gender/age prediction CNNs proposed in Chapter 5). Therefore, the reconstruction of the input face resumes to the inference of an optimal latent vector $z^*$. Here and below, we refer to such reconstruction as *projection of an input face $x_{(y^0)}$ onto the synthetic manifold $\bar{N}^x$*.

In the present work, we employ the approach from [Per+16] consisting in the posterior training of a separate encoder CNN to perform the synthetic manifold projection (the details are provided in Paragraph 6.3.3.1). However, as already reported in Section 6.2, this approach suffers from the poor preservation of the original person's identity in the reconstructed face despite the latter is essential for face editing. Therefore, in Paragraph 6.3.3.2, we propose a novel approach to optimize the initial manifold projection focusing on the preservation of the original identities.

#### 6.3.3.1 Initial Manifold Projection

As in [Per+16; Zhu+16], we design an encoder CNN $E$ to learn the mapping from the image space $N^x$ to the latent space $N^z$ inverting the mapping learned by the generator $G$ of GA-cGAN. The architecture of the encoder CNN is provided in Table 6.3.1. In order to train $E$, we generate a dataset of 100K {latent vector, synthetic image} pairs: $\{z^{(i)}, G(z^{(i)}, y^{(i)})\}$, $i = 1, \ldots, 10^5$, where $z^{(i)} \sim N(0,I)$ are random latent vectors and $y^{(i)} \sim U$ are random gender/age conditions sampled from the discrete uniform distribution. Given a synthetic image $G(z^{(i)}, y^{(i)})$, $E$ must output the source latent vector $z^{(i)}$. Therefore, $E$ is trained to minimize the Euclidean distances between the estimated latent vectors $\bar{z}^{(i)} = E(G(z^{(i)}, y^{(i)}))$ and the

ground truth ones $z^{(i)}$. More precisely, the loss function $L_E$ for training of $E$ is the following:

$$L_E = \frac{1}{m}\sum_{k=1}^{m}[\sum_{j=1}^{100}(\bar{z}_j^{(k)} - z_j^{(k)})^2] \tag{6.3.1}$$

where $m$ is the mini-batch size.

Further in this chapter, we refer to the manifold projections $G(E(x), y)$ (of an input face $x$ with gender/age attributes $y$) as *intital manifold projections* and denote them as $\bar{x}^0$.

#### 6.3.3.2 Identity-Preserving Projection Optimization

In Subsection 6.3.4, we experimentally show that the initial manifold projections globally approximate a human face, but they do not convey the more subtle details of the input loosing the original person's identity. Therefore, we propose a novel optimization approach to improve the initial manifold projections.

In [Zhu+16], the similar problem of image reconstruction enhancement is solved by optimizing the latent vector $z$ to minimize the pixelwise Euclidean distance between the ground truth image $x$ and the reconstructed image $\bar{x}$. However, in the context of face reconstruction, the described "pixelwise" latent vector optimization has two obvious downsides: firstly, it increases the blurriness of reconstructions and secondly (and more importantly), it focuses only on superficial details of input face images which have a strong impact on pixel level (such as image lightning, presence of sunglasses etc.), but often misses the identity traits.

Contrary to the "pixelwise" latent vector optimization, our "identity-preserving" optimization approach focuses both on reconstructing the superficial pixel-level information, and also on preserving the original human identity. The key idea is simple: given a face recognition neural network $FR$ able to recognize a person's identity in an input face image $x$, the difference between the identities in the original and reconstructed images $x$ and $\bar{x}$ can be expressed as the Euclidean distance between the corresponding embeddings $FR(x)$ and $FR(\bar{x})$. Hence, minimizing this distance should improve the identity preservation in the reconstructed image $\bar{x}$:

$$z^* = \underset{z}{\operatorname{argmin}} ||FR(x) - FR(\bar{x})||_{L_2} \tag{6.3.2}$$

In order to learn the embeddings $FR(\cdot)$, we have trained a face recognition CNN $FR$ (the architecture of which is provided in the last column of Table 6.3.1) in a similar way as the face recognition CNN in Chapter 5. More precisely, we have empirically found the last convolutional layer of $FR$ (highlighted in bold in Table 6.3.1) to be the optimal embedding layer as the good trade-off between purely superficial pixel-level information (which is encoded in the early layers of $FR$) and purely identity-dependent information (which is encoded in the fully-connected layers of $FR$).

The generator $G(z, y)$ and the face recognition network $FR(x)$ are differentiable with respect to their inputs, so the optimization problem 6.3.2 can be solved using the L-BFGS-B algorithm [Byr+95] with backtracking line search. The L-BFGS-B algorithm is initialized with the output of the encoder $E$ on the input face $x$: $z^0 = E(x)$.

Further in this chapter, we refer to the face reconstructions with "pixelwise" and "identity-preserving"

latent vector optimizations as *optimized manifold projections* and denote them respectively as $\bar{x}^{pixel}$ and $\bar{x}^{IP}$. In Subsection 6.3.4, it is shown both subjectively and objectively that $\bar{x}^{IP}$ better preserves a person's identity than $\bar{x}^{pixel}$ while keeping the superficial face details intact.

### 6.3.4 Experimental Evaluation of Manifold Projection Approaches

As already mentioned on multiple occasions in the present section, the initial face reconstruction via manifold projection is the cornerstone of the face editing with our GA-cGAN. Below, we experimentally compare the quality of the identity preservation with the synthetic reconstructions by three approaches discussed in Subsection 6.3.3, namely: using initial manifold projections $\bar{x}^0$, using "pixelwise" manifold projections $\bar{x}^{pixel}$, and finally using the proposed "identity-preserving" manifold projections $\bar{x}^{IP}$.

#### 6.3.4.1 Datasets

GA-cGAN has been trained using the *IMDB-Wiki_cleaned* dataset presented in Chapter 5. In order to stabilize the GA-cGAN training, we have limited the non-relevant variations in the training data by using only frontal faces from *IMDB-Wiki_cleaned* which have been automatically filtered using the open source pose estimator [Uři+16a]. Thus, about 120K frontal faces have been detected, 110K of which have been used for the GA-cGAN training, and the remaining 10K faces have been utilized for the face reconstruction evaluation (*cf.* Protocol 1 in Paragraph 6.3.4.2).

In order to evaluate the face reconstruction quality in more strict conditions (*cf.* Protocol 2 in Paragraph 6.3.4.2) we have also employed the *LFW* dataset.

#### 6.3.4.2 Quantitative Comparison

| Face Crop | Manifold Projection | Protocol 1 (FV accuracy) | Protocol 2 (FV accuracy) |
|---|---|---|---|
| **"face-only"** | Initial ($\bar{x}^0$) | 89.0% | 78.1% |
| | "Pixelwise" ($\bar{x}^{pixel}$) | 94.5% | 78.5% |
| | Our "Identity-Preserving" ($\bar{x}^{IP}$) | **97.6%** | **82.0%** |
| "face+40%" | Initial ($\bar{x}^0$) | 53.2% | 75.4% |
| | "Pixelwise"-optimized ($\bar{x}^{pixel}$) | 59.8% | 74.3% |
| | Our "Identity-Preserving"-optimized ($\bar{x}^{IP}$) | 82.9% | 79.8% |

Table 6.3.2 – Comparison of the identity preservation in the synthetic face reconstructions produced by GA-cGAN with three manifold projection approaches presented in Subsection 6.3.3: initial manifold projections ($\bar{x}^0$), "pixelwise" manifold projections ($\bar{x}^{pixel}$) and our "identity-preserving" manifold projections ($\bar{x}^{IP}$). Evaluation is performed according to two face crops ("face-only" and "face+40%") and two experimental protocols: in the first one, the optimal theoretical Face Verification (FV) accuracy is of 100.0%, while in the second one, the optimal FV accuracy is of 89.4% (*cf.* Paragraph 6.3.4.2 for details).

In order to quantitatively evaluate to what extent the original face identities are preserved with the synthetic reconstructions, we employ the OpenFace face recognition software [ALS16]. OpenFace has been chosen as one of the most popular, well-documented and easy-to-use open-source projects for face recognition which ensures the reproducibility of our results. In this chapter, we use OpenFace exclusively as a black box for face verification: a pair of face images is given to its input, and the

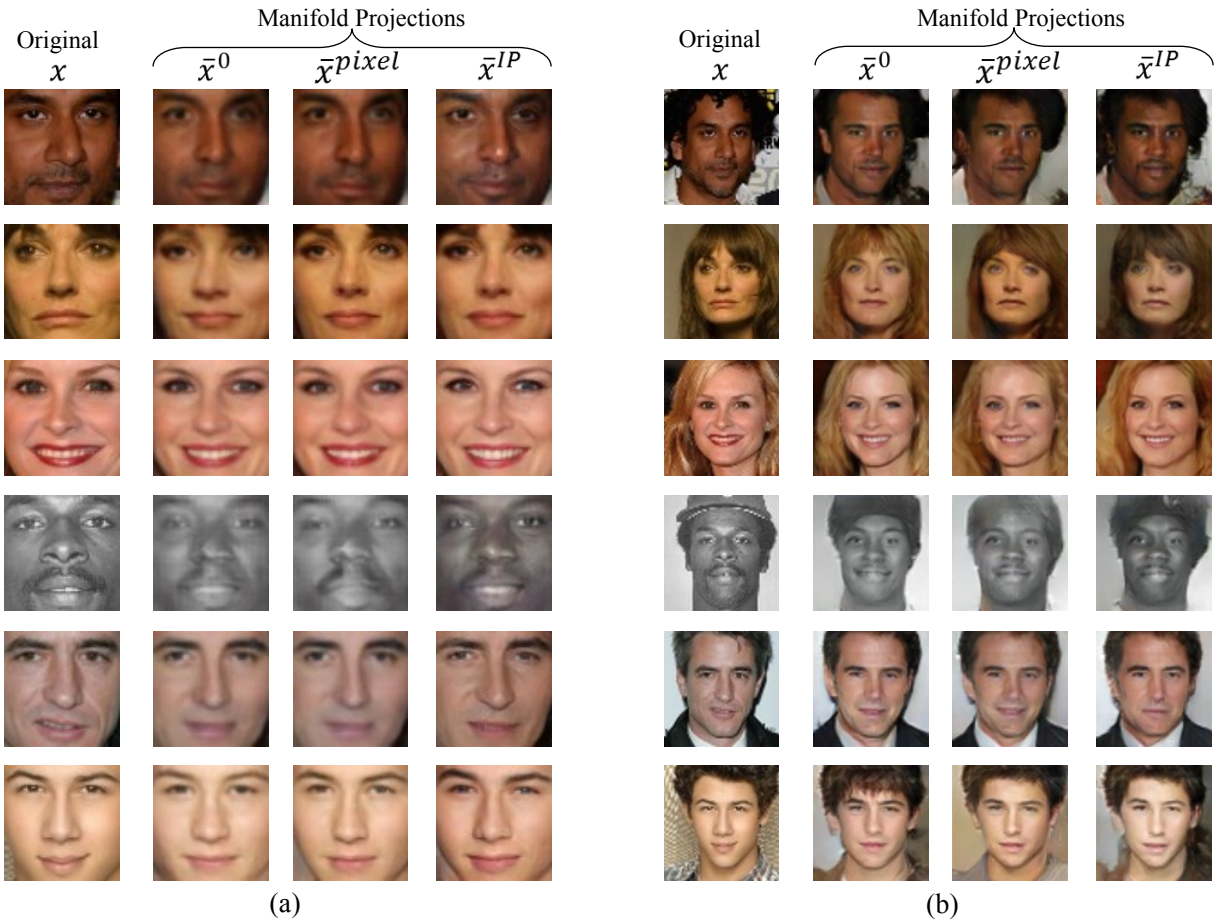



Figure 6.3.4 – (Better viewed in color). Synthetic face reconstructions via three manifold projection approaches: initial manifold projections ($\bar{x}^0$), "pixelwise" manifold projections ($\bar{x}^{pixel}$) and our "identity-preserving" manifold projections $\bar{x}^{IP}$. Reconstructions are produced by (a) "face-only" GA-cGAN, and (b) "face+40%" GA-cGAN.

software outputs a single value which is the relative distance between the provided pair of faces. The smaller is the distance, the closer are the respective human identities according to the software. The authors of OpenFace recommend using 0.99 as a threshold value to decide whether a pair of photos belong to the same person or not.

In particular, we employ OpenFace to evaluate three manifold projection approaches within two experimental protocols of varying difficulty. In the first protocol, the face verification software directly compares the original faces with the corresponding synthetic reconstructions. In practice, the evaluation is performed on 10K of *IMDB-Wiki_cleaned* images which are not used during the training of GA-cGAN (*cf.* Paragraph 6.3.4.1). For each of the three compared manifold projection approaches, we calculate the percentage of the {original, synthetic reconstruction} pairs which are predicted as "positive" by OpenFace (*i.e.* for which the software outputs a face verification distance inferior or equal to the 0.99 threshold). The higher is the resulting percentage the better is the identity preservation. The optimal score for Protocol 1 is obviously of 100.0% (which occurs when all faces are reconstructed well enough to make OpenFace believe that the original and synthetic faces belong to the same person). The results of the experimental comparison of three manifold projection approaches according to Protocol 1 are

presented in the third column of Table 6.3.2.

The first experimental protocol allows to compare the quality of the identity preservation of three manifold projection approaches. However, the ultimate objective of the present chapter is to design a face editing model which can ameliorate an off-the-shelf face verification software in the cross-age evaluation scenario. This implies that the normalized synthetic faces are sufficiently realistic to be used instead of the original ones for face verification. Therefore, in the second experimental protocol, we measure the difference between the face verification accuracies of OpenFace on original face images and on the respective synthetic reconstructions. More precisely, Protocol 2 is the following: given a pair of original faces $\{x_1, x_2\}$, they are firstly reconstructed with the synthetic ones $\{\bar{x}_1, \bar{x}_2\}$ which are then given at the input of the face verification software. The evaluation is performed following the standard *LFW* face verification protocol using 10-fold cross-validation [1]. Contrary to Protocol 1, in this experiment, the maximal score is limited by the OpenFace score on the original *LFW* images. We have estimated this score by aligning the original *LFW* dataset according to the used "face-only" crops and obtained 89.4% (which is slightly lower than the score reported in the original OpenFace article [ALS16] because of the difference in face alignments). The results of the evaluation according to Protocol 2 are regrouped in the fourth column of Table 6.3.2.

As one can see from the results in Table 6.3.2, the second experimental protocol is much more challenging than the first one. Naturally, when a reconstructed face is compared with the original one, OpenFace often classifies the pair as positive even in case of imperfect identity preservation (just because of high visual resemblance of the two images on pixel level). At the same time, such superficial similarity is not enough in Protocol 2 as in this case, the compared synthetic images are reconstructions of two different photos.

Anyway, the results of the comparative evaluation of three considered manifold projection approaches agree between both experimental protocols. Thus, we observe that "pixelwise" optimization slightly improves the quality of the reconstruction with respect to the initial manifold projections. Due to the reasons explained above, the effect is more visible in Protocol 1 than in Protocol 2. However, the performed experiments clearly demonstrate the advantage of our "identity-preserving" projection optimization approach with respect to the basic "pixelwise" optimization. The proposed approach outperforms the "pixelwise" baseline by 3.1 and 3.5 points for the two experimental protocols, respectively. In Protocol 1, "identity-preserving" optimization allows obtaining a quasi-perfect score of 97.6%. At the same time, the second experimental protocol demonstrates that the gap between the optimal face verification score of OpenFace (89.4%) and the best face verification score obtained with the synthetic faces (82.0%) is still very important. In Section 6.4, we show that in order to be used in the cross-age face verification scenario, the quality of face reconstruction with GA-cGAN should be further improved.

Finally, in order to quantitatively confirm the choice of the "face-only" crops for training of GA-cGAN (which is qualitatively motivated in Subsection 6.3.1), we evaluate the original identity preservation both with GA-cGANs trained on "face-only" and on "face+40%" crops. The comparison between the lower and the upper parts of Table 6.3.2 clearly demonstrates the advantages of "face-only" crops with respect to the "face+40%" ones according to both experimental protocols.

1. The details of the *LFW* face verification protocol are provided here: http://vis-www.cs.umass.edu/lfw/.

#### 6.3.4.3 Qualitative Comparison

The results of the experimental comparison between the manifold projection approaches and face crops presented in Paragraph 6.3.4.2 can be easily confirmed visually. To this end, in Figure 6.3.4, we illustrate the synthetic reconstructions of several faces from the evaluation part of the *IMDB-Wiki_cleaned* dataset.

The fact that "identity-preserving" manifold projections better reflect the original person's identity than alternative approaches is directly perceivable both in "face-only" and "face+40%" versions of GA-cGAN. Moreover, contrary to "pixelwise" manifold projections, the "identity-preserving" ones are less blurry which also makes them more realistic.

The advantage of "face-only" crops over the "face+40%" ones is also apparent when the same lines are compared between Figure 6.3.4-(a) and Figure 6.3.4-(b). Indeed, the "face-only" reconstructions much better transfer subtle facial details such as the form of the eyes in line 5 and the nose in line 6.

### 6.3.5 Identity-Preserving Face Reconstruction with GA-cGAN: Summary

In this section, we have designed and trained GA-cGAN, a cGAN which is able to generate synthetic faces of high visual fidelity within the required gender and age categories. In order to make possible the application of GA-cGAN for artificial face aging/rejuvenation and gender swapping, we have also proposed a novel approach for the inference of an optimal latent vector $z^*$ given an input face $x$.

Summarizing, our contributions are as following:

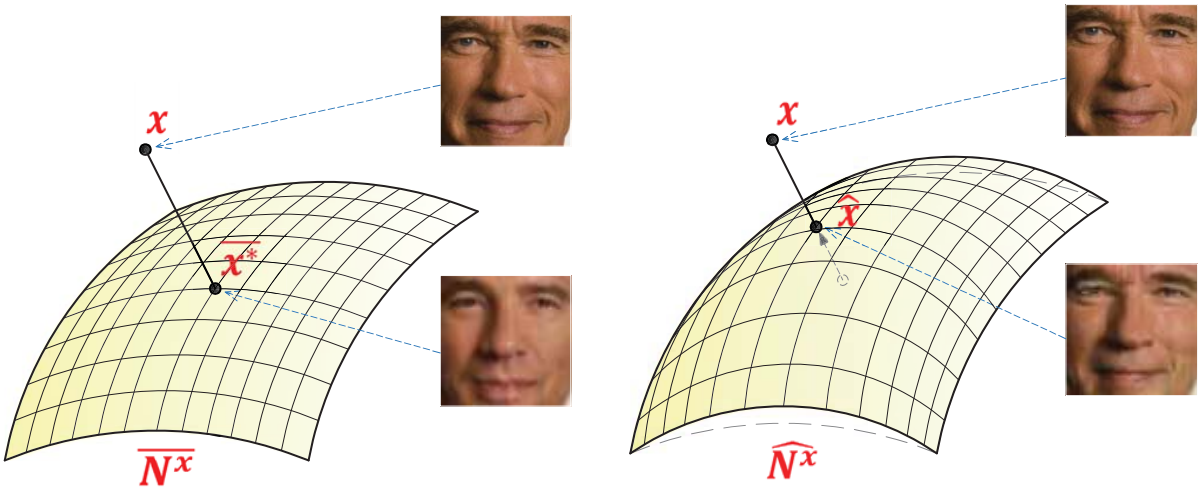
1. We have shown both objectively and subjectively that our novel "identity-preserving" manifold projection approach allows to better preserve the original person's identity in the synthetically reconstructed face than it is done by existing approaches.
2. Using "identity-preserving" manifold projection, the designed GA-cGAN can be used to convincingly perform aging/rejuvenation and gender swapping. The corresponding experiments are presented below in Paragraph 6.4.3.2.

Based on the results of this section, further in the chapter, we always employ the proposed "identity-preserving" approach of manifold projection. Therefore, for the sake of simplicity, we will use the simplified notation $\bar{x}^*$ (instead of $\bar{x}^0$ and $\bar{x}^{IP}$ used in the present section) to denote the final result of the manifold projection (*i.e.* after "identity-preserving" optimization).

## 6.4 Boosting Cross-Age Face Verification with Age Normalization

In Section 6.3, we have designed GA-cGAN, a generative model which can be applied for synthetic aging/rejuvenation and gender swapping. The main objective of the present section is to use GA-cGAN in order to improve the accuracy of an off-the-shelf face verification solution in the cross-age scenario. The idea is to normalize ages in a pair of face images (with the help of GA-cGAN) prior to face verification.

However, the experimental results (Protocol 2) in Paragraph 6.3.4.2 demonstrate that even though our "identity-preserving" manifold projection approach significantly improves the original identity preservation with respect to existing methods, the face verification score calculated with reconstructed synthetic images is more than 7 points below the one calculated with original images on the *LFW* benchmark. It




(a): before LMA (b): after LMA

Figure 6.4.1 – (Better viewed in color). Local Manifold Adaptation (LMA) approach to improve the identity preservation in the synthetically reconstructed face. (a) Input face $x$ is reconstructed by projecting it on the synthetic manifold $\bar{N}^x$ (using "identity-preserving" manifold projection as proposed in Section 6.3). (b) LMA locally modifies the synthetic manifold $\bar{N}^x$ transforming it to the new manifold $\widehat{N^x}$. As a result, the initial face $x$ and its projection $\widehat{x}$ on the new manifold are brought closer than they were before LMA.

suggests that a positive effect from age normalization by GA-cGAN can be negatively compensated by the partial loss of the original identities (we confirm this conjecture in Paragraph 6.4.3.3).

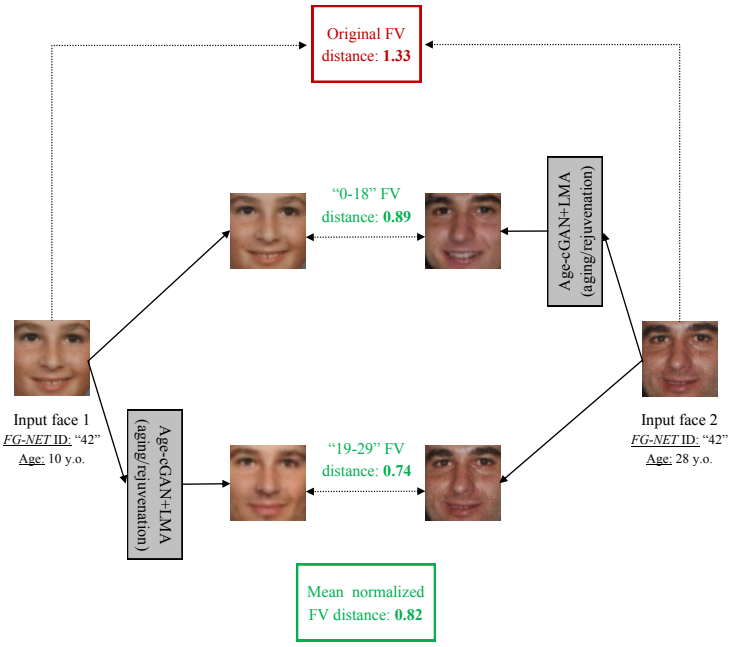
Therefore, the identity preservation in the synthetic reconstructions should be further improved before they can be used instead of original faces in cross-age face verification. To this end, in Subsection 6.4.1, we design a novel Local Manifold Adaptation (LMA) approach which extends the "identity-preserving" manifold projection presented in Section 6.3. Moreover, in Subsection 6.4.2 we propose two alternative algorithms to use the resulting GA-cGAN+LMA face editing method for age normalization.

An extensive experimental evaluation in Subsection 6.4.3 illustrates the advantages of face editing by GA-cGAN with LMA. Finally, in Paragraph 6.4.3.3, we use the designed age normalization algorithms to boost the accuracy of an off-the-shelf software in the cross-age face verification scenario .

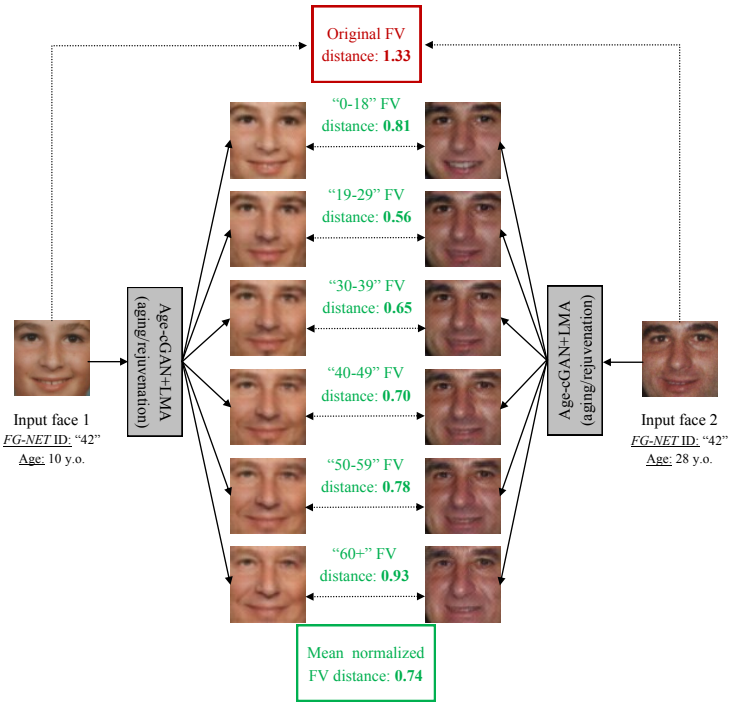
### 6.4.1 Local Manifold Adaptation

Despite GANs are arguably the most powerful generative models today, their variability and expressiveness are obviously limited. In other words, the generator $G$ which is trained on (no matter how big but) *finite* number of faces cannot exactly reproduce the details of all real-life face images with their *infinite* possibilities of minor facial details, accessories, etc. As a result, even the optimal projection $\bar{x}^*$ of a natural input face $x$ on the manifold $\bar{N}^x$ is still quite different from the original (*cf.* Figure 6.4.1-(a)) in terms of subtle facial details (*i.e.* unique form of the mouth, of the eyes, of the nose, skin particularities, etc.). When taken together, these subtle facial details transform in rather significant identity differences between the original and the reconstructed faces.

A natural way to remedy this problem is to modify the designed synthetic face manifold $\bar{N}^x$ given an input face $x$ in order to bring closer $x$ and its projection $\bar{x}^*$. In particular, we propose changing only




(a): HS age normalization




(b): FS age normalization

Figure 6.4.2 – (Better viewed in color). (a) "Half-Synthetic" (HS) and (b) "Fully-Synthetic" (FS) age normalization algorithms improving cross-age Face Verification (FV). GA-cGAN+LMA is used to perform aging/rejuvenation. A pair of faces from the *FG-NET* dataset belonging to the same person at different ages are compared with the OpenFace FV software [ALS16]. Without age normalization, the software incorrectly classifies the faces as a negative pair: the estimated FV distance of 1.33 (*cf.* red rectangles) is well above the software rejection threshold of 0.99. After age normalization, the mean estimated FV distances by the same software are of 0.82 and 0.74 for HS and FS algorithms, respectively (*cf.* green rectangles). This allows both algorithms to correctly classify the initial pair as positive.

the *local* area around the projection $\bar{x}^*$ in the manifold $\bar{N}^x$. We therefore refer to our approach as Local Manifold Adaptation (LMA) (*cf.* Figure 6.4.1-(b)). The key idea of LMA is inspired from the state-of-the-art methods [KSSS14; Shu+15] which automatically adapt their respective aging/rejuvenation models given a particular input face.

Since the synthetic manifold $\bar{N}^x$ is completely defined by the generator, LMA is performed by a slight automatic modification of the generator $G$ with respect to an input image $x$. More precisely, the key idea of LMA is the following: instead of aging/rejuvenating an input image $x_{(y^0)}$ of age $y^0$ with a general generator $G$ (issued from the GA-cGAN training), we firstly customize the general generator $G$ to better fit the input face $x_{(y^0)}$ obtaining a new generator $G_{x_{(y^0)}}$. After LMA, $G_{x_{(y^0)}}$ can produce an improved reconstruction $\widehat{x}_{(y^0)} = G_{x_{(y^0)}}(z^*, y^0)$ of the input face $x_{(y^0)}$. Our intuition suggests that if the LMA reconstruction $\widehat{x}_{(y^0)}$ is closer to the original face $x_{(y^0)}$ than the GA-cGAN reconstruction $\bar{x}^*_{(y^0)}$, then the aged/rejuvenated face $\widehat{x}_{(y^1)} = G_{x_{(y^0)}}(z^*, y^1)$ will also better preserve the original identity than the one obtained via the general generator $G$. This intuition is confirmed by the experiments in Paragraph 6.4.3.3.

The same optimization objective as in Equation 6.3.2 (*i.e.* the distance between natural and reconstructed face recognition embeddings: $||FR(x) - FR(G(z^*, y))||_{L_2}$) is used to customize $G$ for an input face $x$. However, in case of LMA, we "freeze" the previously found $z^*$ and optimize $G$ instead. Given the fact that inputs ($z^*$ and $y$) of the generator are fixed, we employ a classical backpropagation algorithm to optimize $G$. In order to make the changes of $\bar{N}^x$ local and to preserve the continuity of the synthetic manifold which is learned during the adversarial training, the number of backpropagation iterations $N_{iter}$ and the used learning rate $\mu$ should be limited. In Paragraph 6.4.3.1, we experimentally find the optimal $N_{iter}$ and $\mu$ for LMA and demonstrate both quantitatively and qualitatively the positive effect of our approach on preserving the original identities in the input face reconstructions.

In order to avoid confusion, further in this chapter, we distinguish the face editing/reconstruction with GA-cGAN via manifold projection onto the original manifold (as in Section 6.3) and via manifold projection onto the locally adapted manifold by referring to the former as "GA-cGAN" and to the latter as "GA-cGAN+LMA".

### 6.4.2 Age Normalization

The objective of age normalization is to compensate the impact of human face aging on the robustness of a face verification software. Usually, the software outputs the relative distance between a pair of input faces. After age normalization this distance should be invariant to age variations in input faces.

Obviously, age normalization can be done differently. For example, in recent works [Shu+15; Wan+16], the authors synthetically aged the younger face of a pair to the age category of the older one. Aging was preferred over rejuvenation because the respective aging methods could perform the age change only in one direction (make faces older).

This is not the case of our GA-cGAN and GA-cGAN+LMA which are able to perform both rejuvenation and aging. More precisely, a pair of input faces can be transformed to belong to any of six age categories. Hence, we can employ two principally different algorithms to normalize ages which are detailed below.

**Half-Synthetic Age Normalization** In the first algorithm, we edit only one face image of an input pair to the age category of the other. As a result, a pair of faces which is given to the input of a face verification software is composed of a synthetic face and an original one. Therefore, we further refer to the first algorithm as "Half-Synthetic" (HS) age normalization.

Due to the fact that we do not know in advance which operation (rejuvenation or aging) is more suited for a particular face pair, our HS age normalization algorithm performs both and averages the results. In particular, given a pair of faces $x_1$ of age $y_1$ and $x_2$ of age $y_2$, we generate two pairs for face verification: $\{\widehat{x_1}_{(y_2)}, x_2\}$ and $\{x_1, \widehat{x_2}_{(y_1)}\}$. Both pairs are evaluated by a face verification software which outputs two respective face distances, and the mean of these distances is taken as the final result of face verification. HS age normalization with Age-cGAN+LMA is illustrated in Figure 6.4.2-(a).

**Fully-Synthetic Age Normalization** Contrary to HS age normalization, in the second algorithm, we replace both original faces of an input pair by the corresponding synthetically edited versions which belong to the same age category. Following the same logic as for the first algorithm, we have baptised the second one "Fully-Synthetic" (FS) age normalization.

FS age normalization proceeds as following: for each pair of face images $x_1$ and $x_2$, we generate six face verification pairs $\widehat{x_1}_{(y_i)}$ and $\widehat{x_2}_{(y_i)}$, $i \in \{1,2,3,4,5,6\}$ belonging to each age category. Thus, FS algorithm performs a uniform normalization between all age categories. The mean of the six corresponding face distances is taken as the final result. FS age normalization is illustrated in Figure 6.4.2-(b).

The effectiveness of HS and FS age normalizations for improving cross-age face verification are compared in Paragraph 6.4.3.3.

### 6.4.3 Experiments

In Subsections 6.4.1 and 6.4.2, we have respectively proposed the LMA approach to improve the identity preservation in synthetic face reconstructions by GA-cGAN, and two age normalization algorithms which apply the designed generative face editing method for cross-age face verification. In this subsection, we firstly experimentally select the optimal hyperparameters $\{N_{iter}, \mu\}$ for LMA in Paragraph 6.4.3.1, and then we evaluate both mentioned contributions highlighting the necessity of LMA for identity preserving face editing with GA-cGAN and the effectiveness of the proposed age normalization algorithms in cross-age face verification scenario in Paragraphs 6.4.3.2 and 6.4.3.3, respectively.

#### 6.4.3.1 Optimal Hyperparameters for LMA

As explained in Subsection 6.4.1, LMA requires two hyperparameters to be selected: the number of backpropagation iterations $N_{iter}$ per face image and the respective learning rate $\mu$. In order to find optimal values for these hyperparamters, we perform a grid search varying $N_{iter}$ from 10 to 50 with a step of 10 and trying 7 different values for $\mu$ between 0.0001 and 0.01. In particular, we have tested five learning rates $\mu \in [0.0001, 0.001]$ which are uniformly spaced with the step of 0.00025, and two significantly bigger learning rates of 0.005 and 0.01 in order to find the learning rate limit for the LMA approach. Thus, 35 pairs of $N_{iter}$ and $\mu$ have been tested in total.

For each pair of hyperparameters $\{N_{iter}, \mu\}$, we measure how well the synthetic face reconstructions $\widehat{x}$ produced by the resulting GA-cGAN+LMA preserve the identity of the original faces $x$. To this end, we

employ the experimental Protocol 2 from Paragraph 6.3.4.2 consisting in face verification evaluation via OpenFace software on the synthetically reconstructed faces from the *LFW* dataset, because this protocol has proven to be much more challenging than Protocol 1 (*cf.* Paragraph 6.3.4.2).

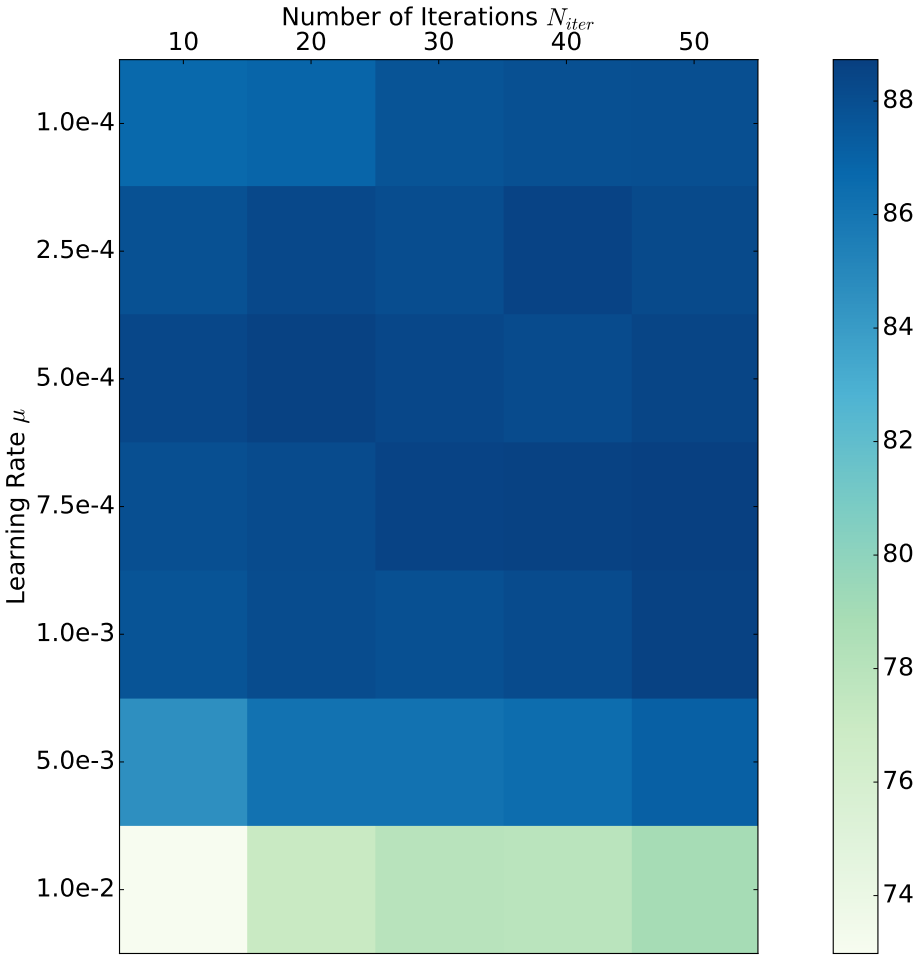



Figure 6.4.3 – (Better viewed in color). Grid search of the optimal hyperparameters (learning rate $\mu$ and the number of backpropagation iterations $N_{iter}$) for LMA. The quality of the original identity preservation by GA-cGAN+LMA is measured via OpenFace face verification software on the standard *LFW* benchmark (*i.e.* Protocol 2 from Paragraph 6.3.4.2).

In Figure 6.4.3, we illustrate the results of the grid search in the form of a heatmap, and in Figure 6.4.4, we provide some examples of the reconstructed faces with the too small, the too big and the optimal learning rates $\mu$ of 0.00001, 0.01 and 0.00075, respectively (for all examples, the number of backpropagation iterations is fixed: $N_{iter} = 50$).

Figure 6.4.4 shows that when $\mu$ is too small, the reconstructed face identity is far from the original one (in this case, LMA does not bring anything with respect to the "identity-preserving" manifold projections), while the excessively big $\mu$ destroys the synthetic manifold and the resulting reconstruction is completely degenerated. More formally, depending on the tested hyperparameters, the resulting face verification scores have varied between 73.0% and 88.7% (*cf.* Figure 6.4.3). Overall, one can observe that good learning rate values $\mu$ lie in the segment $[0.00025, 0.001]$ and LMA generally requires at least $N = 30$ iterations to converge. The best face verification score of 88.7% is obtained with $N = 50$ and $\mu = 0.00075$, so these values are selected as the optimal hyperparameters.

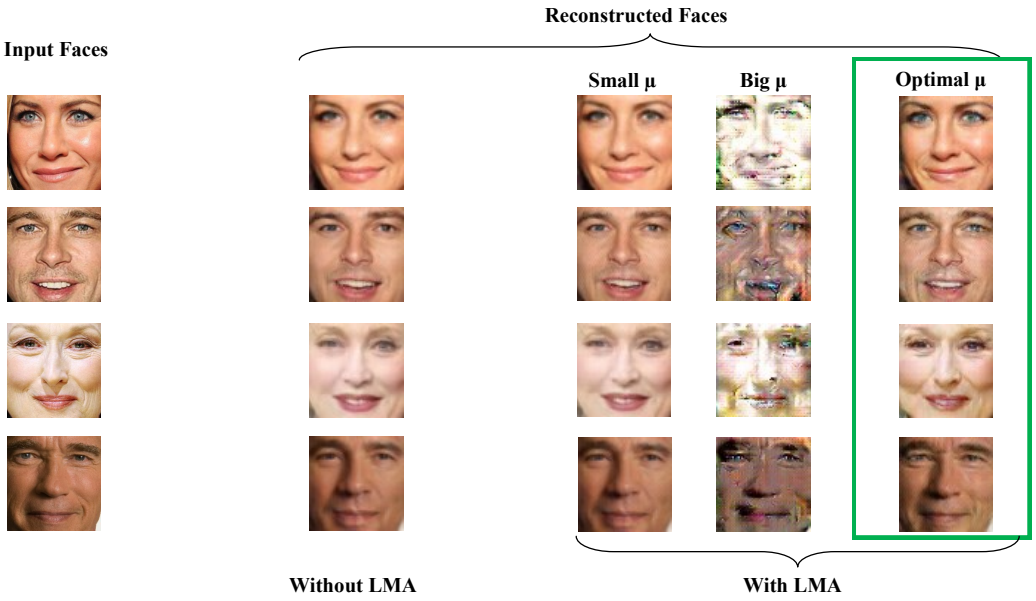



Figure 6.4.4 – (Better viewed in color). Face reconstruction by GA-cGAN with and without Local Manifold Adaptation (LMA). For LMA-enhanced reconstructions, the impact of the learning rate $\mu$ is illustrated (for all examples, the number of backpropagation iterations is fixed: $N_{iter} = 50$).

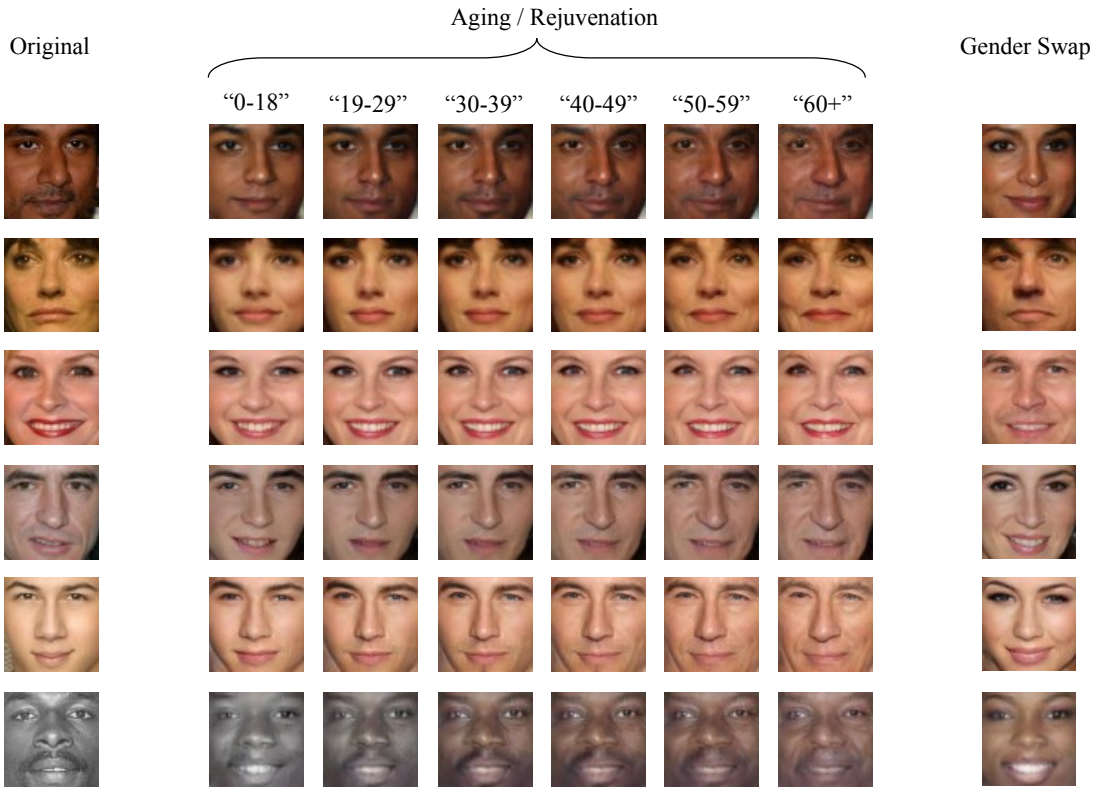
When compared to the results in Table 6.3.2, we observe that the score of GA-cGAN+LMA (88.7%) is almost 7 points above the one of GA-cGAN (82.0%), and is only 0.7 points below the maximal score calculated on the original *LFW* images (89.4%). It is also interesting to mention that when GA-cGAN+LMA is evaluated according to the experimental Protocol 1 from Paragraph 6.3.4.2, it obtains the maximal score of 100.0%. Finally, it is important to highlight that the proposed LMA approach does not extend a lot the execution time of the aging/rejuvenation process with respect to the basic GA-cGAN. Thus, 50 backpropagation iterations of LMA take about 0.4 second on Tesla K40c GPU for a single image, while the process of finding an optimal $z^*$ with L-BFGS-B takes about 1.5 seconds.

Summarizing, LMA approach further improves the quality of the identity preservation with respect to the "identity-preserving" optimized manifold projection which is proposed in Section 6.3. In Paragraph 6.4.3.3, we demonstrate that the proposed HS and FS age normalization algorithms which use GA-cGAN+LMA as a face editing method boost the face verification accuracy of an off-the-shelf face verification software in the cross-age scenario.

#### 6.4.3.2 Gender/Age Editing with GA-cGAN+LMA

The results of Paragraph 6.4.3.1 confidently demonstrate the advantage of GA-cGAN+LMA over GA-cGAN in terms of the identity preservation in the synthetic reconstructions. However, our objective is face editing and not just face reconstruction. Therefore, in the present paragraph, we visually compare the results of face editing by our GA-cGAN and GA-cGAN+LMA as well as by several alternative state-of-the-art approaches.

**GA-cGAN vs. GA-cGAN+LMA** Figure 6.4.5 illustrates synthetic face editing of several photos from the evaluation part of *IMDB-Wiki_cleaned* by GA-cGAN and GA-cGAN+LMA. Both face editing meth-




(a): Face editing with GA-cGAN


(b): Face editing with GA-cGAN+LMA

Figure 6.4.5 – (Better viewed in color). Examples of face editing of several images from the evaluation part of the *IMDB-Wiki_cleaned* dataset by (a) GA-cGAN and (b) GA-cGAN+LMA. For both methods, “Identity-Preserving” optimized manifold projections are used (*cf.* Section 6.3).

ods manage to synthesize faces with the required gender/age conditions which is particularly remarkable for GA-cGAN+LMA. Indeed, the proposed LMA approach modifies the learned synthetic manifold $\bar{N}^x$ which potentially could have negatively affected the aging/rejuvenation process and the gender swapping. This unwanted side effect is prevented by the fact that we carefully select the hyperparameters $\mu$ and $N_{iter}$ keeping the manifold alterations local.

Examples in Figure 6.4.5 also show some interesting properties of the synthetic manifold and the training dataset. For example, the synthetic faces of women tend to smile more than those of men (this is particularly visible in lines 4 and 6). Obviously, the reason for that is the learned bias from the training dataset of celebrities. In the same spirit, the black and white input photo in the last line has turned into a colored one after aging. This is due to the fact that in the training dataset, many childhood photos of contemporary actors are very old (and therefore, black and white), while the respective adulthood photos are more recent (and colorful).

Globally, the face editing results of GA-cGAN+LMA seem much more convincing than those of GA-cGAN. The only downside of the LMA approach is that it can create slight noisy artefacts when the input face is very far from the initial manifold (*cf.* for example, the last line of Figure 6.4.5-(b)). However, we believe that the pixel-level regularization should be able to remove such noise and "polish" the face editing results. This is a part of our future work.

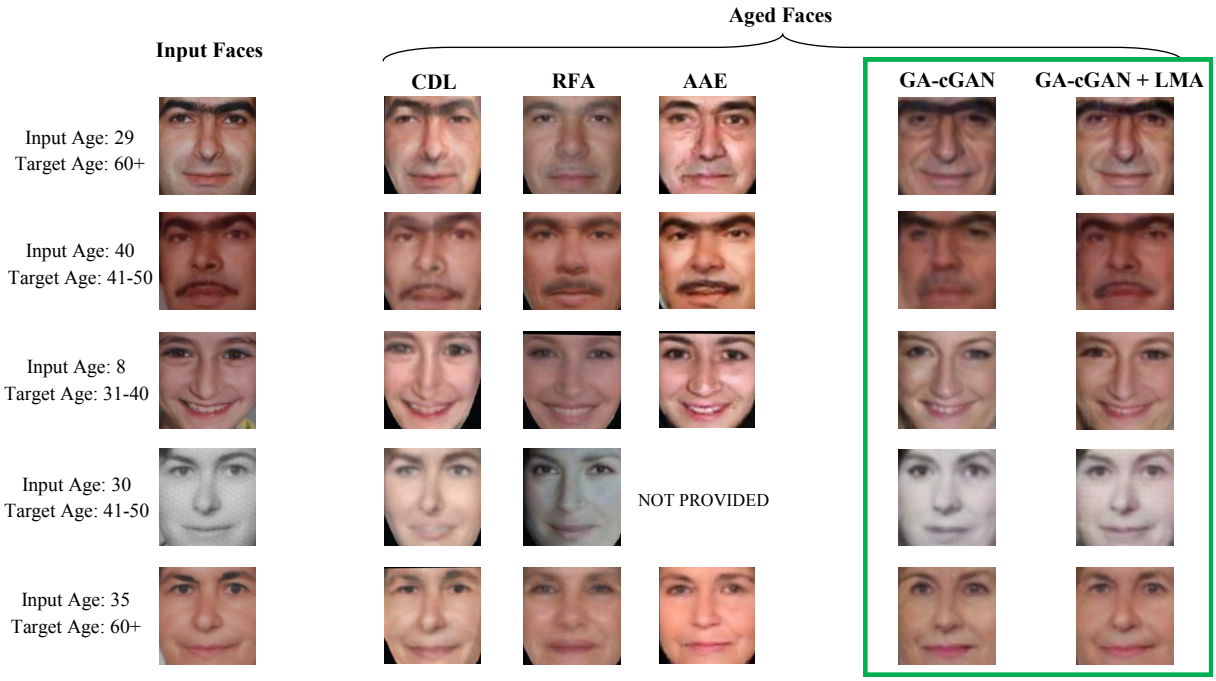



Figure 6.4.6 – (Better viewed in color). Comparison of face aging by our GA-cGAN and GA-cGAN+LMA versus alternative methods from the recent works. Each line corresponds to aging of a face image from the *FG-NET* dataset: the initial and the target ages are provided at the beginning of the line. 5 methods are compared: Coupled Dictionary Learning (CDL) [Shu+15], Recurrent Face Aging (RFA) [Wan+16], Adversarial AutoEncoder (AAE) [ZSQ17], GA-cGAN (proposed in Section 6.3), and GA-cGAN+LMA (proposed in Section 6.4). Our methods are highlighted by a green rectangle.

**Qualitative Comparison with Alternative Face Aging Methods** As explained in Chapter 3, there are few studies on gender swapping. At the same time, face aging/rejuvenation is a well-studied problem with a plethora of existing solutions. Below, we compare the proposed GA-cGAN and GA-cGAN+LMA face editing methods with the most recent alternatives [Shu+15; Wan+16; ZSQ17].

Despite some of the recent works [Shu+15; Wan+16] reported improving cross-age face verification accuracy with their respective methods, it is unfortunately impossible to quantitatively compare their improvements with our results from Table 6.4.1 (presented in the next Paragraph 6.4.3.3), because of the differences in experimental protocols. In particular, unlike us, the authors of [Shu+15; Wan+16] employed private face verification software, and did not provide explicit details on how the face verification pairs were composed from the *FG-NET* images.

Therefore, we propose to qualitatively compare the faces aged by our methods and by the listed alternatives on some examples from *FG-NET* in Figure 6.4.6. To this end, we directly use illustrations from the respective articles. It should be noted that the qualitative comparison in Figure 6.4.6 is provided only for information, and cannot serve for strong conclusions due to its subjectiveness and a very small number of available aged faces.

Nevertheless, even the few examples in Figure 6.4.6 are enough to show that our GA-cGAN+LMA face editing method is at least competitive with respect to the previous works. Indeed, contrary to CDL [Shu+15] and RFA [Wan+16], GA-cGAN+LMA quasi-perfectly preserves the original person's identity. Moreover, the synthetic images of Age-cGAN+LMA look much more realistic than those of AAE [ZSQ17] (most visible in lines 1 and 3). Finally, contrary to non-generative methods (CDL [Shu+15] and RFA [Wan+16]) which are trained only to perform face aging, our GA-cGAN+LMA is a universal face editing method. Indeed, it is trained to both perform face aging/rejuvenation and gender swapping, and it can be easily be adapted to also model other face attributes (like the presence of glasses or the presence of beard) just by adding the corresponding conditions to the cGAN.

#### 6.4.3.3 Boosting Cross-Age Face Verification

Below, we illustrate how GA-cGAN+LMA can be used for a practical application: improving the face verification accuracy in the cross-age evaluation scenario. To this end, we utilize our face editing method for face rejuvenation/aging in the frame of the two age normalization algorithms presented in Subsection 6.4.2

In particular, we have selected the *FG-NET* dataset for cross-age face verification experiments. *FG-NET* and *CACD* [CCH14] are the two most used benchmarks for cross-age face verification. However, *CACD* is composed of celebrities and have a lot of intersections with *IMDB-Wiki_cleaned* which is utilized for training of our GA-cGAN. Moreover, OpenFace face verification software, which we employ in this chapter, was trained using the images of the same celebrities as in *CACD*. Therefore, in order to avoid biased results, we have not used the *CACD* dataset for cross-age face verification in this work.

In order to evaluate the proposed HS and FS age normalization algorithms on cross-age face verification, we have selected all pairs from *FG-NET* where both face images of a pair (1) belong to the same person, (2) belong to different age categories (according to the six age categories of GA-cGAN), and (3) are of at least 10 years old (there are almost no children younger than 10 years old in the training *IMDB-Wiki_cleaned* dataset, so we do not evaluate our algorithm on children images). In the *FG-NET*

dataset, there are 1519 face pairs fulfilling the three conditions, and we select all of them as positive pairs for face verification. We also randomly select the same number of negative pairs (composed of photos of different persons) following the same conditions (2) and (3) as for positive pairs. Among the selected pairs (both positive and negative), 61.4% have an age gap of 10-20 years, 24.0% of 20-30 years, 11.1% of 30-40 years, and 3.5% of 40+ years.

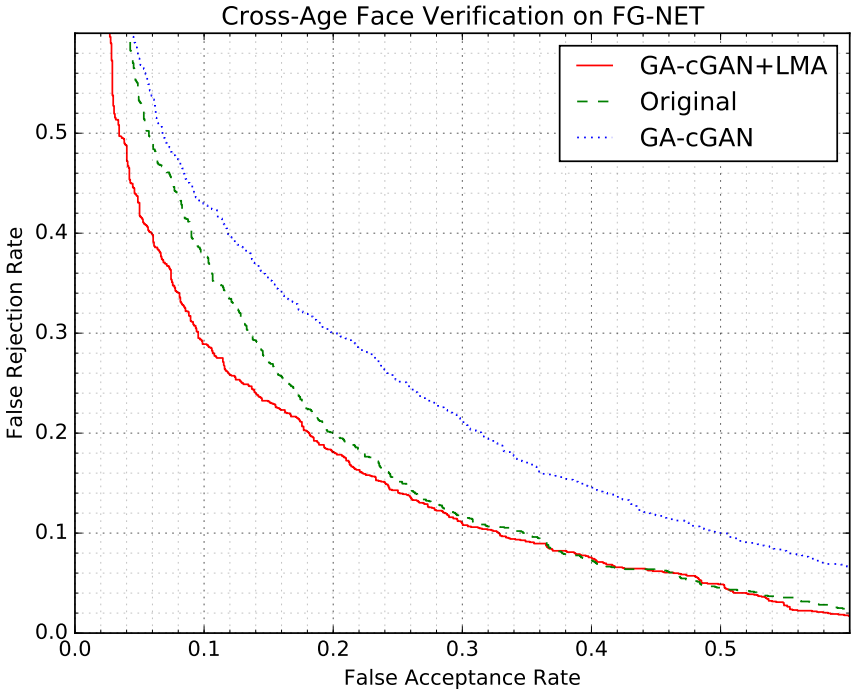



Figure 6.4.7 – "FAR vs. FRR" curves of cross-age face verification on the *FG-NET* dataset. The curves have been calculated calculated (1) on "Fully-Synthetic" (FS) age-normalized pairs generated by GA-cGAN+LMA, (2) on original pairs, and (3) on FS age-normalized pairs generated by basic GA-cGAN.

| Tested Cross-Age Pairs | | AUC | FRR@10 | EER |
|---|---|---|---|---|
| All | Normalized (FS) | **89.5%** | **29.3%** | **18.9%** |
| | Normalized (HS) | 89.2% | 31.2% | 19.3% |
| | Original | 87.6% | 37.1% | 20.5% |
| ≥ 40 | Normalized (FS) | 84.5% | 37.7% | 23.6% |
| | Normalized (HS) | **85.1%** | **37.7%** | **17.9%** |
| | Original | 80.4% | 49.1% | 26.4% |

Table 6.4.1 – Impact of age normalization with GA-cGAN+LMA on cross-age Face Verification (FV) on the *FG-NET* dataset, and comparison of 2 age normalization algorithms presented in Subsection 6.4.2: (1) Fully-Synthetic (FS) and (2) Half-Synthetic (HS). Evaluation on all cross-age positive/negative pairs and also on the pairs with a particularly huge age gap (at least, 40 years of difference). Results are provided for 3 metrics: Area Under ROC Curve (AUC), False Rejection Rate (FRR) when False Acceptance Rate (FAR) is of 10% (FRR@10), and Equal Error Rate (EER).

In Figure 8.4.5, we compare the "False Acceptance Rate (FAR) vs. False Rejection rate (FRR)" curves calculated on the original pairs and on the ones after FS age normalization. In order to highlight the benefits of the proposed LMA approach, we also plot the curve for the FS age-normalized pairs for which the corresponding aging/rejuvenation is performed without LMA (*i.e.* following the basic

GA-cGAN). Age normalization by GA-cGAN+LMA (the red curve) significantly increases the accuracy of face verification with respect to the original pairs (the green dashed curve). Obviously, the biggest improvements are due to falsely rejected original positive pairs (the upper left corner of Figure 8.4.5), *i.e.* pairs of images of the same person at different ages which are initially rejected by OpenFace, but which are accepted after removing the age difference. At the same, the blue curve of Figure 8.4.5 perfectly demonstrates that LMA is indispensable for age normalization with GA-cGAN, because without it, the positive effect of age normalization is not enough to compensate the degradation of performances due to imperfect face reconstructions.

Table 6.4.1 summarizes the comparison of face verification on original and age-normalized image pairs according to three popular metrics, namely: AUC, False Rejection Rate (FRR) when False Acceptance Rate (FAR) is of 10% (FRR@10) and Equal Error Rate (EER). The results are provided for the two age normalization algorithms proposed in this section, namely HS and FS. Both algorithms significantly boost face verification with respect to all three metrics, and in particular, for FRR@10, the improvement reaches almost 6 and 8 points for HS and FS normalizations, respectively. Moreover, the bigger is the initial age gap between the tested faces, the more important is the impact of age normalization. This is illustrated in the lower part of Table 6.4.1, where we present the scores for face verification on the test subset containing pairs with at least 40 years of age gap. In this case, both age normalization algorithms boost FRR@10 by almost 12 points.

Finally, the obtained results do not allow to convincingly tell which of the two age normalization algorithms (HS or FS) is more effective. Thus, FS normalization demonstrates slightly better scores on the whole test dataset, while HS one is vaguely preferable for the 40 years gap subset. All in all, the fact that two different age normalization algorithms significantly improve the cross-age face verification scores proves the universality and the high potential of our GA-cGAN+LMA face editing method.

### 6.4.4 GA-cGAN+LMA to Improve Cross-Age Face Verification: Summary

In the present section, we have successfully applied our GA-cGAN generative model for age normalization in order to improve the cross-age verification accuracy of an off-the-shelf OpenFace face recognition software. To make it possible, we have improved the "identity-preserving" manifold projection from Section 6.3 with the novel LMA approach. The effectiveness of the resulting GA-cGAN+LMA face editing method for age normalization has been evaluated according to two different age normalization algorithms.

More precisely, our contributions can be summarized as following:

1. The proposed LMA approach further improves the identity preservation of the "identity-preserving" manifold projection while extending its running time by only 0.4 second per image. As a result, the synthetic reconstructions by GA-cGAN+LMA can be used instead of original faces for face verification with almost zero loss of quality.
2. Synthetic aging/rejuvenation and gender swapping by GA-cGAN+LMA produce visually plausible results which are favourably compared to the state-off-the-art alternatives.
3. Age normalization based on GA-cGAN+LMA boosts FRR@10 of the OpenFace face verification software by about 8 points in average and by about 12 points in case of the biggest age gaps.

## 6.5 Conclusion

In the presented chapter, we have subsequently explored numerous problems of synthetic face generation and editing of increasing difficulty.

We have started in Subsections 6.3.1 and 6.3.2 by designing a first cGAN for face synthesis conditioned on gender and age information. We have baptised our model as GA-cGAN and has shown that it can sample random face images within the required gender and age categories which together belong to a continuous manifold.

The next step has been applying the designed model for face editing. We have demonstrated that the principal downside of existing cGAN-based methods is the fact that they poorly preserve the original person's identity. Thus, in Paragraph 6.3.3.2, we have proposed a novel approach which significantly improves identity preservation with respect to the existing alternatives.

Finally, our ultimate challenge has been improving the accuracy of an off-the-shelf face verification software in the cross-age evaluation scenario. Despite the effectiveness of our "identity-preserving" manifold projection approach, we have discovered that face editing by GA-cGAN still loses too much identity information to be directly applied for the stated problem. Therefore, in Subsection 6.4.1, we have extended GA-cGAN with the LMA approach which adapts the face editing for a particular input face. The resulting GA-cGAN+LMA face editing method boosts the accuracy of a popular face verification software on a very challenging cross-age benchmark (*cf.* Paragraph 6.4.3.3).

Chapter 7

# General Conclusion

**Contents**



This last chapter concludes the present manuscript summarizing the main results which have been obtained in the frame of this PhD. We start by recalling the key contributions of our work in Section 7.1. Then, we discuss the limitations of the proposed approaches suggesting the possibilities for the future research in Section 7.2. Finally, the publications associated with this PhD are listed in Section 7.3.

## 7.1 Summary of the Contributions

In this PhD, we have been interested in bridging the gap between the rapidly developing domain of deep learning and the domain of gender and age analysis of human faces. In this context, the following two primary objectives have been set for this work: on the one hand, learning to automatically annotate a given face image with the corresponding gender and age semantics (*i.e.* the problem of gender/age prediction), and on the other hand, learning to synthesize new face images with the required gender and age attributes (*i.e.* the problems of face synthesis and face editing).

**Preliminrary Studies** Before directly addressing these main objectives, we have started our research with two preliminary studies which have been dedicated to better understanding of CNNs, the models which have been further used throughout the whole PhD.

In particular, in the first study, we have compared the effectiveness of the features learned by CNNs versus several relevant hand-crafted feature representations for the problem of gender recognition from images of pedestrians. This problem is notorious for its complexity and the fact that a relatively small number of images (about 10K) is available for training. Despite the lack of the training data, we have shown that the features learned by a very compact *pedestrian_CNN* generalize significantly better on new datasets than the hand-crafted HOG features (even though, the latter were explicitly designed for the

detection of pedestrians [DT05]). Moreover, this first study has perfectly illustrated the power of transfer learning, as the best pedestrian gender CA (of 79.5% in the cross-dataset evaluation scenario) has been obtained by fine-tuning of *AlexNet* which was initially trained on the *ImageNet* dataset.

The second preliminary study has been focused on training of various CNNs from scratch for gender prediction from faces. The problem of face gender recognition is not only simpler than the problem of pedestrian gender recognition (thus, recent works reported gender CAs of 95+% for the former), but also plenty of face images with gender annotations are available in public access (for example, about 450K in the *CASIA WebFace* dataset). Hence, having enough training data, we have explicitly avoided the usage of pretrained CNNs in the second study, which has allowed us to measure the direct impact of different parameters of the CNN architecture on the resulting performances. As a result, by sequentially reducing the depth and the width of a relatively complex *start_CNN* architecture, we have been surprised to discover that a shallow *optimized_CNN* performs as good as the initial deep *start_CNN*. Moreover, the best gender CA of *optimized_CNN* of 96.9% is not better than the one obtained with the LBP hand-crafted features and an SVM classifier in [JC15].

All in all, the experimental results of the two preliminary studies have demonstrated that the quality of the features learned by a CNN depends not only on the depth of the neural architecture and the amount of training data, but also on the complexity of the target problem. This conclusion has strongly impacted our further research by highlighting the importance of the CNN pretraining on a more challenging problem even in the case, when there is no lack of training data for the problem of interest. In particular, it explains the necessity of face recognition pretraining prior to training of CNNs for gender/age prediction which is discussed hereafter.

**Gender Recognition and Age Estimation** Our second contribution addresses the first primary objective of this PhD which consists in finding the optimal ways of training CNNs for gender recognition and age estimation.

To this end, we have identified several parameters which, according to the previous studies, have the biggest impact on the performances of gender recognition and age estimation CNNs. These parameters concern practically all aspects of the CNN design and training, in particular: (1) the form of the CNN inputs (in our case, face crops), (2) the CNN depth, (3) the form of the CNN outputs (*i.e.* target encodings and loss functions), and (4) different forms of transfer learning (pretraining and multi-task learning).

By experimentally comparing various configurations of these parameters, we have made the following general conclusions:

1. Label Distribution Age Encoding (LDAE) is more effective for age estimation than commonly used pure classification and pure regression encodings.
2. While both face recognition pretraining and multi-task learning are helpful, the two transfer learning techniques are not complimentary, and the former is more effective. Face recognition pretraining is also better suited to the target problems than *ImageNet* pretraining.
3. On its own, gender recognition training fails to learn expressive feature representations. The performances of gender CNNs completely depend on the quality of the face recognition pretraining.

When combined with very deep CNN architectures, namely, *VGG-16* and *ResNet-50*, these observations have allowed us to train the state-of-the-art CNNs for gender recognition and age estimation. More

precisely, the designed models obtain the top performances on three mostly used benchmarks: *LFW*, *MORPH-II* and *FG-NET*. When evaluated on the very challenging private *PBGA* dataset, which is balanced between the two genders and has a uniform distribution of ages, our best CNNs reach the gender CA of 98.7% and the age MAE of 4.26 years.

Our age CNN achieves the human-level of biological age estimation which is illustrated by its comparison with the participants of the TV show "Guess My Age". Moreover, this biological age CNN can be also effortlessly adapted to estimate apparent ages. Thus, our winning entry to the ChaLearn AAEC contest has confirmed the selected training approach for apparent age estimation as well.

**Face Synthesis and Editing** Finally, our last contribution consists in proposing a novel approach for face synthesis and editing with the required gender and age semantics, which fulfils the second main objective of this PhD.

The face synthesis part of the problem is the simple one, and it has been resolved by designing GA-cGAN, the first cGAN which can synthesize faces of high visual fidelity within the required gender and age categories.

However, application of GA-cGAN for face editing requires a separate approach for the inference of an optimal latent vector $z^*$ given an input face $x$ (or in other words, for synthetic reconstruction of $x$). Therefore, we have proposed "identity-preserving" manifold projection algorithm which allows to preserve the original person's identity in the reconstructed synthetic face better than existing alternative latent vector inference approaches. In particular, when comparing the original faces with their synthetic reconstructions synthesized by GA-cGAN via the "identity-preserving" manifold projection algorithm, a typical face verification software "believes" that the two faces belong to the same person in 97.6% of cases (which is more than 3 points better than the similar score of previous approaches).

Nevertheless, some practical applications of face editing (for example, cross-age face verification) require even higher (*i.e.* quasi-perfect) quality of identity preservation. To this end, we have suggested an extension of our "identity-preserving" manifold projection algorithm, which ameliorates the latter by increasing its running time by only 0.4 seconds per face image. We have called this extension Local Manifold Adaptation (LMA), and its key idea consists in locally changing the manifold of GA-cGAN in order to reduce the distance between it and an input face prior to the manifold projection. LMA significantly improves the quality of the "identity-preserving" manifold projection algorithm which has been illustrated both quantitatively and qualitatively.

The resulting GA-cGAN+LMA face editing method has shown visually convincing results for automatic gender swapping and aging/rejuvenation in human faces. Moreover, in order to demonstrate its practical pertinence, we have employed GA-cGAN+LMA for age normalization prior to face verification in a cross-age evaluation scenario. As a result, we have obtained improvements of the face verification score by about 8 points in average, and by about 12 points in case of the biggest age gaps.

Last but not the least, it is important to highlight that in this manuscript, we have used our GA-cGAN+LMA face editing method exclusively for aging/rejuvenation and gender swapping, but the main advantage of the proposed method is its universality, meaning that it can be potentially applied for modifying of any face attributes.

## 7.2 Limitations and Future Work

The results obtained during this PhD are globally very promising and encouraging. However, in this section, we focus on the limitations of the designed models and approaches, and, at the same time, on the possible solutions to overcome them. This way, we suggest the directions for the future research which have not been explored in the present manuscript. For simplicity, similarly to the previous one, this section is organized into three parts: one per contribution.

**Importance of the Problem Complexity for Feature Learning** The results of the preliminary studies have indicated that the quality of the learned features (which are issued after the CNN training) depends on the target problem. In particular, the more challenging it is, the more expressive and generative are the resulting learned features. Despite this conclusion has been further implicitly confirmed in Chapter 5 (by comparison of gender recognition and age estimation problems, and the fact that the more complicated age estimation problem requires deeper CNN architectures when training is done from scratch), we believe that such an important and general conjecture deserves to be verified via an explicitly dedicated experiment on a more conventional domain than gender/age analysis: for example, on *ImageNet* classification.

More precisely, *ImageNet* is an image dataset organized according to the *WordNet* ontology [Mil95]. For example, in *ImageNet*, a class "poodle dog" is a subclass of the bigger class "dog" which, in turn, is a subclass of the class "domestic animal", etc. This hierarchical structure of *ImageNet* allows to easily control the complexity of the classification problem by varying the number of classes. Indeed, by choosing the level of the class abstraction (*i.e.* many fine-grained classes or few coarse classes), we implicitly set the difficulty of classification while preserving the same number of training images.

We have the following experimental protocol in mind: using a standard CNN architecture (*e.g. ResNet-50*), train $N$ CNNs for classification of the same set of training images but with gradually increasing number of classes. Then, the idea is to compare the *features* learned by these CNNs (without loss of generality, the last but one layer of *ResNet-50* can be used for feature extraction). The most objective way to perform this comparison is transfer learning, in other words, reusing the learned features for other problems in a similar way as it was done by Razavian et al. [SR+14]. We expect the features learned by CNNs with many fine-grained classes to be more general, than the features learned by CNNs with few coarse classes, and to better adapt for other applications.

**Gender/Age Prediction** We also anticipate several possibilities for improvement of our gender and age recognition CNNs.

Thus, a relatively straightforward amelioration consists in a separate processing of the children photos. Indeed, the *IMDB-Wiki_cleaned* dataset (which was used for training of our models) contains very few images of children (*cf.* Figure 5.2.4 in Chapter 5), which results in gender/age CNNs that are not optimized for the youngest age categories. For example, as described in Section 5.4 of Chapter 5, due to the fact that there were many images of children in the ChaLearn Apparent Age Estimation Competition (AAEC), we had decided to train a separate "children" CNN which finally played an important part in our winning solution. Obviously, the same can be done for our best gender recognition and biological age estimation CNNs. More generally, a hierarchical approach for gender/age prediction (*i.e.*, when a

face image is firstly classified in a coarse age category and then, gender/age prediction is done within the selected category by a specialized model) proved to be helpful in several previous studies [Han+15; LYK15], and therefore, is definitely a promising direction to explore.

We have also noticed that our age estimation CNN is quite sensible to changes in face expressions. In particular, our model generally predicts smiling faces to be younger than neutral and frowning ones. In our opinion, this effect is due to a bias in the training dataset where younger people tend to smile more than older ones. More generally, the lack of robustness to face expressions is a known issue in age estimation research [GW12], and there are at least two possibilities to deal with this problem. Firstly, the invariance to face expressions can be learned during the training by integrating a dedicated objective to the loss function. Alternatively, expression normalization can be performed prior to age estimation using, for example, our GA-cGAN+LMA face editing method (for that, the GA-cGAN generative model must be trained with expression conditions).

Otherwise, we believe that the accuracy of our age estimation CNN would be further refined if more training images were available. Indeed, the experiments in Chapter 5 have shown that, unlike gender CNNs, the accuracy of which is completely determined by the face recognition pretraining, the age estimation training takes advantage of deep CNN architectures. Thus, we expect that more training data will help avoiding overfitting and will allow using more complex CNN architectures than *VGG-16* for age estimation (for example, *ResNet-50*).

In Chapter 5, we have also found that a straightforward multi-task learning for gender recognition and age estimation favourably compares with respect to mono-task learnings when the CNNs are trained from scratch. We anticipate that this effect will be even more evident in multi-task learning with $k > 2$ modalities (such as presence of smile, presence of glasses, facial pose etc.) Ideally, we expect that multi-task learning for $k$ various modalities can substitute the need for face recognition pretraining (we have seen that the two transfer learning approaches are interconnected). In order to verify this hypothesis, a big face dataset with the corresponding annotations is needed. For example, the recent CelebA dataset [Liu+15c] containing about 200K face images annotated with 40 binary labels could have been a good candidate for generalizing our work to other facial traits, if it had been also annotated with ages.

In this PhD, we have been working exclusively with static RGB images of faces. At the same time, some works reported that facial visual traits can be better perceived in action. For example, there are two studies [Dib+12; BDB16] which demonstrated that a video of a smiling person adds additional information for gender recognition and age estimation with respect to the sequence of static images. Therefore, exploring of the face dynamics in the context of gender/age prediction seems an interesting possibility for future work. Similarly to video sequences which can potentially provide additional information for gender/age prediction with respect to static images, the alternative modes of image acquisition can also be used to capture some complimentary face details comparing to the conventional digital cameras. For example, there are some works attempting to improve the human demographics prediction by using RGB-depth [HMD12], 3D [HUP09], and even near-infrared [CR11] face images. However, one of the main difficulties of applying deep learning in the mentioned contexts is the lack of large publicly available datasets of faces in the respective formats.

Finally, in this manuscript, we have evaluated our gender recognition and age estimation CNNs on three public benchmarks and on the internal challenging *PBGA* dataset which gives an idea of the

respective gender and age prediction accuracies in average. However, from the applicative point of view, it would be very useful if it was possible to guarantee a certain level of the gender/age prediction confidence given a particular input photo. This can be achieved by evaluating our models on specific sets of face images which are selected according to some predefined criteria reflecting the difficulty of the gender and age analysis. For example, such criteria could be the spatial orientation of the face (e.g. whether it is frontal, half-profile or profile), the resolution of the input image (the corresponding evaluation has been performed in Paragraph 5.3.3.1 of Chapter 5), the presence of face accessories (such as sunglasses, hats or scarfs), the face expression, etc. In order to properly perform this evaluation, a laboratory-collected dataset of faces (balanced between genders and ages) where the same persons are photographed in all respective conditions is needed.

**Face Synthesis and Editing** The most evident limitation of our GA-cGAN generative model and of the resulting GA-cGAN+LMA method for automatic face editing is undoubtedly the fact that they only deal with small images of 64x64 pixels. The reason for that is the prohibitive difficulty of training of a cGAN on images of a bigger resolution. For example, extending the DCGAN architecture [RMC16] (which is used in this manuscript) even to the resolution of 96x96 pixels results in a cGAN which does not converge at all. However, the recently proposed Wasserstein GAN by Arjovsky et al. [ACB17] remedies this problem by significantly stabilizing the GAN convergence which allows using almost arbitrary CNN architecture in a GAN framework. Therefore, it seems promising to train a Wasserstein version of our GA-cGAN.

Another possible workaround for the same problem of a GAN convergence on images of higher resolution is to directly train a generative model for face *editing* instead of *generating* the new faces from scratch. For example, a very recent work by Zhu et al. [Zhu+17] is a good illustration to this idea, as the authors trained their model (Cycle-GAN) to directly transform images from one domain to the other (*e.g.* substituting horses with zebras and vice versa). For example, if Cycle-GAN was applied for face aging, an input domain could be young faces, while the target domain could be senior faces. On the one hand, this transformation approach allows working with images of almost arbitrary resolution, but on the other hand, it loses the universality of our face editing method. Indeed, unlike GA-cGAN+LMA, the approach of Zhu et al. requires training of a separate Cycle-GAN for each target domain (in other words, for each face editing operation: gender swapping, face rejuvenation etc.) Moreover, Cycle-GAN cannot be used for synthesis of random face with the required face attributes.

Otherwise, the proposed GA-cGAN is trained with only six age categories which limits the age precision in the resulting synthesized faces. Therefore, a possible improvement of our model consists in refining these age categories (*i.e.* splitting them into more age classes of smaller sizes) or completely replacing them with real age values. For example, such refined model would be very useful for generating new synthetic datasets with accurate age annotations (which can be further used for age estimation training). However, learning of a GA-cGAN with fine-grained age categories would require a significantly bigger training dataset than *IMDB-Wiki_cleaned* which is used in the present manuscript. Indeed, we have empirically observed that if a certain age is underrepresented during the training, cGAN "learns by heart" the respective examples, and, as a result, always generates the same face for this age.

Finally, it is was shown by a number of studies [Sal+16; OOS16] that the more supervised are cGANs

(*i.e.* the more conditions are used during the training), the better is their convergence. Therefore, extending our GA-cGAN with other face modalities (such as presence of eyeglasses or beard) will not only allow synthesizing and editing of the respective face attributes, but might also improve the visual quality of the resulting synthetic faces. However, similar to the case of gender/age prediction CNNs, exploring this research direction requires a face dataset which is simultaneously annotated with gender, age and other face attributes. We suggest using CelebA dataset for this purpose by automatically annotating it with age labels (for example, this can be done using our age CNN).

## 7.3 List of Publications

The results described in this manuscript were partly presented in a number of peer-reviewed publications as well as invited talks which are listed below.

**International Journals**

1. "Minimalistic CNN-based ensemble model for gender prediction from face images". *G. Antipov*, S.-A. Berrani, J.-L. Dugelay. Pattern Recognition Letters, 2016
2. "Effective training of convolutional neural networks for face-based gender and age prediction". *G. Antipov*, M. Baccouche, S.-A. Berrani, J.-L. Dugelay. Pattern Recognition, 2017.

**International Conferences**

1. "Learned vs. hand-crafted features for pedestrian gender recognition". *G. Antipov*, S.-A. Berrani, N. Ruchaud, J.-L. Dugelay. ACM International Conference on Multimedia 2015.
2. "The impact of privacy protection filters on gender recognition". N. Ruchaud, *G. Antipov*, P. Korshunov, J.-L. Dugelay, T. Ebrahimi, S.-A. Berrani. SPIE International Conference on Optical Engineering, 2015.
3. (*BEST PAPER AWARD, 1st place in Apparent Age Estimation Competition*) "Apparent age estimation from face images combining general and children-specialized deep learning models". *G. Antipov*, M. Baccouche, S.-A. Berrani, J.-L. Dugelay. ChaLearn LAP Workshop at IEEE International Conference on Computer Vision and Pattern Recognition 2016.
4. "Face aging with conditional generative adversarial networks". *G. Antipov*, M. Baccouche, J.-L. Dugelay. IEEE International Conference on Image Processing 2017.
5. "Boosting cross-age face verification via generative age normalization". *G. Antipov*, M. Baccouche, J.-L. Dugelay. IEEE/IAPR International Joint Conference on Biometrics 2017.

**National Conferences, Invited Talks and Presentations**

1. "Learned vs. hand-crafted features for pedestrian gender recognition". Réunion GdR ISIS "Bilan TRECVid 2014 et apprentissage profond", 2015.
2. "Deep learning for gender recognition from faces and bodies". Réunion GdR ISIS "Bilan TRECVid 2015 et apprentissage profond", 2016.

3. (Presented by M. Baccouche) “Apparent age estimation from face images combining general and children-specialized deep learning models”. Reconnaissance de Formes et Intelligence Artificielle : “Apprentissage profond pour la perception et la robotique”, 2016.
4. (*BEST POSTER AWARD*) “Deep learning for estimation of human semantic traits”. Journée des Doctorants d’Orange Labs, 2016.
5. “Deep learning for semantic description of visual human traits”. Journée de la Biométrie. ENSICAEN, 2017.

Chapitre 8

# Résumé Étendu en Français

**Contents**

## 8.1 Introduction Générale

### 8.1.1 Contexte et Motivations

Les concepts du genre et de l'âge sont extrêmement importants dans notre vie quotidienne. En effet, la compréhension de ces concepts par des humains est profonde et intuitive : non seulement nous pouvons immédiatement estimer le genre et l'âge d'un inconnu dans la rue, mais nous sommes aussi souvent capables de reconnaître visuellement des liens parentaux entre un père et un fils, ou des relations familiales entre un frère et une sœur. Une question donc se pose naturellement : l'intelligence artificielle, est-elle capable d'effectuer les mêmes tâches ?

En effet, l'analyse visuelle automatique de l'apparence humaine est un domaine de recherche très demandé aujourd'hui. Le développement inédit des caméras digitales et de l'Internet ont augmenté de manière significative le nombre de photos qui sont constamment crées et partagées entre les gens partout dans le monde. Par conséquent, l'indexation et l'organisation de telles quantités d'images de visages digitales ne sont possibles qu'avec des solutions efficaces pour la reconnaissance et la description sémantiques de visages humains.

Récemment, une famille de méthodes d'apprentissage automatique qui est basée sur les réseaux de neurones artificiels profonds s'est avérée être particulièrement adaptée aux larges quantités de données d'apprentissage et peut se généraliser remarquablement mieux que les approches alternatives. Cette famille est connue aujourd'hui sous le nom "d'apprentissage profond" ("deep learning" en anglais).

Durant les dernières années, l'apprentissage profond a battu plusieurs records dans les différents domaines de l'intelligence artificielle [Hin+12 ; SVL14 ; Xio+15]. Cependant, le progrès le plus incontestable a été constaté dans le domaine de la vision par ordinateur grâce aux réseaux de neurones convolutifs (CNNs) [LeC+89]. Aujourd'hui, quasiment toutes les solutions de l'état de l'art pour la classification d'images et de vidéos [Kar+14], la segmentation d'images [Luc+16], la restauration et la super-résolution [RIM17], la reconnaissance optique des caractères [Iba+14] et la reconnaissance faciale [SKP15] sont basées sur des CNNs.

Ainsi, la présente thèse est dédiée à l'exploration et à la conception des méthodes d'apprentissage profond dans le cadre de l'analyse du genre et de l'âge dans les images de visages.

### 8.1.2 Objectifs

Dans cette thèse, nous nous sommes particulièrement intéressés à deux objectifs complémentaires, à savoir : (1) la reconnaissance du genre et de l'âge à partir d'images de visages, et (2) la synthèse et l'édition du genre et de l'âge dans des images de visages.

#### 8.1.2.1 Prédiction du Genre et de l'Âge

Le premier objectif constitue un problème de description du visage (à l'entrée nous avons une image et il faut trouver son annotation). Plus particulièrement, par la prédiction du genre, nous entendons la classification binaire avec deux classes (femmes et hommes), alors que l'estimation de l'âge constitue un problème de régression où une valeur réelle (l'âge de la personne) est attendue en sortie.

Les méthodes classiques pour la classification du genre et l'estimation de l'âge suivent un procédé typique consistant en deux étapes : l'extraction des descripteurs d'image de visage en question (souvent sous forme d'un vecteur), et ensuite, la classification (ou bien la régression) à partir de ces descripteurs pour obtenir le résultat final. Dans ce schéma, l'étape d'extraction des descripteurs a une importance primordiale car elle définit quelles informations sur le visage en question sont données à l'entrée de la méthode de classification (ou de régression). Ainsi, dans l'état de l'art, nous pouvons globalement distinguer quatre familles de descripteurs de visages qui ont été utilisées pour les problèmes en question et qui sont listées ci-dessous :

— Les *descripteurs anthropométriques* [PBP92 ; Fel97] constituent un ensemble de distances et de proportions géométriques qui sont calculées entre les points de repères de visages.

— Les *descripteurs basés sur la texture* [YA07 ; XSL08] se focalisent sur les motifs récurrents dans les images de visages.

— Les *descripteurs appris via une variété* [HP09 ; GM11] sont obtenus en projetant les images de visages sur une variété où les genres et les âges sont facilement distinguables.

— Les *descripteurs basés sur l'apparence* [LTC02 ; GZSM07] combinent à la fois les informations extraites de la texture et de la forme des visages.

Contrairement aux travaux précédents, nous fusionnons les étapes d'extraction des descripteurs et de prédiction du genre et de l'âge dans un modèle en employant les CNNs. Plus particulièrement, notre objectif est de trouver des méthodes optimales pour la conception et l'apprentissage de CNNs pour ces deux problèmes.

#### 8.1.2.2 Synthèse et Édition du Genre et de l'Âge

Le deuxième objectif de la thèse est complémentaire au premier, car pour la synthèse et l'édition d'images de visages, à l'entrée, nous avons les descriptions sémantiques ciblées (*i.e.* le genre et l'âge), et à la sortie, nous devons générer une image de visage qui correspond à ces descriptions.

L'édition de visage est un problème plus complexe que la synthèse puisqu'il exige le fait que l'image synthétisée doit correspondre à l'image donnée à l'exception des paramètres modifiées. Par exemple, dans le cadre de l'édition de l'âge (*i.e.* vieillissement / rajeunissement), nous devons changer l'apparence de l'âge de la personne mais préserver tous les autres paramètres du visage. De même, le changement de genre ne doit pas affecter l'âge de la personne, son expression faciale, etc.

Les travaux précédents se focalisent davantage sur le problème d'édition de visage. Notamment, les deux familles d'approches suivantes prévalent dans l'état de l'art :

— Les approchent qui effectuent le vieillissement / rajeunissement ou le changement de genre via la *modélisation paramétrique* des différentes parties du visage (*i.e.* les muscles, le crâne, la peau, etc.) [Suo+10 ; Shi+12].

— Les approches qui se basent sur la conception des *prototypes* moyennés pour chaque genre ou pour chaque catégorie d'âge [TBP01 ; KSSS14]. L'édition est ensuite effectuée avec la projection de l'image initiale sur les prototypes pré-calculés.

Dans ce travail, notre objectif est de proposer une approche neuronale générique qui est applicable pour la synthèse des images de visages aléatoires avec le genre et l'âge cibles, mais qui peut être également adaptée pour l'édition d'un visage donné. Pour cela nous nous focalisons sur les modèles neuronaux génératifs [Goo+15] qui ont connu un progrès significatif récemment.

### 8.1.3 Contributions et Organisation de la Thèse

Les trois contributions principales de cette thèse sont présentées dans trois sections séparées.

D'abord, dans la Section 8.2, nous effectuons deux études préliminaires qui illustrent la caractéristique très importante des CNNs. En particulier, nous démontrons que l'efficacité des descripteurs appris avec un CNN dépend non seulement de la taille de l'architecture employée, et de la quantité de données d'apprentissage, mais aussi de la complexité du problème cible. Cette conclusion est prise en compte plus tard dans la thèse, pendant la conception de CNNs pour la classification du genre et l'estimation de l'âge.

La Section 8.3 répond au premier objectif de la thèse. Plus particulièrement, nous proposons une stratégie d'apprentissage de CNNs pour la classification du genre et de l'âge qui est validée empiriquement. Elle nous permet de concevoir les CNNs qui surpassent l'état de l'art actuel sur les problèmes en question.

Le deuxième objectif de la thèse est ensuite abordé dans la Section 8.4. Nous proposons un modèle génératif pour la synthèse d'image de visage conditionnée sur le genre et l'âge. Ensuite, nous présentons une méthode permettant d'appliquer ce modèle pour l'édition d'images existantes. Afin de démontrer l'utilité pratique de notre approche, nous l'appliquons à la normalisation des âges de visages ce qui permet d'améliorer les performances d'un moteur de vérification faciale.

Finalement, la Section 8.5 dresse un bilan des résultats de cette thèse et présente des pistes pour les futurs travaux.

## 8.2 Études Préliminaires

### 8.2.1 Introduction

La présente section est dédiée à deux études préliminaires distinctes ayant pour but de mieux comprendre le fonctionnement des CNNs, qui sont les principaux modèles utilisés pour la classification du genre et l'estimation de l'âge dans la Section 8.3.

La première étude présentée dans la Sous-section 8.2.2 porte autour de la comparaison entre deux types de descripteurs d'images : les descripteurs manuellement conçus (*i.e.* ceux qui sont extraits via un algorithme prédéfini par des humains selon une expertise liée à un domaine particulier) et les descripteurs appris automatiquement par des CNNs.

Le but de la deuxième étude résumée dans la Sous-section 8.2.3 est d'évaluer le rapport entre la complexité de l'architecture de CNN (*i.e.* le nombre de paramètres, la profondeur, etc.) utilisés pour résoudre un certain problème et les performances finales. Nous nous sommes particulièrement focalisés sur l'apprentissage de CNNs à partir de zéro (sans utiliser de modèles pré-entrainés) ce qui nous permet d'évaluer l'impact du problème cible sur l'efficacité de l'apprentissage.

### 8.2.2 Étude 1 : Descripteurs Appris par des CNNs vs. Descripteurs Manuellement Conçus

Malgré le fait que dans cette thèse nous nous soyons principalement intéressés à l'analyse d'images de visages, nous avons choisi le problème de la classification du genre à partir d'*images de piétons* afin d'effectuer la comparaison entre les descripteurs manuellement conçus et les descripteurs appris par les CNNs dans le cadre de la première étude préliminaire. Ce choix est dicté par le fait que la classification du genre est plus compliquée quand l'estimation est faite à partir d'images de piétons par rapport à l'estimation à partir d'images de visages. En effet, la comparaison des descripteurs sur un problème complexe est plus représentative puisque dans ce cas là, la qualité des représentations d'images peut vraiment faire la différence.

#### 8.2.2.1 Descripteurs Comparés

Dans le cadre de cette étude, six descripteurs d'images de piétons ont été choisis pour la comparaison.

Nous avons notamment sélectionné les **trois descripteurs manuellement conçus** qui sont pertinents pour la représentation d'images de piétons et la classification du genre. Ces descripteurs sont listés ci-dessous :

— Les descripteurs Re-Identification (RI) [Lay+12] ont été choisis en tant que descripteurs explicitement conçus pour l'identification de piétons.

— Les descripteurs Local Binary Pattern (LBP) [OPH96] sont souvent utilisés pour la classification du genre.

— Enfin, les descripteurs Histogram of Oriented Gradients (HOG) sont les plus adaptés, selon les travaux précédents [Col+09 ; BMM11 ; GS+16], à la classification du genre à partir d'images de piétons.

Dans notre implémentation, une image de piéton de taille 150x50 pixels est représentée par un vecteur de 2500 ou 3000 valeurs réelles selon les descripteurs manuellement conçus utilisés.

Les **trois descripteurs appris** sont obtenus avec les activations issues des avant dernières couches des trois CNNs présentés ci-dessous :

— Le premier réseau est nommé *pedestrian_CNN* et son architecture est largement inspirée par le fameux *LeNet-5* [LeC+98] (*cf.* la première colonne du Tableau 8.2.1). L'architecture de *pedestrian_CNN* étant assez compacte, elle est particulièrement adaptée à l'apprentissage sur des corpus de petite taille (ce qui est le cas dans la présente étude). Les descripteurs extraits avec *pedestrian_CNN* sont des vecteurs réels de taille 100.

— Nous avons également comparé les descripteurs universels d'apprentissage profond qui sont issus de l'avant dernière couche d'*AlexNet* [KSH12] (4096 valeurs réelles, *cf.* la deuxième colonne du Tableau 8.2.1). *AlexNet* a été pré-entrainé sur ImageNet pour la classification multi-objets [Rus+15].

— Afin d'évaluer l'importance de l'adaptation au domaine cible et de l'efficacité du transfer learning, nous avons affiné *AlexNet* pour la classification du genre à partir d'images de piétons. L'extraction

| *pedestrian_CNN* | *AlexNet* |
|---|---|
| Entrée : 150x50 | Entrée : 227x227 |
| Conv : 100@5x5 | Conv : 96@11x11 |
| MaxPool : 2x2 | |
| Conv : 100@5x5 | Conv : 256@5x5 |
| MaxPool : 2x2 | |
| — | Conv : 384@3x3 |
| — | Conv : 384@3x3 |
| — | Conv : 256@3x3 |
| — | MaxPool : 2x2 |
| — | FC : 4096 |
| **FC : 100** | **FC : 4096** |
| Softmax : 2 | |

TABLE 8.2.1 – Les architectures de CNNs utilisées pour l'extraction des descripteurs appris. Les couches dont les activations constituent les descripteurs sont mises en gras. "Conv : N@MxM" désigne une couche convolutionnelle avec N noyaux de convolution de taille MxM. "MaxPool : MxM" désigne un sous-échantillonnage de type "Max-Pooling" par un facteur M. "FC : N" désigne une couche entièrement connectée avec N neurones.

des descripteurs appris avec ce réseau affiné (le réseau est appelé *AlexNet-FT* dans la suite de cette section) est effectuée de la même façon que pour *AlexNet* classique.

#### 8.2.2.2 La Collection *PETA* et le Protocole Expérimental

(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

FIGURE 8.2.1 – Exemples d'images de piétons issus des différents corpus de la collection *PETA* : (a) *CUHK* ; (b) *PRID* ; (c) *GRID* ; (d) *MIT* ; (e) *VIPeR* ; (f) *3DPeS* ; (g) *CAVIAR* ; (h) *i-LIDS* ; (i) *SARC3D* ; (j) *TownCentre*. Les proportions originales des images sont préservées.

Pour nos expérimentations, nous avons utilisé la collection *PETA_cleaned* (un sous-ensemble de *PETA* [Den+14] nettoyé manuellement) qui est à notre connaissance la plus large source d'images de piétons en accès libre aujourd'hui (8,635 images). *PETA_cleaned* est composé de dix corpus qui ont été récoltés indépendamment les uns des autres ce qui assure une grande variété entre les images de *PETA_cleaned* en termes de poses, résolutions, fonds, etc. (*cf.* la Figure 8.2.1).

La comparaison des six descripteurs en question est effectuée sur les trois expérimentations suivantes de la complexité croissante.

**L'expérimentation 1** constitue les conditions les plus favorables pour les modèles de classification du genre car ces derniers sont séparément entrainés et évalués sur les cinq plus grands corpus de *PETA_cleaned* (*CUHK*, *PRID*, *GRID*, *MIT* et *VIPeR*). Étant donné que la variation entre les images d'un même corpus est a priori moins élevée qu'entre les images issues de corpus différents, le protocole de l'expérimentation 1 peut être résumé en "données d'apprentissage homogènes, évaluation intra-corpus".

**L'expérimentation 2** augmente la difficulté de la classification du genre puisque cette fois-ci, les modèles (et les descripteurs appris par CNNs) sont entrainés et évalués sur le mélange des cinq plus grands corpus utilisés dans l'expérimentation 1. Par conséquence, ce protocole expérimental peut être résumé en "données d'apprentissage hétérogènes, évaluation intra-corpus".

**L'expérimentation 3** est la plus proche des conditions réelles d'utilisation d'un système de classification du genre et représente donc un véritable défi pour les descripteurs d'images de piétons en comparaison. Dans ce cas-là, les modèles et les descripteurs appris sont entrainés sur le même mélange des cinq plus grands corpus que dans l'expérimentation 2. En revanche, l'évaluation est effectuée sur le mélange de cinq autres corpus de *PETA_cleaned* (*3DPeS*, *CAVIAR*, *i-LIDS*, *SARC3D* et *TC*) qui n'ont pas été utilisés dans les deux premières expérimentations. L'expérimentation 3 se résume donc en "données d'apprentissage hétérogènes, évaluation inter-corpus".

#### 8.2.2.3 Résultats

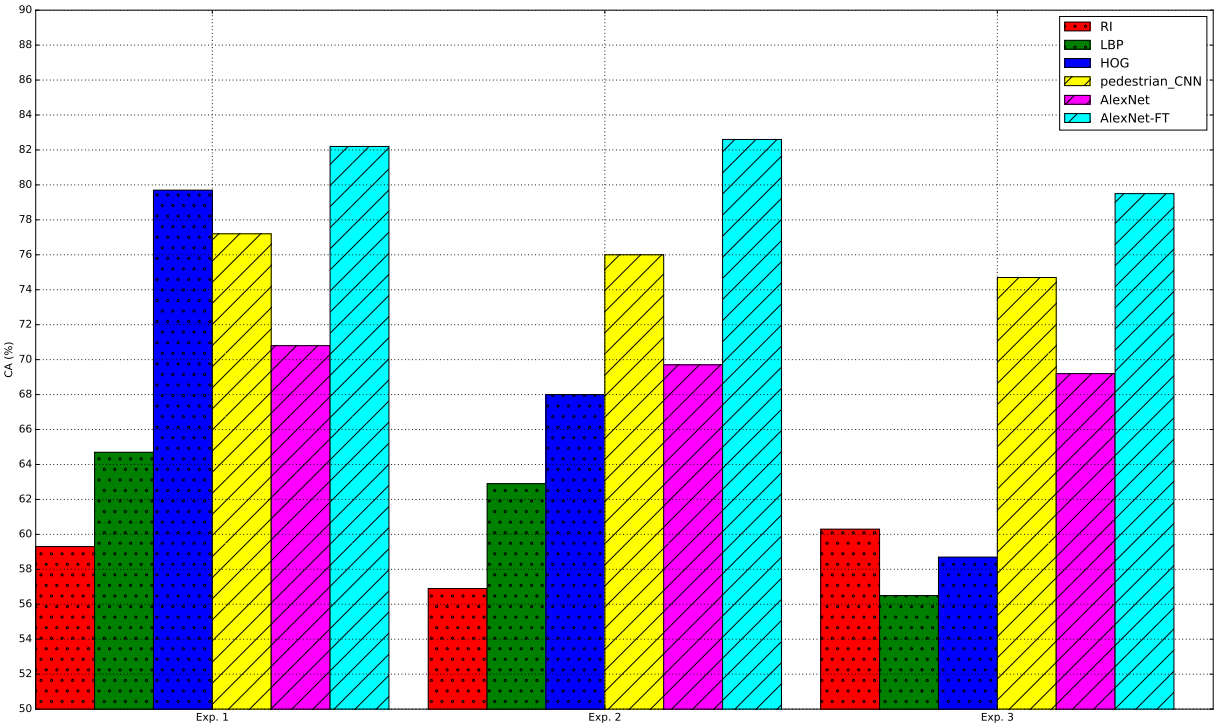



FIGURE 8.2.2 – Descripteurs manuellement conçus vs. descripteurs appris. Exp. 1 : données d'apprentissage homogènes, évaluation intra-corpus. Exp. 2 : données d'apprentissage hétérogènes, évaluation intra-corpus. Exp. 3 : données d'apprentissage hétérogènes, évaluation inter-corpus.

Afin d'assurer l'objectivité de la comparaison, nous utilisons la même méthode de classification pour les six descripteurs, à savoir : SVM linéaire [CV95]. Les taux de classification (CA) du genre obtenus par SVM pour les six descripteurs dans le cadre des trois expérimentations présentées au préalable sont

illustrés sur la Figure 8.2.2.

Comme nous pouvons le remarquer dans la Figure 8.2.2, les descripteurs manuellement conçus (notamment les descripteurs *HOG*) concurrencent les descripteurs appris par les CNNs seulement dans les conditions très limitées de l'expérimentation 1. Les expérimentations 2 et 3 démontrent que les descripteurs *HOG* n'arrivent pas à s'adapter aux données d'apprentissage hétérogènes et surtout à l'évaluation inter-corpus. Cela n'est pas le cas pour les descripteurs appris par les CNNs car ils obtiennent des résultats similaires dans les trois expérimentations, ce qui prouve leur aptitude à s'adapter à des données variées et à se généraliser face à des données inconnues. Finalement, il est aussi important de souligner que les descripteurs *AlexNet-FT* surpassent nettement les descripteurs *AlexNet* ce qui démontre une plus grande efficacité d'adaptation des descripteurs au domaine cible.

### 8.2.3 Étude 2 : L'Architecture de CNN pour l'Apprentissage à Partir de Zéro

Dans la deuxième étude préliminaire, nous nous intéressons à l'évaluation de l'influence de la taille de l'architecture du CNN sur la précision de la classification du genre à partir d'images de visages. Pour cela, nous commençons par entrainer un CNN à l'architecture assez profonde et complexe (qui est nommé *start_CNN*), puis, nous simplifions cette architecture de façon progressive sans sacrifier sa précision. L'architecture de *start_CNN* est présentée dans la première colonne du Tableau 8.2.2.

#### 8.2.3.1 Algorithme pour l'Optimisation de l'Architecture du CNN

Afin d'optimiser *start_CNN* pour la classification du genre à partir d'images de visages, nous proposons l'Algorithme 8.1 qui est simple mais efficace. L'Algorithme 8.1 débute par l'évaluation du CA de *start_CNN* (qui est désigné comme $CA_0$). Ensuite, l'optimisation est faite en trois étapes qui se suivent de façon successive : (1) la minimisation de la profondeur de l'architecture (*i.e.* le nombre de couches cachées), (2) la minimisation de la largeur de l'architecture (*i.e.* le nombre de noyaux de convolution), et enfin (3) la minimisation du nombre de neurones dans la couche entièrement connectée. Dans l'Algorithme 8.1, les performances de toutes les architectures intermédiaires sont comparées à $CA_0$, et à chaque étape, l'optimisation s'arrête dès que le CA du CNN en question est dégradé par rapport à $CA_0$.

#### 8.2.3.2 Expérimentations et Résultats

Nous avons utilisé le corpus CASIA WebFace [Yi+14] pour nos expérimentations dans le cadre de la deuxième étude préliminaire. Parmi environ 500K images de ce corpus, 450K images ont été utilisées pour l'apprentissage et les 50K autres images ont constitué le corpus de validation.

Dans le Tableau 8.2.2, nous illustrons le processus d'optimisation de *start_CNN* selon l'Algorithme 8.1. Les architectures qui ont été sélectionnées après chaque étape sont soulignées en gras. Le CNN *I* est la sortie finale de l'algorithme. Ce réseau est 10 fois plus rapide et prend 15 fois moins d'espace mémoire que *start_CNN*. Nous avons également remarqué que CNN *I* nécessite seulement 50% des données d'apprentissage disponible (*i.e.* de 450K images de visages) pour arriver à ces performances maximales.

Ainsi, les performances du CNN appris à partir de zéro pour la classification du genre à partir d'images de visages sont saturées avec l'architecture de seulement 4 couches convolutionelles et 225K

**input :** L'architecture initiale start_CNN ; les corpus d'apprentissage et de validation
**output:** L'architecture optimisée optimized_CNN

```
1  train (start_CNN);
2  CA_0 := evaluate_CA (start_CNN);
   /* Étape 1. Optimisation du nombre de couches convolutionelles
      et de la taille de la rétine.                                  */
3  current_CNN := start_CNN ;
4  repeat
5  |  current_optimal_CNN := current_CNN ;
6  |  current_CNN := remove_ConvBlock (current_CNN);
7  |  train (current_CNN);
8  |  CA_current := evaluate_CA (current_CNN);
9  until criterion (CA_0,CA_current) or current_CNN .#ConvBlocks < 2;
   /* Étape 2. Optimisation du nombre de noyaux de convolution.      */
10 current_CNN := current_optimal_CNN ;
11 repeat
12 |  current_optimal_CNN := current_CNN ;
13 |  current_CNN := half_CNNWidth (current_CNN);
14 |  train (current_CNN);
15 |  CA_current := evaluate_CA (current_CNN);
16 until criterion (CA_0,CA_current) or current_CNN .#ConvFeatureMaps < 2;
   /* Step 3. Optimisation du nombre de neurones dans la couche
      entièrement connectée.                                         */
17 current_CNN := current_optimal_CNN ;
18 repeat
19 |  current_optimal_CNN := current_CNN ;
20 |  current_CNN := half_FCNeurons (current_CNN);
21 |  train (current_CNN);
22 |  CA_current := evaluate_CA (current_CNN);
23 until criterion (CA_0,CA_current);
24 optimized_CNN := current_optimal_CNN ;
```

**Algorithm 8.1:** Optimisation de l'architecture du CNN. L'algorithme suppose que l'architecture initiale de *start_CNN* est composée de plusieurs blocs de couches convolutionelles suivis par des sous-échantillonnages du facteur de 2. Le nombre de noyaux de convolution (#*ConvFeatureMaps*) dans chaque couche convolutionelle ainsi que le nombre de neurones dans la couche entièrement connectée (#*FCNeurons*) est supposé être une puissance de 2.

images d'apprentissage.

### 8.2.4 Conclusion

Dans cette section, nous avons effectué deux études préliminaires.

D'abord, nous avons comparé les capacités des descripteurs manuellement conçus et des descripteurs appris par des CNNs (1) à s'adapter à l'hétérogénéité des données d'apprentissage, et (2) à se généraliser face aux données inconnues. Le premier de ces points est important dans un contexte où les corpus d'apprentissage contemporains proviennent souvent de sources très variées sur Internet, alors que le

| ***start_CNN*** | Étape 1 | | | Étape 2 | Étape 3 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *A* | ***B*** | *C* | *D* | *E* | *F* | *G* | *H* | ***I*** | *J* | *K* | *L* | *M* |
| Entrée : 128x128 | Entrée : 64x64 | Entrée : 32x32 | Entrée : 16x16 | Entrée : 32x32 | Entrée : 32x32 | | | | | | | | |
| Conv : 32@3x3<br>Conv : 32@3x3 | Conv : 32@3x3<br>Conv : 32@3x3 | Conv : 32@3x3<br>Conv : 32@3x3 | Conv : 64@3x3<br>Conv : 64@3x3 | Conv : 16@3x3<br>Conv : 16@3x3 | Conv : 32@3x3<br>Conv : 32@3x3 | | | | | | | | |
| MaxPool : 2x2 | Max-Pool : 2x2 | Max-Pool : 2x2 | Max-Pool : 2x2 | Max-Pool : 2x2 | MaxPool : 2x2 | | | | | | | | |
| Conv : 32@3x3<br>Conv : 32@3x3 | Conv : 32@3x3<br>Conv : 32@3x3 | Conv : 64@3x3<br>Conv : 64@3x3 | — | Conv : 32@3x3<br>Conv : 32@3x3 | Conv : 64@3x3<br>Conv : 64@3x3 | | | | | | | | |
| MaxPool : 2x2 | Max-Pool : 2x2 | Max-Pool : 2x2 | — | Max-Pool : 2x2 | MaxPool : 2x2 | | | | | | | | |
| Conv : 64@3x3<br>Conv : 64@3x3 | Conv : 64@3x3<br>Conv : 64@3x3 | — | — | — | — | | | | | | | | |
| MaxPool : 2x2 | Max-Pool : 2x2 | — | — | — | — | | | | | | | | |
| Conv : 64@3x3<br>Conv : 64@3x3 | — | — | — | — | — | | | | | | | | |
| MaxPool : 2x2 | — | — | — | — | — | | | | | | | | |
| FC : 512 | | | | | FC : 256 | FC : 128 | FC : 64 | FC : 32 | FC : 16 | FC : 8 | FC : 4 | FC : 2 | — |
| Softmax : 2 | | | | | | | | | | | | | |

TABLE 8.2.2 – Optimisation de l'architecture de *start_CNN* par l'Algorithme 8.1. "Conv : N@MxM" désigne une couche convolutionnelle avec N noyaux de convolution de taille MxM. "MaxPool : MxM" désigne un sous-échantillonnage de type "Max-Pooling" par un facteur M. "FC : N" désigne une couche entièrement connectée avec N neurones.

deuxième est tout simplement essentiel pour pouvoir utiliser les modèles d'apprentissage automatique dans des applications réelles. Cette étude a illustré de façon claire les avantages des descripteurs appris par les CNNs par rapport aux descripteurs manuellement conçus, et confirme d'autant plus le choix de l'apprentissage profond pour la suite de la thèse.

Ensuite, dans le cadre de la deuxième étude préliminaire, nous avons remarqué que les performances des CNNs appris à partir de zéro pour la classification du genre étaient saturées avec une architecture assez légère. Nous soupçonnons que cela est lié à la simplicité relative du problème de la classification du genre par rapport à d'autres problèmes (comme la classification ImageNet [Rus+15], par exemple) où la complexité des réseaux de neurones a permis d'apprendre des descripteurs plus discriminants. Cette supposition est confirmée par la suite dans la Section 8.3.

## 8.3 Prédiction du Genre et de l'Âge à Partir d'Images de Visages

### 8.3.1 Introduction

La présente section se focalise autour de l'une des deux principales contributions de cette thèse, à savoir la conception de CNNs performants pour la classification du genre et l'estimation de l'âge.

Tout d'abord, nous identifions et comparons les principaux axes de variations entre les approches existantes sur la prédiction du genre et de l'âge par les CNNs. Cela nous permet de formuler des principes de conception et d'apprentissage efficaces de CNNs pour les tâches dont il est question dans la Sous-section 8.3.2.

Ensuite, nous utilisons ces principes pour apprendre les CNNs profonds qui dépassent l'état de l'art sur la classification du genre et l'estimation de l'âge biologique dans la Sous-section 8.3.3.

Enfin, nous démontrons que notre meilleur CNN pour l'estimation de l'âge biologique peut être facilement adapté à l'estimation de l'âge apparent. Ainsi, nous détaillons la solution qui nous a permis de gagner la compétition internationale sur l'estimation de l'âge apparent dans la Sous-section 8.3.4.

### 8.3.2 Conception de CNN et Stratégie d'Apprentissage

Après avoir analysé l'état de l'art sur la prédiction du genre et de l'âge par des CNNs, nous avons identifié les quatre principaux paramètres de conception d'architecture et d'apprentissage de CNNs qui distinguent les études existantes, à savoir : (1) l'encodage de la sortie d'âge dans des CNN de l'estimation de l'âge, (2) le cadrage du visage à l'entrée de CNN, (3) la profondeur de CNN, et (4) la stratégie d'apprentissage utilisée.

#### 8.3.2.1 Paramètres Étudiés

<table>
<tr><td rowspan="2">Paramètres</td><td colspan="2">Valeurs Testées</td></tr>
<tr><td>CNN du Genre</td><td>CNN de l'Âge</td></tr>
<tr><td rowspan="3">Encodage de l'Âge</td><td rowspan="3">N/A</td><td>0/1-CAE</td></tr>
<tr><td>RVAE</td></tr>
<tr><td>LDAE</td></tr>
<tr><td rowspan="2">Cadrage du Visage</td><td colspan="2">“face-only”</td></tr>
<tr><td colspan="2">“face+40%”</td></tr>
<tr><td rowspan="4">Profondeur du CNN</td><td colspan="2">2 couches convolutionnelles</td></tr>
<tr><td colspan="2">4 couches convolutionnelles</td></tr>
<tr><td colspan="2">6 couches convolutionnelles</td></tr>
<tr><td colspan="2">8 couches convolutionnelles</td></tr>
<tr><td rowspan="4">Pré-Apprentissage / Apprentissage Multi-Tâches</td><td colspan="2">Pas de pré-apprentissage, mono-tâche</td></tr>
<tr><td colspan="2">Pré-entrainé pour FR, mono-tâche</td></tr>
<tr><td colspan="2">Pas de pré-apprentissage, multi-tâches</td></tr>
<tr><td colspan="2">Pré-entrainé pour FR, multi-tâches</td></tr>
</table>

TABLE 8.3.1 – Les paramètres de conception d'architecture et d'apprentissage de CNN pour la prédiction du genre et de l'âge comparés dans la Sous-section 8.3.2. FR = Reconnaissance Faciale.

Afin de choisir des valeurs optimales pour chacun de ces paramètres, nous effectuons les expérimentations dans des conditions contrôlées.

Plus particulièrement, nous comparons les paramètres d'architecture et les stratégies d'apprentissage de CNNs qui sont résumés dans le Tableau 8.3.1 et détaillés ci-dessous :

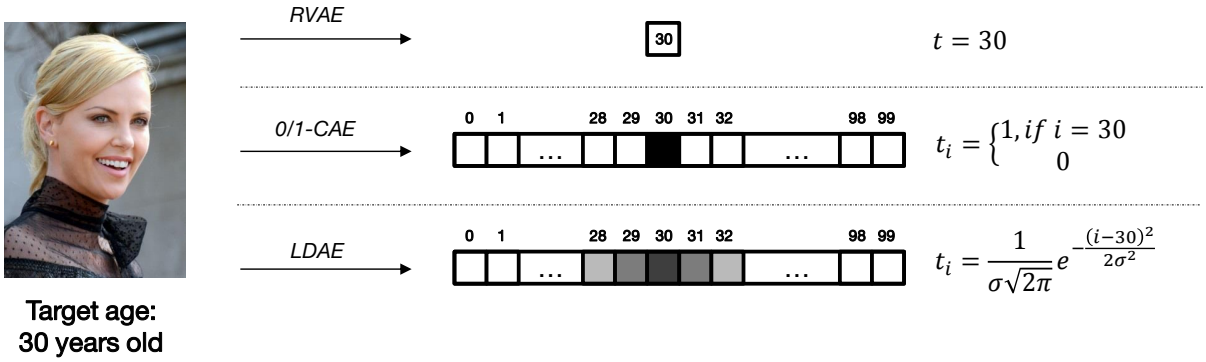



FIGURE 8.3.1 – Exemple de l'encodage de l'âge avec les trois approches comparées dans la Sous-section 8.3.2. $t$ désigne les encodages. $\sigma$ est un hyper-paramètre LDAE (nous utilisons $\sigma = 2.5$).

(a) (b) (c)

FIGURE 8.3.2 – Cadrages d'images de visages comparés dans la Sous-section 8.3.2. (a) Image initiale. (b) Cadrage "face-only". (c) Cadrage "face+40%".

— Dans un premier temps, nous choisissons **l'encodage optimal de l'âge** pendant l'apprentissage. En effet, ce choix (qui est trivial dans le cas de la classification du genre) joue un rôle important pour les CNNs de l'estimation de l'âge en définissant la fonction de coût utilisée pour l'apprentissage. Ainsi, nous comparons les trois méthodes d'encodage les plus utilisées dans l'état de l'art, à savoir : 0/1-CAE, RVAE et LDAE (*cf.* la Figure 8.3.1). Les encodages Real-Valued Age Encoding (RVAE) [Liu+15a] et 0/1-Classification Age Encoding (0/1-CAE) [RTVG16] constituent deux approches opposées, car RVAE encode un âge en tant que valeur réelle unique (ce qui transforme le CNN en un modèle de régression), alors que 0/1-CAE encode un âge avec un vecteur binaire (ce qui transforme le CNN en un modèle de classification). Le Label Distributed Age Encoding (LDAE) [GYZ13] est un mélange des deux méthodes présentées auparavant puisque, comme cela est illustré dans la Figure 8.3.1, l'encodage LDAE est un vecteur de valeurs réelles (ce qui transforme le CNN en un modèle de classification "douce").

— Le **Cadrage du visage** détermine quelles parties du visage sont données à l'entrée du CNN. Nous examinons les deux façons alternatives de découper les images de visage en carré (ce qui est nécessaire pour pouvoir utiliser les architectures de CNN les plus récentes) qui sont présentées dans la Figure 8.3.2. Le cadrage "face-only" se focalise seulement sur la partie centrale du visage, alors que le cadrage "face+40%" inclut aussi les parties périphériques (comme les cheveux et les oreilles) avec le risque d'intégrer occasionnellement un peu le fond de l'image.

— Nous évaluons également l'influence de **la profondeur du CNN** sur la précision de la classification du genre et l'estimation de l'âge. Nous nous focalisons notamment uniquement sur le nombre de couches convolutionnelles comme cela a été conseillé dans plusieurs travaux précédents [LCY14; Sze+15].

— L'**utilisation du pré-apprentissage** et l'apprentissage simultané pour plusieurs tâches (*i.e.* **l'apprentissage multi-tâches**) peuvent être considérés comme deux approches alternatives de *transfer learning* [PY10]. En effet, dans les deux cas, l'objectif est d'utiliser les connaissances apprises pour une autre tâche afin d'enrichir les descripteurs appris pour une tâche en question. Pour cette raison, nous évaluons ces deux stratégies d'apprentissage de CNN ensemble.

#### 8.3.2.2 Protocole Expérimental

| *fast_CNN_2* | *fast_CNN_4* | *fast_CNN_6* | *fast_CNN_8* |
|---|---|---|---|
| Taille de la rétine : 32x32 pour "face-only", 64x64 pour "face+40%" | | | |
| Conv1_1 : 32@3x3 | Conv1_1 : 32@3x3 | Conv1_1 : 32@3x3 | Conv1_1 : 32@3x3 |
| — | Conv1_2 : 32@3x3 | Conv1_2 : 32@3x3 | Conv1_2 : 32@3x3 |
| — | — | Conv1_3 : 32@3x3 | Conv1_3 : 32@3x3 |
| — | — | — | Conv1_4 : 32@3x3 |
| MaxPool : 2x2 | MaxPool : 2x2 | MaxPool : 2x2 | MaxPool : 2x2 |
| Conv2_1 : 32@3x3 | Conv2_1 : 32@3x3 | Conv2_1 : 32@3x3 | Conv2_1 : 32@3x3 |
| — | Conv2_2 : 32@3x3 | Conv2_2 : 32@3x3 | Conv2_2 : 32@3x3 |
| — | — | Conv2_3 : 32@3x3 | Conv2_3 : 32@3x3 |
| — | — | — | Conv2_4 : 32@3x3 |
| MaxPool : 2x2 | MaxPool : 2x2 | MaxPool : 2x2 | MaxPool : 2x2 |
| FC : 512 | FC : 512 | FC : 512 | FC : 512 |
| Couche de sortie | | | |

TABLE 8.3.2 – Les architectures de CNN utilisées pour la comparaison des paramètres dans la Sous-section 8.3.2. "Conv : N@MxM" désigne une couche convolutionnelle avec N noyaux de convolution de taille MxM. "MaxPool : MxM" désigne un sous-échantillonnage de type "Max-Pooling" par un facteur M. "FC : N" désigne une couche entièrement connectée avec N neurones.

Afin d'effectuer une comparaison qui soit la plus objective possible, nous avons employé la même architecture simple (présentée dans le Tableau 8.3.2) dans toutes nos expérimentations. Les paramètres comparés (sauf la profondeur du CNN) sont ainsi évalués avec l'architecture *fast_CNN_4* détaillée dans la colonne 2 du Tableau 8.3.2 (les autres architectures du Tableau 8.3.2 servent uniquement pour l'expérimentation sur la profondeur). Les expérimentations se suivent dans l'ordre du Tableau 8.3.1, et, sauf indication contraire, les paramètres optimaux qui ont été retenus lors de chaque étape de l'expérimentation sont réutilisés dans les étapes suivantes.

L'apprentissage de tous les CNNs est effectué sur le corpus *IMDB-Wiki_cleaned* (la version nettoyée du corpus *IMDB-Wiki*) contenant 250K images de visages de célébrités. À notre connaissance, il s'agit du plus grand corpus de visages en libre accès annoté à la fois avec le genre et l'âge. Ensuite, l'évaluation et la comparaison des résultats sont faites sur le corpus interne de non-célébrités *PBGA* contenants 3540 images de visages. Collecté manuellement, ce corpus est parfaitement équilibré entre les deux genres et les âges de 12 et à 70 ans.

#### 8.3.2.3 Résultats

Les résultats de nos expérimentations sur la conception d'architecture de CNN et sur la stratégie d'apprentissage pour les deux tâches en question peuvent être résumés dans les cinq conclusions qui sont présentées ci-dessous :

1. Le LDAE est l'approche optimale pour l'encodage de l'âge.
2. Le cadrage "face+40%" est mieux adapté aux CNN de la classification du genre et l'estimation de l'âge que le cadrage "face-only".
3. L'estimation de l'âge est un problème plus complexe que celui de la classification du genre, mais dans les deux cas, le transfer learning s'est avéré très utile pour améliorer la convergence de CNNs.
4. Le pré-apprentissage pour la reconnaissance faciale est particulièrement efficace comme technique de transfer learning dans le cadre des deux problèmes étudiés.
5. Le pré-apprentissage pour la reconnaissance faciale englobe les effets positifs de l'apprentissage multi-tâches pour la prédiction du genre et de l'âge. Ces deux approches de transfer learning ne doivent pas être utilisées ensemble.

### 8.3.3 CNNs les Plus Performants pour la Prédiction du Genre et de l'Âge

Afin de concevoir les CNN les plus performants pour la classification du genre et l'estimation de l'âge à partir d'images de visages, nous utilisons les conclusions de la Sous-section 8.3.2 ainsi que les architectures de CNN de l'état de l'art. Plus particulièrement, nous adoptons les architectures de *VGG-16* [SZ15] et de *ResNet-50* [He+15]. *VGG-16* est un choix naturel puisque cette architecture peut être considérée comme une version plus profonde et plus complexe des *fast_CNN*s utilisés dans la Sous-section 8.3.2. En même temps, les *ResNet*s de différentes tailles sont les architectures les plus efficaces pour les problèmes de prédictions aujourd'hui (le *ResNet-50* constitue notamment un très bon équilibre entre le temps de traitement d'une image et les performances obtenues [CPC16]).

Les CNNs *VGG-16* et *ResNet-50* pour la prédiction du genre et de l'âge sont entrainés selon les règles suivantes (qui sont motivées par les conclusions de la Sous-section 8.3.2) :

— Chaque CNN est d'abord pré-entrainé pour la reconnaissance faciale.

— Les CNNs pour la classification du genre et pour l'estimation de l'âge sont entrainés séparément.

— Le LDAE est utilisé pour encoder les âges.

Comme dans la Sous-section 8.3.2, nous entraînons les CNNs sur le corpus *IMDB-Wiki_cleaned* et évaluons leurs performances sur le corpus interne *PBGA*.

| Référence | Année | Approche | CA | |
|---|---|---|---|---|
| | | | *LFW* | *MORPH-II* |
| [GM10] | 2010 | BIF + OLPP | N/A | 97.8% |
| [GM11] | 2011 | BIF + kPLS | N/A | 98.2% |
| [Sha12] | 2012 | LBP + SVM | 94.8% | N/A |
| [TP13] | 2013 | Multiscale LBP + SVM | 95.6% | N/A |
| [GM14] | 2014 | BIF + kCCA | N/A | 98.4% |
| [YLL14] | 2014 | Multi-scale CNN | N/A | 97.9% |
| [Yan+15a] | 2015 | CNN | N/A | 97.9% |
| [Han+15] | 2015 | BIF + hierarchical SVM | N/A | 97.6% |
| | | Human Estimators | N/A | 96.9% |
| [DBM15] | 2015 | FIS + SVM/RBF | 93.4% | N/A |
| [JC15] | 2015 | LBP + C-Pegagos | 96.9% | N/A |
| [MAP16] | 2016 | Local CNN | 94.5% | N/A |
| *Ce travail (Section 8.2)* | *2016* | *Ensemble de optimized_CNNs* | *97.3%* | *N/A* |
| [CSLNRB16] | 2016 | LBP/HOG/CNN + SVM | 98.0% | N/A |
| [Moe+17] | 2017 | SLCDL + CRC | 96.4% | N/A |
| **Ce travail** | **2017** | **ResNet-50 CNN** | **99.3%** | **99.4%** |

TABLE 8.3.3 – Comparaison de notre meilleur CNN pour la classification du genre à partir d'images de visages avec l'état de l'art sur les corpus *LFW* et *MORPH-II*.

| Référence | Année | Approche | MAE | |
|---|---|---|---|---|
| | | | *FG-NET* | *MORPH-II* |
| [Zho+05] | 2005 | Boosting + Regression | 7.48 | N/A |
| [GZSM07] | 2007 | AGES | 6.77 | 8.83 |
| [Guo+08] | 2008 | OLPP + regression | 5.07 | N/A |
| [Luu+09] | 2009 | AAM + SVR | 4.37 | N/A |
| [ZY10] | 2010 | MTWGP | 4.83 | 6.28 |
| [GM10] | 2010 | BIF + OLPP | N/A | 4.33 |
| [Luu+11] | 2011 | CAM + SVR | 4.12 | N/A |
| [CCH11] | 2011 | OHRANK | 4.85 | 5.69 |
| [GM11] | 2011 | BIF + kPLS | N/A | 4.18 |
| [GM14] | 2014 | BIF + kCCA | N/A | 3.92 |
| [YLL14] | 2014 | Multi-scale CNN | N/A | 3.63 |
| [Han+15] | 2015 | BIF + hierarchical SVM | 4.8 | 3.8 |
| | | Human Estimators | 4.7 | 6.3 |
| [WGK15] | 2015 | Unsupervised CNN | 4.11 | 3.81 |
| [Yan+15a] | 2015 | Ranking CNN | N/A | 3.48 |
| [LYK15] | 2015 | Hierarchical grouping and fusion | 2.81-3.55 | 2.97-3.63 |
| [Niu+16] | 2016 | Ordinal CNN | N/A | 3.27 |
| [RTVG16] | 2016 | ImageNet VGG-16 CNN + regression | 3.09 | 2.68* |
| [Liu+17] | 2017 | Group-aware CNN | 3.93 | 3.25 |
| **Ce travail** | **2017** | **VGG-16 CNN + LDAE** | **2.84** | **2.99/2.35*** |

TABLE 8.3.4 – Comparaison de notre meilleur CNN pour l'estimation de l'âge à partir d'images de visages avec l'état de l'art sur les corpus *MORPH-II* et *FG-NET*. (*) Un protocole différent (80% du corpus pour l'apprentissage, 20% du corpus pour l'évaluation).

Nous avons trouvé que le réseau *ResNet-50* obtient le meilleur taux de classification du genre (CA de 98.7% contre CA de 97.1% par *VGG-16*) alors que le réseau *VGG-16* surpasse légèrement *ResNet-50* pour l'estimation de l'âge (Mean Absolute Error (MAE) de 4.26 ans contre MAE de 4.33 ans). Ainsi, nous avons choisi le *ResNet-50* comme notre meilleur modèle pour la classification du genre et le *VGG-16* comme notre meilleur modèle pour l'estimation de l'âge.

Pour comparer ces modèles avec l'état de l'art, nous les avons évalués sur les trois corpus publics les plus utilisés, à savoir : Labeled Faces in the Wild (LFW) [Hua+07] (pour la classification du genre), MORPH-II [RJT06] (pour la classification du genre et pour l'estimation de l'âge) et FG-NET [1] (pour l'estimation de l'âge). Les résultats sont présentés dans les Tableaux 8.3.3 et 8.3.4. À titre indicatif, dans le Tableau 8.3.3, nous fournissons également les scores de la classification du genre par *optimized_CNN* conçu dans la Section 8.2. Comme nous pouvons le constater, les CNNs proposés surpassent nettement l'état de l'art sur les deux tâches (incluant des travaux très récents également basés sur l'apprentissage profond).

### 8.3.4 La Compétition ChaLearn pour l'Estimation de l'Âge Apparent

Notre *VGG-16* CNN qui a été détaillé et évalué dans la Sous-section 8.3.3 est conçu pour l'estimation de *l'âge biologique* (*i.e.* le temps passé depuis la naissance de la personne en question). En même temps, dans certains scénarios, il est aussi important de pouvoir estimer de façon automatique *l'âge apparent* d'un individu (*i.e.* l'âge qui aurait été perçu par un humain moyen). Ainsi, en 2016, une compétition internationale sur l'estimation de l'âge apparent (AAEC) a été organisée par ChaLearn [Esc+16]. Afin de valider notre approche de l'apprentissage de CNN pour l'estimation de l'âge dans le cadre des âges apparents, nous avons participé à cette compétition.

Les organisateurs de AAEC ont collecté un corpus contenant 7591 images de visages (dont 5613 fournies pour l'apprentissage et 1978 utilisées pour l'évaluation). Chaque image de visage a été annotée par au moins 10 humains ce qui permet de calculer la moyenne $\mu$ et l'écart type $\sigma$ de l'âge apparent. Les approches des participants de la compétition ChaLearn ont été évaluées selon la métrique $\varepsilon$ suivante (le meilleur score correspond à la valeur la plus faible) :

$$\varepsilon = 1 - e^{-\frac{(\hat{x}-\mu)^2}{2\sigma^2}} \tag{8.3.1}$$

où $\hat{x}$ est une estimation de l'âge apparent par un algorithme évalué. Par conséquent, les erreurs sur les images avec de faibles écarts types (*i.e.* sur les images sur lesquelles les estimations humaines étaient proches) sont davantage pénalisées que les erreurs sur les images avec des écarts types élevés.

La dernière observation a une importance capitale pour AAEC. En effet, en analysant le corpus de la compétition, nous avons remarqué que la confiance des estimateurs humains dépend de la catégorie de l'âge. Plus particulièrement, les écarts types sont considérablement plus faibles pour les images d'enfants que pour les images d'adultes (*cf.* Figure 8.3.3). Cela veut dire que les résultats de la compétition AAEC dépendent beaucoup de la précision de l'estimation de l'âge apparent des enfants.

C'est pour cette raison que nous avons entrainé deux CNNs séparés sur le corpus de la compétition

1. www.cse.msu.edu/rgroups/biometrics/Publications/Databases/FGNETAgeEstimation.zip

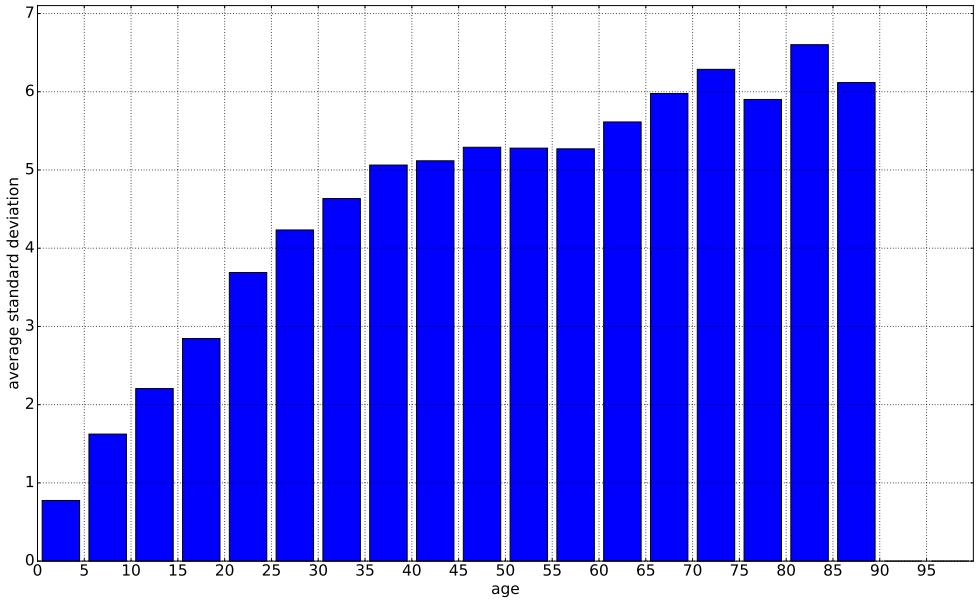



FIGURE 8.3.3 – Histogramme des écarts types d'âges apparents pour les différentes catégories d'âge. Calculé en se basant sur les annotations humaines du corpus de la compétition ChaLearn AAEC [Esc+16]. Chaque colonne correspond à un intervalle de cinq ans.

pour l'estimation de l'âge apparent (en affinant *VGG-16* pour l'âge biologique présenté dans la Sous-section 8.3.3), à savoir : un modèle "général" pour tous les âges et un modèle "enfants" dédié aux enfants de 0 à 12 ans. Pour l'apprentissage du modèle "enfants", nous avons manuellement collecté environ 6K images de visages d'enfants sur Internet. Pendant la phase d'évaluation, nous avons utilisé nos deux modèles de la manière suivante. D'abord, l'estimation de l'âge apparent avec le CNN "général" a été faite. Si l'âge estimé était supérieur à 12 ans, nous avons gardé cette estimation comme réponse finale. Dans le cas contraire, l'estimation par le CNN "enfants" a été faite et considérée comme réponse finale.

| Position | Équipe | Le score $\varepsilon$ | Référence |
|---|---|---|---|
| **1** | **OrangeLabs** | **0.2411** | Ce travail |
| 2 | palm_seu | 0.3214 | [Huo+16] |
| 3 | cmp+ETH | 0.3361 | [Uři+16b] |
| 4 | WYU_CVL | 0.3405 | — |
| 5 | ITU_SiMiT | 0.3668 | [CMAKE16] |
| 6 | Bogazici | 0.3740 | [Gur+16] |
| 7 | MIPAL_SNU | 0.4569 | — |
| 8 | DeepAge | 0.4573 | — |

TABLE 8.3.5 – Le classement finale de la compétition ChaLearn AAEC [Esc+16].

En utilisant cet algorithme de fusion des modèles, nous avons remporté la compétition ChaLearn AAEC. Comme cela peut être constaté dans le Tableau 8.3.5, l'écart des performances entre notre solution et celle de la deuxième place est assez important.

### 8.3.5 Conclusion

Cette section a été consacrée à l'un des deux principaux problèmes de la présente thèse, à savoir la conception de modèles performants pour la classification du genre et pour l'estimation de l'âge à partir d'images de visages.

Nous avons d'abord identifié et comparé les principaux paramètres de la conception et d'apprentissage de CNNs pour les tâches en question. Par conséquent, nous avons établi quelques règles pour un apprentissage efficace qui sont résumées ci-dessous :

1. L'estimation de l'âge est un problème plus complexe que la classification du genre. Le premier demande plus de données d'apprentissage ainsi que des architectures plus profondes (en cas d'apprentissage à partir de zéro).
2. Le pré-apprentissage pour la reconnaissance faciale s'est avéré indispensable pour améliorer l'apprentissage de CNNs pour la prédiction du genre et de l'âge.
3. L'encodage de l'âge LDAE a été montré plus efficace que les encodages 0/1-CAE et RVAE.

Ces conclusions nous ont permis d'apprendre les CNNs très profondes à la base des architectures *VGG-16* et *ResNet-50* qui obtiennent les performances de l'état de l'art sur les corpus de référence les plus utilisés.

De plus, nous avons démontré que notre meilleur CNN pour l'estimation de l'âge biologique peut être simplement adapté à l'estimation de l'âge apparent. Ainsi, nous avons remporté la complétion ChaLearn AAEC sur l'estimation de l'âge apparent en 2016 ce qui confirme les capacités de généralisation de notre CNN pour l'estimation de l'âge.

## 8.4 Synthèse et Édition du Genre et de l'Âge dans des Images de Visage

### 8.4.1 Introduction

Cette dernière section de contribution est entièrement consacrée au deuxième objectif principal de notre travail, présenté dans la Section 8.1, à savoir : la synthèse et l'édition d'images de visages pour un genre et à un âge déterminés.

La présente section est partagée en deux parties. Nous commençons dans la Section 8.4.2 par la proposition de GA-cGAN, qui est, à notre connaissance, le premier réseau de neurones génératif adversaire pour la synthèse d'images de visages ciblant un genre et un âge prédéfinis. Ensuite, nous proposons une méthode permettant l'utilisation de GA-cGAN pour l'édition du genre et de l'âge dans des images de visages existantes tout en préservant l'identité de la personne.

La deuxième partie de l'étude, décrite dans la Section 8.4.3, porte sur l'application de la méthode d'édition de l'âge proposée au préalable (*i.e.* la méthode de vieillissement et de rajeunissement) afin d'améliorer la vérification faciale de cette même personne à des âges différents. Pour cela, nous proposons une extension de la méthode générale qui préserve davantage l'identité originale durant le processus de vieillissement ou de rajeunissement en augmentant légèrement le temps de traitement. Par conséquent,

nous arrivons à démontrer que la normalisation de l'âge avec notre méthode permet d'améliorer les performances d'un moteur de vérification faciale dans le cadre de l'évaluation avec des écarts d'âge.

### 8.4.2 GA-cGAN pour la Synthèse et l'Édition du Genre et de l'Âge

#### 8.4.2.1 Conception et Apprentissage de GA-cGAN

Notre méthode pour la synthèse et l'édition du genre et de l'âge est basée sur des réseaux de neurones génératifs adversaires (GANs) [Goo+15]. Nous commençons donc par un petit rappel des notions essentielles sur les GANs.

**GANs et cGANs : Théorie** Un GAN est une paire de réseaux de neurones : un générateur $G$ et un discriminateur $D$. $G$ génère des images synthétiques $x$ à partir de vecteurs latents $z$ qui sont échantillonnés selon la loi normale $p_z \sim N(0,I)$. Autrement dit, $G$ définit une fonction de l'espace latent $N^z$ vers l'espace d'images $N^x$. Dans ce travail, $N^x$ est un espace d'images de visages. L'objectif du générateur est d'imiter la distribution $p_{data}$ des images de visages. En même temps, le discriminateur essaie de distinguer les images de visages naturelles qui suivent la distribution $p_{data}$ des images synthétisées par le générateur. Ayant en entrée une image $x$ (synthétique ou naturelle), $D$ produit une probabilité que $x$ est plutôt une image naturelle qu'une image synthétique. Par conséquent, les deux réseaux ont des objectifs opposés et sont optimisés de façon itérative dans le cadre d'un jeu "minimax". Plus formellement, l'apprentissage de GAN peut être défini comme une optimisation de la fonction $v(\theta_G, \theta_D)$, où $\theta_G$ et $\theta_D$ sont des paramètres de $G$ et $D$, respectivement :

$$\min_{\theta_G}\max_{\theta_D} V(\theta_G,\theta_D) = \mathbb{E}_{x\sim p_{data}}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1-D(G(z)))] \tag{8.4.1}$$

Un GAN conditionnel (cGAN) [MO14 ; Gau14] est une extension du GAN classique qui permet la génération d'images de visages avec certains attributs (*i.e.* "conditions"). En pratique, ces conditions $y \in N^y$ peuvent contenir des informations quelconques (liées aux visages) : le niveau de luminosité, la pose, les attributs faciaux, etc. L'optimisation des cGAN est très similaire à l'optimisation des GAN classiques (*cf.* Equation 8.4.1). La seule différence est liée au fait que l'information conditionnelle supplémentaire est donnée aux entrées de $G$ et $D$ dans le cas des cGAN :

$$\min_{\theta_G}\max_{\theta_D} V(\theta_G,\theta_D) = \mathbb{E}_{x,y\sim p_{data}}[\log D(x,y)] + \mathbb{E}_{z\sim p_z(z),\tilde{y}\sim p_y}[\log(1-D(G(z,\tilde{y}),\tilde{y}))] \tag{8.4.2}$$

**GA-cGAN** Afin de synthétiser des images de visages avec des attributs définis de genre et d'âge, nous entrainons GA-cGAN, un cGAN conditionné sur le genre et l'âge.

Plus particulièrement, les conditions $y = (y_g, y_a)$ à l'entrée de GA-cGAN sont des vecteurs binaires de taille 7 composés de deux parties, à savoir : $y_g$ (une valeur qui encode le genre) et $y_a$ (un vecteur binaire de taille 6 qui encode l'âge). En effet, nous identifions 6 catégories d'âge, à savoir : "0-18", "19-29", "30-39", "40-49", "50-59" et "60+" ans. Les catégories d'âge sont choisies de façon à ce qu'il y ait au moins 5K images d'apprentissage dans chacune des catégories (nous utilisons le corpus *IMDB-Wiki_cleaned* pour l'apprentissage).

| Générateur $G$ | Discriminateur $D$ | Encodeur $E$ | Reconnaissance Faciale $FR$ |
|---|---|---|---|
| 4x4(2x2)@512, BN, ReLU | 4x4(2x2)@64, LReLU | 5x5(2x2)@32, BN, ReLU | 3x3(1x1)@32, BN, ↓, ELU |
| 4x4(2x2)@256, BN, ReLU | 4x4(2x2)@128, BN, LReLU | 5x5(2x2)@64, BN, ReLU | 3x3(1x1)@64, BN, ↓, ELU |
| 4x4(2x2)@128, BN, ReLU | 4x4(2x2)@256, BN, LReLU | 5x5(2x2)@128, BN, ReLU | 3x3(1x1)@128, BN, ↓, ELU |
| 4x4(2x2)@64, BN, ReLU | 4x4(2x2)@512, BN, LReLU | 5x5(2x2)@256, BN, ReLU | 3x3(1x1)@256, BN, ↓, ELU |
| 4x4(2x2)@3, Tanh | 4x4(1x1)@1, Sigmoid | FC 4096, BN, ReLU | **3x3(1x1)@512, BN, ↓, ELU** |
| — | — | FC 100 | FC 4096, BN, ELU |
| — | — | — | FC 4096, BN, ELU |
| — | — | — | FC 128, Normalize |

TABLE 8.4.1 – Les architectures de CNNs utilisées dans la Section 8.4. kxk(sxs)@M désigne une couche convolutionnelle (ou une couche déconvolutionnelle pour $G$) composée de $M$ noyaux de convolution de taille k et de pas s ; FC N désigne une couche entièrement connectée de N neurones ; BN désigne “batch normalization” ; ↓ désigne un sous-échantillonnage de type “MaxPooling” par un facteur 2.

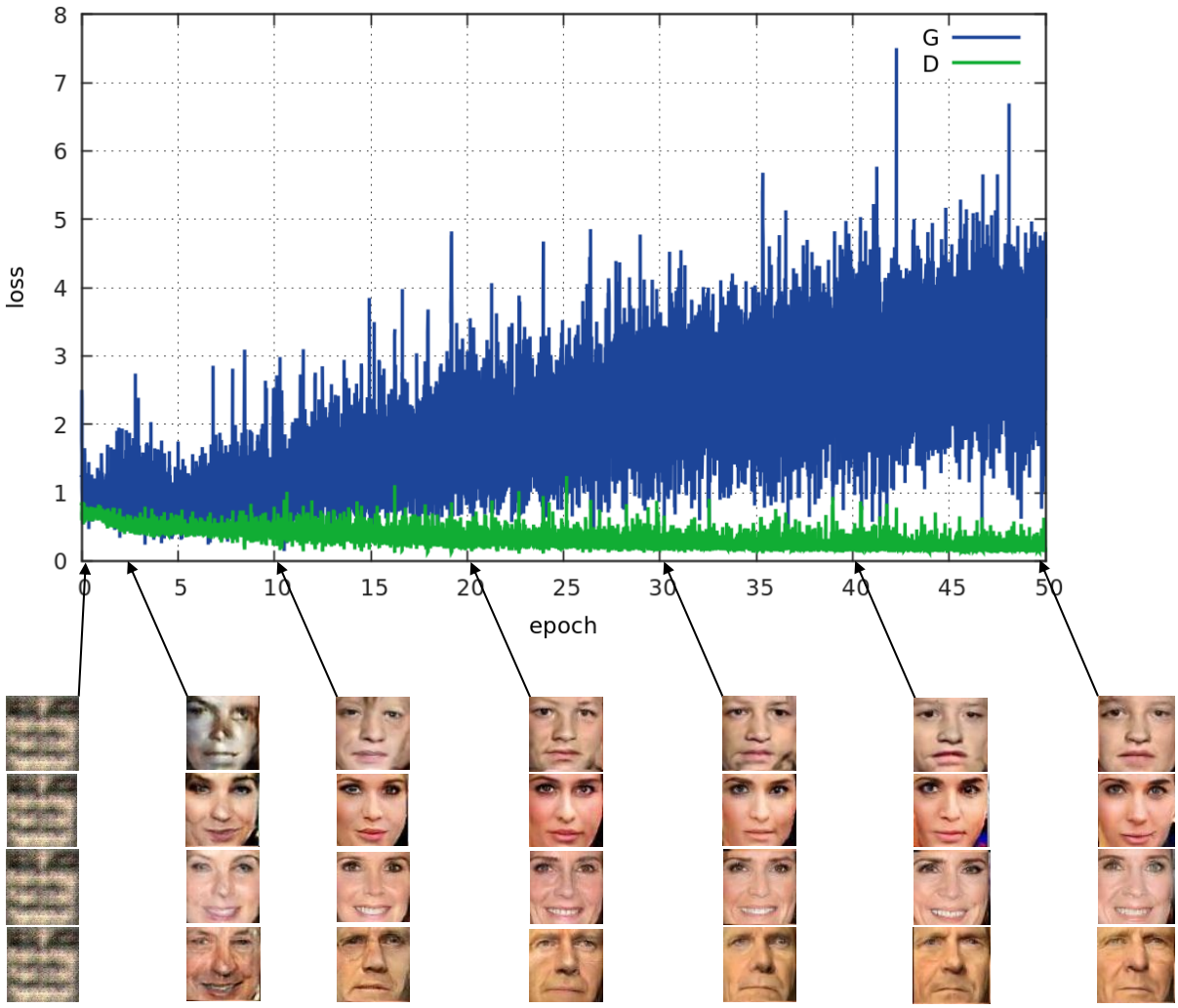



FIGURE 8.4.1 – La progression de l’apprentissage de GA-cGAN. En haut : les courbes des fonctions de perte pour le générateur $G$ et pour le discriminateur $D$. En bas : des exemples d’images de visages synthétisés par $G$ aux différentes étapes de l’apprentissage (les entrées $(z, y)$ sont fixées).

Dans notre GA-cGAN, le générateur $G$ et le discriminateur $D$ sont des CNNs. Nous employons les architectures de $G$ et de $D$ qui avaient été initialement proposées par Radford el al. [RMC16] (*cf.* les deux premières colonnes du Tableau 8.4.1). Les conditions $y$ sont injectées dans $G$ et $D$ selon les conseils

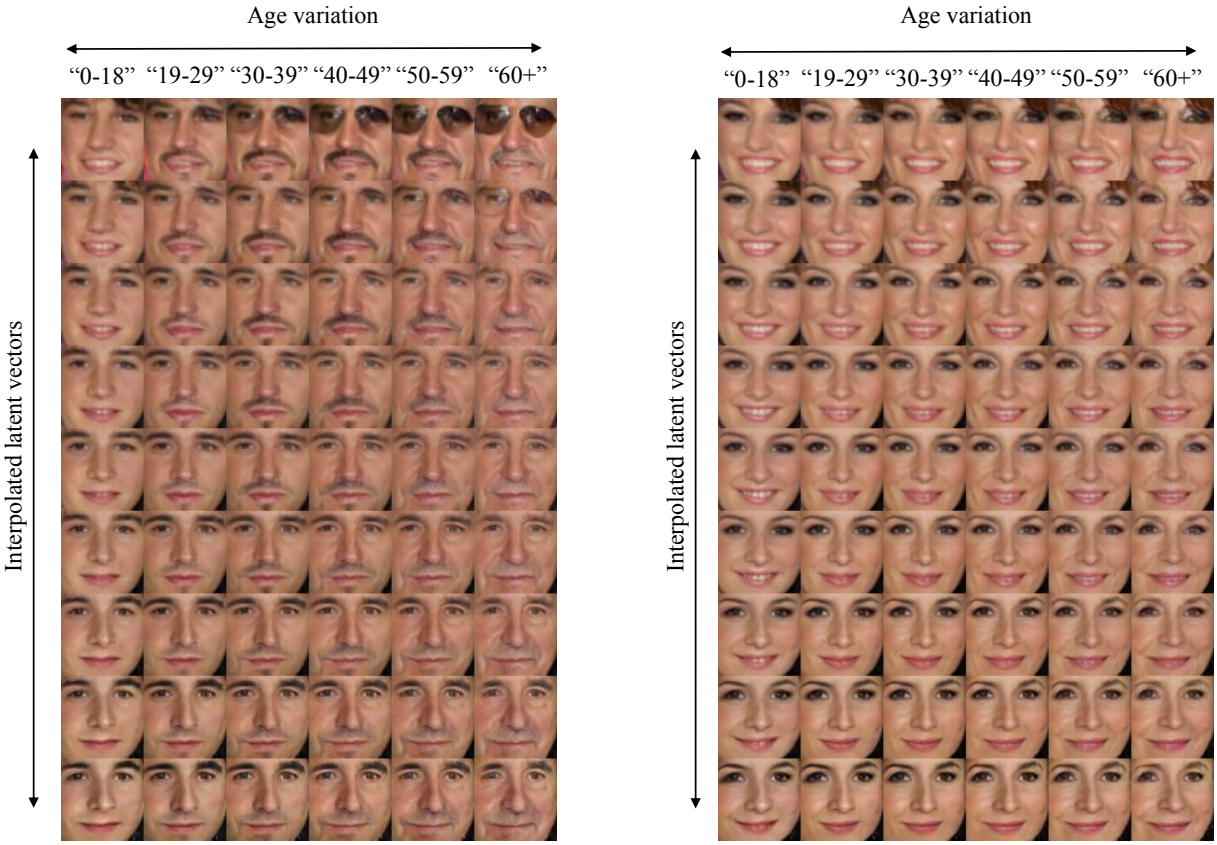



FIGURE 8.4.2 – Exploration de la variété synthétique $\bar{N}^x$ apprise par GA-cGAN. Chaque visage est synthétisé par le générateur $G$ avec des entrées particulières : $(z,(y_g,y_a))$. Les lignes correspondent à la progression dans l'espace latent $N^z$, les colonnes représentent les six conditions d'âge $y_a$ tandis que la condition binaire du genre est altérée $y_g$ entre (a) et (b).

décrits dans [Per+16] (*i.e.* à l'entrée de $G$ et à la première couche de $D$).

Dans la Figure 8.4.1, nous illustrons la progression de l'apprentissage de GA-cGAN. Comme nous pouvons le constater, la partie de la fonction de perte correspondante au générateur croît tout au long de l'apprentissage tandis que la partie correspondante au discriminateur diminue (comportement assez habituel pour des cGANs). La Figure 8.4.1 démontre également comment la qualité des images des visages synthétisées s'améliore pendant les premières phases de l'apprentissage et se stabilise après une quarantaine de phases.

Lorsque l'apprentissage de GA-cGAN est terminé, le générateur $G$ peut synthétiser des images de visages plausibles. En variant les vecteurs latents $z$ et les conditions du genre et de l'âge $y$ à l'entrée de $G$, différentes images synthétiques $\bar{x}$ sont produites. Il est très important que la transformation apprise par le générateur soit continue, ou, autrement dit, des faibles variations dans les vecteurs latents $z$ ou dans les conditions $y$ se traduisent par des petits changements d'images de visages $\bar{x}$ générées. Ainsi, on peut dire que l'ensemble de toutes les images synthétiques qui peuvent être générées avec des différents $z$ et $y$ forme une variété synthétique $\bar{N}^x$. Une partie de cette variété pour GA-cGAN est illustrée dans la Figure 8.4.2.

#### 8.4.2.2 Édition du Genre et de l'Âge avec GA-cGAN

Afin d'éditer une image naturelle $x$ avec un cGAN, il est d'abord nécessaire de la reconstruire (*i.e.* de l'approximer) avec une image synthétique $\bar{x}$ qui fait partie de la variété $\bar{N}^x$. En d'autres termes, nous devons trouver un vecteur latent $z^*$ que nous allons donner en entrée du générateur $G$ pour qu'il produise une reconstruction plausible $\bar{x} = G(z^*, y)$ de l'image initiale $x$ avec les conditions de genre et d'âge $y$. On peut définir une telle reconstruction comme *une projection de l'image du visage $x$ sur la variété synthétique $\bar{N}^x$*.

Dès que la reconstruction $\bar{x}$ de qualité suffisante est obtenue, la modification de l'image $x$ est triviale avec des cGANs. En effet, comme cela peut être constaté dans la Figure 8.4.2, le générateur apprend à démêler les informations encodées par des vecteurs latents $z^*$ et par des conditions $y$. Autrement dit, si nous gardons le même vecteur latent $z^*$ et, en même temps, changeons les conditions $y$, alors seules les conditions $y$ vont changer dans le visage synthétisé.

Ainsi, dans le cas de GA-cGAN, si nous considérons que le genre et l'âge $y^0$ de l'image initiale sont connus au préalable, et si le vecteur latent $z^*$ permettant de reconstruire $\bar{x}_{(y^0)} = G(z^*, y^0)$ est trouvé, alors le visage synthétique avec le genre et l'âge cible $y^1$ peut être généré par la simple substitution de la condition à l'entrée du générateur : $\bar{x}_{(y^1)} = G(z^*, y^1)$.

#### 8.4.2.3 Inférence du Vecteur Latent Optimal

Contrairement aux autoencodeurs, les GANs ne possèdent pas de mécanisme explicite pour la projection sur la variété synthétique. Cependant, ce problème est souvent contourné par l'apprentissage d'un réseau de neurones séparé (un encodeur $E$) qui a pour but d'approximer la transformation inverse par rapport au générateur. Lorsque l'encodeur $E$ est entrainé, la reconstruction peut être simplement faite de la manière suivante : $\bar{x} = G(E(x), y)$, où $y$ sont les conditions (*i.e.* le genre est l'âge) initiales. Par exemple, dans ce travail, nous entrainons un encodeur $E$ avec l'architecture qui est présentée dans la troisième colonne du Tableau 8.4.1.

Les reconstructions obtenues par l'encodeur $E$ arrivent globalement à approximer des images de visages initiales, mais beaucoup de détails sont manquants, et bien souvent, l'identité de la personne originale n'est pas préservée dans le visage synthétique (ce qui n'est pas satisfaisant ici). Zhu et al. [Zhu+16] ont récemment proposé une approche pour remédier à ce problème en optimisant le vecteur latent $z^0$ identifié par l'encodeur. L'idée consiste à minimiser la distance entre l'image originale et sa reconstruction au niveau des pixels. Étant donné que le générateur est un réseau de neurones (et donc différentiable), l'optimisation peut être faite avec l'algorithme L-BFGS-B [Byr+95].

Une telle optimisation apporte en effet une amélioration par rapport à la reconstruction naïve via l'encodeur, par contre, comme nous allons le démontrer par la suite, elle n'est pas suffisante pour pouvoir préserver efficacement l'identité originale. Par conséquent, dans cette thèse, nous proposons une méthode alternative d'optimisation du vecteur latent qui se focalise sur la préservation d'identité. Plus précisément, au lieu de minimiser la distance entre l'image originale et sa reconstruction au niveau des pixels, nous minimisons la distance au niveau des descripteurs d'identités à partir d'images de visage. En pratique, les descripteurs d'identités sont extraits grâce à un réseau de neurones $FR$ (*cf.* la quatrième colonne du Tableau 8.4.1) qui est entrainé pour la reconnaissance faciale. Ainsi, le vecteur latent optimal

$z^*$ est une solution au problème d'optimisation suivant :

$$z^* = \underset{z}{\operatorname{argmin}} ||FR(x) - FR(\bar{x})||_{L_2} \tag{8.4.3}$$

Afin d'évaluer la qualité de la préservation d'identité originale par les trois méthodes d'inférence du vecteur latent qui sont proposées ci-dessus (à savoir : l'inférence via l'encodeur $E$, l'inférence via l'optimisation au niveau des pixels et l'inférence proposée dans cette étude), nous effectuons une comparaison objective grâce au moteur de vérification faciale OpenFace [ALS16] qui est utilisé comme une boîte noire. OpenFace prend une paire d'images de visages à l'entrée et produit une valeur réelle à la sortie qui correspond à la distance entre les identités dans les deux visages (si cette distance est inférieure à un seuil pré-défini, le logiciel "pense" que les deux visages appartiennent à la même personne, et vice versa). En particulier, nous employons deux protocoles d'évaluation décrits ci-dessous :

| Reconstruction | Protocole 1 (FV score) | Protocole 2 (FV score) |
|---|---|---|
| Encodeur ($\bar{x}^0$) | 89.0% | 78.1% |
| Optimisation au niveau des pixels ($\bar{x}^{pixel}$) | 94.5% | 78.5% |
| Optimisation au niveau des descripteurs d'identité ($\bar{x}^{IP}$) | **97.6%** | **82.0%** |
| Optimisation au niveau des descripteurs d'identité + LMA ($\bar{x}^{IP+LMA}$) | **100.0%** | **88.7%** |

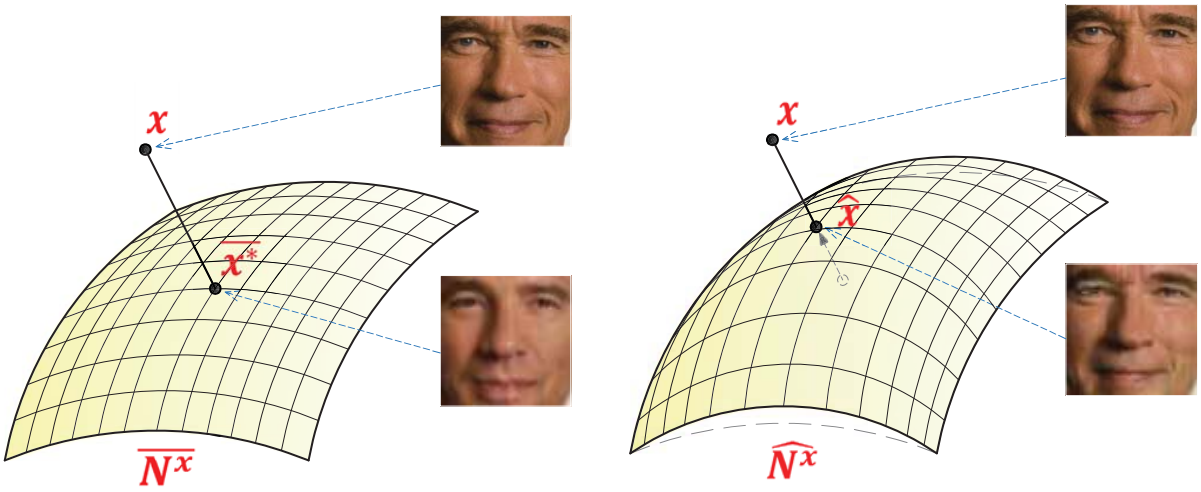
TABLE 8.4.2 – Comparaison des quatre méthodes de la reconstruction d'images de visages présentées dans les Sous-section 8.4.2 et 8.4.3, à savoir : la reconstruction simple avec l'encodeur $E$ ($\bar{x}^0$), la reconstruction via l'optimisation au niveau des pixels ($\bar{x}^{pixel}$), (notre approche pour) la reconstruction via l'optimisation au niveau des descripteurs d'identité ($\bar{x}^{IP}$) et (notre approche pour) l'amélioration de la reconstruction précédente via LMA. L'évaluation est faite selon les deux protocoles : dans le premier, le meilleur score de la vérification faciale (FV) est de 100.0%, alors que dans le deuxième, il est de 89.4%.

— Dans le cadre du premier protocole, les 10K images de visages $x$ sont données à l'entrée d'OpenFace jumelées avec ses reconstructions $\bar{x}$ (obtenues avec les trois méthodes comparées). Pour chaque méthode, nous évaluons donc le pourcentage de cas dans lesquels le logiciel classifie une paire constituée de l'image originale et de sa reconstruction comme une paire positive. Dans ce cas là, le score idéal est évidemment de 100%.

— Le deuxième protocole constitue un défi plus complexe pour les trois méthodes d'inférence de vecteur latent comparées. En effet, dans ce cas là, nous évaluons la possibilité d'utiliser des images reconstruites au lieu des images naturelles pour la vérification d'identité avec OpenFace. Plus précisément, OpenFace obtient une précision de 89.4% (avec le cadrage "face-only" qui est utilisé dans cette étude) sur le corpus *LFW* (selon le protocole d'évaluation classique qui est décrit sur le site officiel de *LFW*[2]). Si les reconstructions des images de visages sont parfaites, le score d'OpenFace sur la version synthétique (*i.e.* reconstruite) de *LFW* ne doit pas baisser par rapport au score original.

Ainsi, dans le Tableau 8.4.2 (les trois premières lignes), nous présentons les résultats de la comparaison des trois méthodes de l'inférence du vecteur latent (ou autrement dit, de la reconstruction d'image de visage) selon les deux protocoles d'évaluations utilisés. Comme cela peut être constaté, notre méthode

2. http://vis-www.cs.umass.edu/lfw/




(a) : avant LMA (b) : après LMA

FIGURE 8.4.3 – La méthode d'adaptation locale de la variété (LMA) pour l'amélioration de la préservation d'identité originale dans une reconstruction synthétique.

qui est basée sur la minimisation de distance entre les descripteurs d'identités, dépasse nettement à la fois la reconstruction avec l'encodeur $E$ et l'optimisation au niveau des pixels.

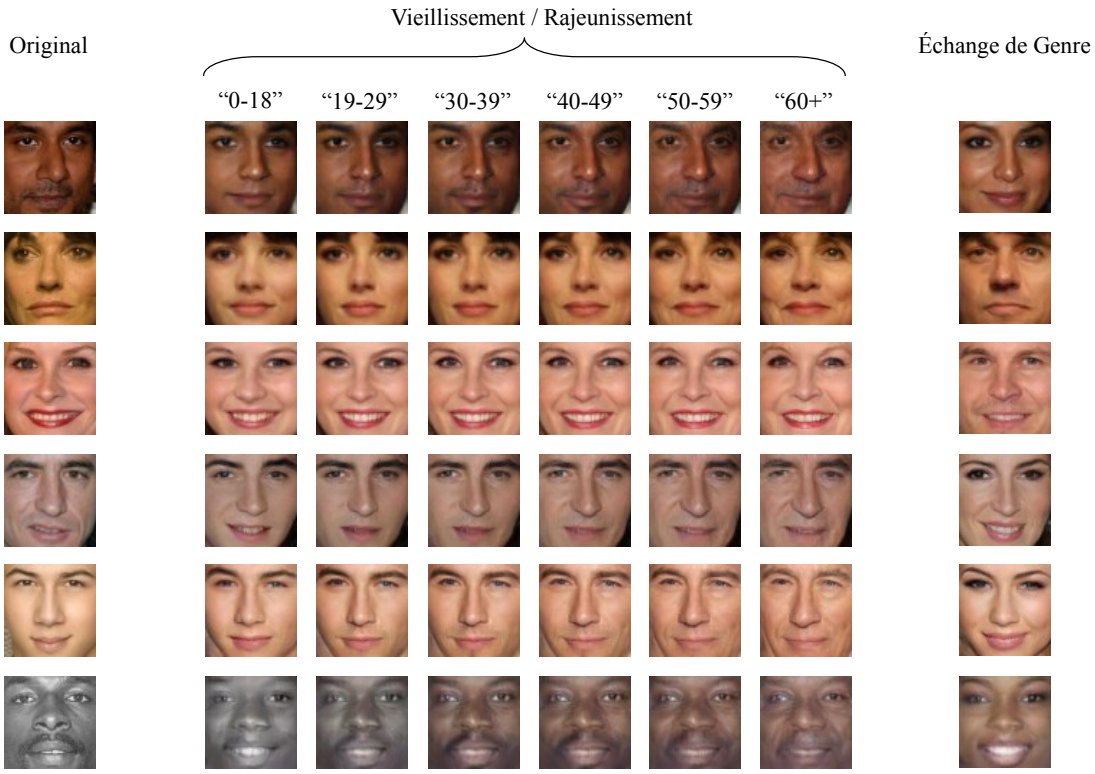
Néanmoins, les scores de notre approche (la ligne 3 dans le Tableau 8.4.2) ne sont pas parfaits. Cela est particulièrement visible dans le cadre du deuxième protocole où nous sommes à plus de 7 points en dessous du score optimal. Dans la Sous-section 8.4.3, nous allons démontrer comment améliorer davantage les reconstructions synthétiques pour pouvoir appliquer GA-cGAN à la normalisation de l'âge dans le cadre de la vérification faciale.

### 8.4.3 Normalisation de l'Âge pour l'Amélioration de la Vérification Faciale

#### 8.4.3.1 L'Adaptaion Locale de la Variété (LMA)

Dans la Sous-section précédente, nous avons constaté que malgré le choix explicit du vecteur latent $z^*$ qui est inféré pour la préservation d'identité, l'image synthétique $\bar{x}^*$ obtenue avec $z^*$ ne permet pas de reconstruire tous les détails de l'image de visage originale $x$. En effet, souvent, une image initiale $x$ est tellement éloignée de la variété synthétique $\bar{N}^x$ (puisque le générateur $G$ ne peut évidemment pas modéliser toutes les variations illimitées d'images de visages naturelles), que même la projection optimale $\bar{x}^*$ de cette image sur $\bar{N}^x$ reste trop éloignée de l'originale (*cf.* la Figure 8.4.3-(a)).

Afin de remédier à ce problème, nous proposons une méthode nommée *l'adaptation locale de la variété* (en anglais Local Manifold Adaptation, ou LMA) qui consiste en la modification locale de la variété $\bar{N}^x$ pour s'approcher de l'image de visage donnée à l'entrée $x$ (*cf.* la Figure 8.4.3-(b)). Pour cela, nous adaptons le générateur $G$ de GA-cGAN en fonction de l'image initiale $x$ pour trouver un nouveau générateur $G_x$ qui (1) est capable de produire une reconstruction $\widehat{x}$ suffisamment proche de $x$ et (2) préserve les capacités de $G$ pour la synthèse d'images de visages réalistes avec les genres et les âges demandés. En pratique, pour trouver $G_x$, nous effectuons un nombre fixe $N_{iter}$ d'itérations de la rétro-propagation avec un taux d'apprentissage très faible $\mu$ pour minimiser $||FR(x) - FR(G(z^*, y^0))||_{L_2}$, où $y^0$ désigne le genre et l'âge originaux de l'image de visage $x$ (les hyper-paramètres $\{N_{iter}, \mu\}$ sont choisis




(a) : Édition d'image de visage sans LMA


(b) : Édition d'image de visage avec LMA

FIGURE 8.4.4 – Exemples de vieillissement / rajeunissement d'image de visage avec GA-cGAN sans et avec LMA. Dans les deux cas, notre méthode de l'inférence de vecteur latent (basée sur l'optimisation au niveau de descripteurs faciales) est utilisée.

empiriquement).

Dans la dernière ligne du Tableau 8.4.2, nous présentons les résultats de préservation d'identité dans les reconstructions améliorées avec LMA. Comme on peut le constater, la préservation d'identité est parfaite selon le protocole 1 et quasi-parfaite selon le protocole 2.

Finalement, dans les Figures 8.4.4-(a) et -(b), nous illustrons quelques exemples d'édition du genre et de l'âge dans les photos obtenues avec la méthode de l'inférence du vecteur latent présentée dans la Sous-section 8.4.2 sans et avec LMA, respectivement. Il est directement perceptible que LMA permet d'augmenter le réalisme des modifications sémantiques dans les visages initiaux à travers une meilleure préservation des identités originales.

#### 8.4.3.2 Expérimentations

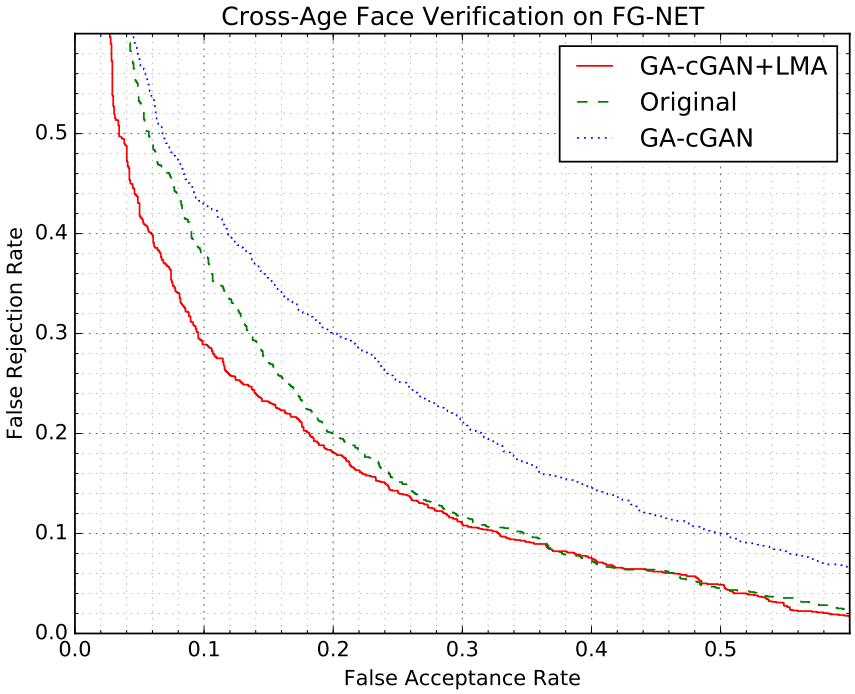



FIGURE 8.4.5 – Les courbes d'erreurs de la vérification faciale avec OpenFace calculées sur le corpus *FG-NET*.

Afin de souligner l'utilité de notre méthode d'édition d'images de visages avec GA-cGAN dans le cadre d'un problème réel, nous l'appliquons à la normalisation de l'âge dans les paires d'images de visage avant la vérification faciale. Plus particulièrement, nous effectuons les expérimentations avec le logiciel de vérification faciale OpenFace sur le corpus *FG-NET*. Pour nos expérimentations, nous avons sélectionné toutes les paires positives de *FG-NET* pour lesquelles l'écart d'âge est supérieur à 10 ans (il y en a 1519) et le même nombre de paires négatives choisies aléatoirement.

Dans la Figure 8.4.5, nous comparons les courbes "FAR contre FRR" obtenues par OpenFace sur les paires originales (la courbe verte) et sur les paires synthétiques avec des âges normalisés par notre méthode (la courbe rouge). Comme cela peut être constaté, la normalisation de l'âge permet de diminuer le taux d'erreur de type "faux négatif" jusqu'à 8 points. De plus, la Figure 8.4.5 démontre également

la nécessité de la méthode LMA proposée dans cette Sous-section, car la normalisation de l'âge avec GA-cGAN sans LMA (la courbe bleu) entraîne une chute drastique des performances.

### 8.4.4 Conclusion

La présente section a abordé les problèmes de synthèse et d'édition d'images de visages avec le genre et l'âge demandés. Les contributions de la section sont résumées ci-dessous :

1. Nous avons entrainé GA-cGAN, le cGAN pour la synthèse d'images de visages conditionné sur le genre et l'âge. GA-cGAN est capable de transformer de façon continue les vecteurs de l'espace latent ($N(0,I)$) vers des images de visages plausibles avec le genre et l'âge souhaité.
2. Nous avons proposé une nouvelle méthode d'inférence du vecteur latent en fonction de l'image de visage en entrée qui surpasse les méthodes existantes sur la qualité de la préservation de l'identité originale. Cette méthode permet la reconstruction et l'édition d'images de visages naturelles avec GA-cGAN.
3. Finalement, nous avons proposé une approche nommée LMA qui permet de perfectionner la préservation de l'identité dans les reconstructions par GA-cGAN. Par conséquent, nous avons démontré qu'avec LMA, GA-cGAN peut être utilisé pour le vieillissement / rajeunissement dans le cadre de la vérification faciale pour compenser les variations d'âge dans les images évaluées.

## 8.5 Conclusion Générale

Les résultats principaux de cette thèse se résument en trois contributions principales qui sont présentées ci-dessous :

— Nos deux études préliminaires ont démontré que l'efficacité des descripteurs appris par les CNNs dépendent de la complexité du problème cible. Plus particulièrement, les descripteurs appris par les CNNs se sont avérés beaucoup plus efficaces que les descripteurs manuellement conçus sur le problème de la classification du genre à partir d'images de piétons. Au contraire, dans le cadre de la classification du genre à partir d'images de visages (qui est un problème plus simple), les CNNs n'arrivent pas à extraire de meilleures descripteurs pendant l'apprentissage à partir de zéro. Ce résultat explique la nécessité du pré-apprentissage sur un problème complexe avant l'apprentissage pour la classification du genre à partir d'images de visages.

— Nous avons effectué une étude poussée qui a abouti à une formulation empirique des principes de la conception et de l'apprentissage optimaux de CNNs pour la prédiction du genre et de l'âge à partir d'images de visages. Notamment, nous démontrons que la reconnaissance faciale est très utile pour le pré-apprentissage de CNNs de la classification du genre. Nous identifions également LDAE comme une méthode d'encodage d'âge très adaptée pour l'apprentissage de CNN de l'estimation de l'âge. Au final, ces principes nous ont permis de concevoir les CNNs de l'état de l'art pour la classification du genre et l'estimation de l'âge sur les trois corpus les plus utilisés ainsi que de gagner une compétition internationale sur l'estimation de l'âge apparent.

— Nous avons proposé le modèle GA-cGAN qui est à notre connaissance le premier cGAN pour la synthèse conditionnée d'images de visages avec le genre et l'âge demandés. Ensuite, nous avons

présenté une nouvelle méthode permettant d'appliquer GA-cGAN à l'édition du genre et de l'âge dans les images de visages en préservant l'identité de l'image originale. Cette méthode est composée de deux étapes : (1) d'abord, nous inférons le vecteur latent qui préserve au maximum l'identité de la personne avec le générateur générique, et (2) ensuite, nous adaptons le générateur à l'image en question pour encore améliorer la préservation de l'identité. Nous illustrons l'intérêt pratique de notre approche, en appliquant GA-cGAN à la normalisation de l'âge dans le cadre de la vérification faciale. Cela permet l'amélioration jusqu'à 8 points des performances d'un moteur de vérification faciale en présence des variations d'âges.

Dans le travail futur, nous planifions de généraliser les conclusions de cette thèse sur les autres modalités faciales (comme l'expression, les différents attributs, etc). Cela concerne non seulement la conception de CNNs pour l'estimation de ces modalités, mais aussi la minimisation de leur impact sur nos modèles de la classification du genre et de l'estimation de l'âge (par exemple, nous avons remarqué que notre meilleure CNN de l'âge n'est pas très robuste aux variations de l'expression faciale). Autrement, nous planifions d'augmenter la résolution d'images de visages synthétisées et éditées par notre GA-cGAN (pour le moment, il s'agit de seulement 64x64 pixels). Pour cela, nous envisageons l'utilisation des Wasserstein GANs qui ont été proposées très récemment [ACB17].

# Bibliography


[ACB17] Martin Arjovsky, Soumith Chintala and Léon Bottou. "Wasserstein gan". In: *CoRR* abs/1701.07875 (2017).

[ALS16] Brandon Amos, Bartosz Ludwiczuk and Mahadev Satyanarayanan. *OpenFace: A general-purpose face recognition library with mobile applications*. Tech. rep. CMU-CS-16-118, CMU School of Computer Science, 2016.

[Bac+12] Moez Baccouche, Franck Mamalet, Christian Wolf, Christophe Garcia and Atilla Baskurt. "Spatio-Temporal Convolutional Sparse Auto-Encoder for Sequence Classification". In: *Proceedings of British Machine Vision Conference*. Surrey, UK, 2012.

[Bac13] Moez Baccouche. "Apprentissage neuronal de caractéristiques spatio-temporelles pour la classification automatique de séquences vidéo". PhD thesis. INSA de Lyon, 2013.

[Bar+13] Oren Barkan, Jonathan Weill, Lior Wolf and Hagai Aronowitz. "Fast high dimensional vector multiplication face recognition". In: *Proceedings of International Conference on Computer Vision*. Sydney, Australia, 2013.

[BCBB11] Juan Bekios-Calfa, Jose M Buenaposada and Luis Baumela. "Revisiting linear discriminant techniques in gender recognition". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33.4 (2011), pp. 858–864.

[BCV13] Yoshua Bengio, Aaron Courville and Pascal Vincent. "Representation learning: A review and new perspectives". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.8 (2013), pp. 1798–1828.

[BDB16] Piotr Bilinski, Antitza Dantcheva and François Brémond. "Can a smile reveal your gender?" In: *Proceedings of Biometrics Special Interest Group*. Darmstadt, Germany, 2016.

[Bel+13] Peter N Belhumeur, David W Jacobs, David J Kriegman and Narendra Kumar. "Localizing parts of faces using a consensus of exemplars". In: *Transactions on Pattern Analysis and Machine Intelligence* 35.12 (2013), pp. 2930–2940.

[Ben+09] Yoshua Bengio et al. "Learning deep architectures for AI". In: *Foundations and trends in Machine Learning* 2.1 (2009), pp. 1–127.

[Ben+14] Yoshua Bengio, Eric Laufer, Guillaume Alain and Jason Yosinski. “Deep generative stochastic networks trainable by backprop”. In: *Proceedings of International Conference on Machine Learning*. Beijing, China, 2014.

[BMM11] Lubomir Bourdev, Subhransu Maji and Jitendra Malik. “Describing people: A poselet-based approach to attribute classification”. In: *Proceedings of International Conference on Computer Vision*. Barcelona, Spain, 2011.

[BP95] D Michael Burt and David I Perrett. “Perception of age in adult Caucasian male faces: Computer graphic manipulation of shape and colour information”. In: *Proceedings of the Royal Society of London B: Biological Sciences* 259.1355 (1995), pp. 137–143.

[BR07] Shumeet Baluja and Henry A Rowley. “Boosting sex identification performance”. In: *International Journal on Computer Vision* 71.1 (2007), pp. 111–119.

[BSF94] Yoshua Bengio, Patrice Simard and Paolo Frasconi. “Learning long-term dependencies with gradient descent is difficult”. In: *IEEE transactions on neural networks* 5.2 (1994), pp. 157–166.

[Byr+95] Richard H Byrd, Peihuang Lu, Jorge Nocedal and Ciyou Zhu. “A limited memory algorithm for bound constrained optimization”. In: *SIAM Journal on Scientific Computing* 16.5 (1995), pp. 1190–1208.

[Cai+06] Deng Cai, Xiaofei He, Jiawei Han and H-J Zhang. “Orthogonal laplacianfaces for face recognition”. In: *IEEE Transactions on Image Processing* 15.11 (2006), pp. 3608–3614.

[Cao+08] Liangliang Cao, Mert Dikmen, Yun Fu and Thomas S Huang. “Gender recognition from body”. In: *Proceedings of ACM Conference on Multimedia*. Vancouver, Canada, 2008.

[CBPA16] Souad Chaabouni, Jenny Benois-Pineau and Chokri Ben Amar. “Transfer learning with deep networks for saliency prediction in natural video”. In: *Proceedings of International Conference on Image Processing*. Phoenix, USA, 2016.

[CCH11] Kuang-Yu Chang, Chu-Song Chen and Yi-Ping Hung. “Ordinal hyperplanes ranker with cost sensitivities for age estimation”. In: *Proceedings of Computer Vision and Pattern Recognition*. Colorado Springs, USA, 2011.

[CCH14] Bor-Chun Chen, Chu-Song Chen and Winston H Hsu. “Cross-age reference coding for age-invariant face recognition and retrieval”. In: *Proceedings of European Conference on Computer Vision*. Zurich, Switzerland, 2014.

[CET+01] Timothy F Cootes, Gareth J Edwards, Christopher J Taylor et al. “Active appearance models”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23.6 (2001), pp. 681–685.

[Che+10] Jie Chen, Shiguang Shan, Chu He, Guoying Zhao, Matti Pietikainen, Xilin Chen and Wen Gao. “WLD: A robust local image descriptor”. In: *Transactions on Pattern Analysis and Machine Intelligence* 32.9 (2010), pp. 1705–1720.

[Cio+12] T Ciodaro, D Deva, JM De Seixas and D Damazio. “Online particle detection with neural networks based on topological calorimetry information”. In: *Journal of physics: conference series*. Vol. 368. 1. 2012, p. 012030.

[CMAKE16] Refik Can Malli, Mehmet Aygun and Hazim Kemal Ekenel. "Apparent Age Estimation Using Ensemble of Deep Learning Models". In: *Proceedings of Computer Vision and Pattern Recognition Workshops*. Las Vegas, USA, 2016.

[Col+09] Matthew Collins, Jianguo Zhang, Paul Miller and Hongbin Wang. "Full body image feature representations for gender profiling". In: *Proceedings of International Conference on Compurer Vision*. Kyoto, Japan, 2009.

[Col+11] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu and Pavel Kuksa. "Natural language processing (almost) from scratch". In: *Journal of Machine Learning Research* 12.Aug (2011), pp. 2493–2537.

[CPC16] Alfredo Canziani, Adam Paszke and Eugenio Culurciello. "An Analysis of Deep Neural Network Models for Practical Applications". In: *CoRR* abs/1605.07678 (2016).

[CR11] Cunjian Chen and Arun Ross. "Evaluation of gender classification methods on thermal and near-infrared face images". In: *Proceedings of International Joint Conference on Biometrics*. Washington DC, USA, 2011.

[CS+16] Modesto Castrillón-Santana, Maria De Marsico, Michele Nappi and Daniel Riccio. "MEG: Texture operators for multi-expert gender classification". In: *Computer Vision and Image Understanding* (2016).

[CSLNRB16] M Castrillón-Santana, J Lorenzo-Navarro and E Ramón-Balmaseda. "Descriptors and regions of interest fusion for gender classification in the wild. Comparison and combination with CNNs." In: *CoRR* abs/1507.06838v2 (2016).

[CUH16] Djork-Arné Clevert, Thomas Unterthiner and Sepp Hochreiter. "Fast and accurate deep network learning by exponential linear units (elus)". In: (2016).

[CV95] Corinna Cortes and Vladimir Vapnik. "Support-vector networks". In: *Machine learning* 20.3 (1995), pp. 273–297.

[Dan+11] Antitza Dantcheva, Carmelo Velardo, Angela D'Angelo and Jean-Luc Dugelay. "Bag of soft biometrics for person identification". In: *Multimedia Tools and Applications* 51.2 (2011), pp. 739–777.

[DBM15] Taner Danisman, Ioan Marius Bilasco and Jean Martinet. "Boosting gender recognition performance with a fuzzy inference system". In: *Expert Systems with Applications* 42.5 (2015), pp. 2772–2784.

[Den+14] Yubin Deng, Ping Luo, Chen Change Loy and Xiaoou Tang. "Pedestrian attribute recognition at far distance". In: *Proceedings of ACM Conference on Multimedia*. Orlando, USA, 2014, pp. 789–792.

[DER16] Antitza Dantcheva, Petros Elia and Arun Ross. "What else does your biometric data reveal? A survey on soft biometrics". In: *Transactions on Information Forensics and Security* 11.3 (2016), pp. 441–467.

[Dib+12] Hamdi Dibeklioğlu, Theo Gevers, Albert Ali Salah and Roberto Valenti. "A smile can reveal your age: Enabling facial dynamics in age estimation". In: *Proceedings of ACM Multimedia*. Nara, Japan, 2012.

[DK17] Alexey Dosovitskiy and Vladlen Koltun. “Learning to act by predicting the future”. In: *Proceedings of International Conference on Learning Representations*. Toulon, France, 2017.

[DLL16] Yuan Dong, Yinan Liu and Shiguo Lian. “Automatic age estimation based on deep learning algorithm”. In: *Neurocomputing* 187 (2016), pp. 4–10.

[Don+14] Chao Dong, Chen Change Loy, Kaiming He and Xiaoou Tang. “Learning a deep convolutional network for image super-resolution”. In: *Proceedings of European Conference on Computer Vision*. Zurich, Switzerland, 2014.

[DT05] Navneet Dalal and Bill Triggs. “Histograms of oriented gradients for human detection”. In: *Proceedings of Computer Vision and Pattern Recognition*. san Diego, USA, 2005.

[Dum+17] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Alex Lamb, Martin Arjovsky, Olivier Mastropietro and Aaron Courville. “Adversarially learned inference”. In: *Proceedings of International Conference on Learning Representations*. Toulon, France, 2017.

[Ekm16] Ari Ekmekji. *Convolutional Neural Networks for Age and Gender Classification*. Tech. rep. Stanford University, 2016. URL: http://cs231n.stanford.edu/reports2016/003_Report.pdf.

[Ela+12] Khaoula Elagouni, Christophe Garcia, Franck Mamalet and Pascale Sébillot. “Combining multi-scale character recognition and linguistic knowledge for natural scene text OCR”. In: *Proceedings of International Workshop on Document Analysis Systems*. Queensland, Australia, 2012.

[Esc+15] Sergio Escalera, Junior Fabian, Pablo Pardo, Xavier Baro et al. “ChaLearn Looking at People 2015: Apparent Age and Cultural Event Recognition Datasets and Results”. In: *Proceedings of International Conference on Computer Vision Workshops*. Santiago, Chile, 2015.

[Esc+16] Sergio Escalera, Mercedes Torres, Brais Martinez, Xavier Baro et al. “ChaLearn Looking at People and Faces of the World: Face Analysis Workshop and Challenge 2016”. In: *Proceedings of Computer Vision and Pattern Recognition Workshops*. Las Vegas, USA, 2016.

[Fel97] Jean-Marc Fellous. “Gender discrimination and prediction on the basis of facial metric information”. In: *Vision research* 37.14 (1997), pp. 1961–1973.

[FGH10] Yun Fu, Guodong Guo and Thomas S Huang. “Age synthesis and estimation via faces: A survey”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.11 (2010), pp. 1955–1976.

[Fis+15] Philipp Fischer, Alexey Dosovitskiy, Eddy Ilg, Philip Häusser, Caner Hazırbaş, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers and Thomas Brox. “Flownet: Learning optical flow with convolutional networks”. In: *Proceedings of International Conference on Computer Vision*. Santiago, Chile, 2015.

[FS89] Itzhak Fogel and Dov Sagi. “Gabor filters as texture discriminator”. In: *Biological cybernetics* 61.2 (1989), pp. 103–113.

[FS97] Yoav Freund and Robert E Schapire. “A decision-theoretic generalization of on-line learning and an application to boosting”. In: *Journal of computer and system sciences* 55.1 (1997), pp. 119–139.

[Gau14] Jon Gauthier. “Conditional generative adversarial nets for convolutional face generation”. In: *Class Project for Stanford CS231N: Convolutional Neural Networks for Visual Recognition* (2014).

[GBC16] Ian Goodfellow, Yoshua Bengio and Aaron Courville. *Deep learning*. 2016.

[GBT07] Douglas Gray, Shane Brennan and Hai Tao. “Evaluating appearance models for recognition, reacquisition, and tracking”. In: *Proceedings of Performance Evaluation for Tracking and Surveillance Workshop*. Rio de Janeiro, Brazil, 2007.

[GD04] Christophe Garcia and Manolis Delakis. “Convolutional face finder: A neural architecture for fast and robust face detection”. In: *IEEE Transactions on pattern analysis and machine intelligence* 26.11 (2004), pp. 1408–1423.

[GH95] Patricia A George and Graham J Hole. “Factors influencing the accuracy of age estimates of unfamiliar faces”. In: *Perception* 24.9 (1995), pp. 1059–1073.

[Gir+14] Ross Girshick, Jeff Donahue, Trevor Darrell and Jitendra Malik. “Rich feature hierarchies for accurate object detection and semantic segmentation”. In: *Proceedings of Conference on Computer Vision and Pattern Recognition*. Colorado, USA, 2014.

[GLS90] Beatrice A Golomb, David T Lawrence and Terrence J Sejnowski. “SEXNET: A Neural Network Identifies Sex From Human Faces”. In: *Proceedings of Advances in Neural Information Processing Systems*. Denver, USA, 1990.

[GM10] Guodong Guo and Guowang Mu. “Human age estimation: What is the influence across race and gender?” In: *Proceedings of Computer Vision and Pattern Recognition Workshops*. San Francisco, USA, 2010.

[GM11] Guodong Guo and Guowang Mu. “Simultaneous dimensionality reduction and human age estimation via kernel partial least squares regression”. In: *Proceedings of Computer Vision and Pattern Recognition*. Colorado Springs, USA, 2011.

[GM14] Guodong Guo and Guowang Mu. “A framework for joint estimation of age, gender and ethnicity on a large database”. In: *Image and Vision Computing* 32.10 (2014), pp. 761–770.

[GMF10] Guodong Guo, Guowang Mu and Yun Fu. “Gender from body: A biologically-inspired approach with manifold learning”. In: *Proceedings of Asian Conference on Computer Vision*. Xi’an, China, 2010.

[GMH13] Alex Graves, Abdel-rahman Mohamed and Geoffrey Hinton. “Speech recognition with deep recurrent neural networks”. In: *Proceedings of International cCnference on Acoustics, Speech and Signal Processing*. Vancouver, Canada, 2013.

[GN07] Asuman Gunay and Vasif V Nabiyev. “Automatic detection of anthropometric features from facial images”. In: *Proceedings of Signal Processing and Communications Applications*. Eskisehir, Turkey, 2007.

[GN08] Asuman Gunay and Vasif V Nabiyev. “Automatic age classification with LBP”. In: *Proceedings of International Symposium on Computer and Information Sciences*. Izmir, Turkey, 2008.

[Goo+15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville and Yoshua Bengio. “Generative adversarial nets”. In: *Proceedings of advances in Neural Information Processing Systems*. Montreal, Canada, 2015.

[Goo16] Ian Goodfellow. “NIPS 2016 Tutorial: Generative Adversarial Networks”. In: *CoRR* abs/1701.00160 (2016).

[GS+16] Ester Gonzalez-Sosa, Antitza Dantcheva, Ruben Vera-Rodriguez, Jean-Luc Dugelay, François Brémond and Julian Fierrez. “Image-based gender estimation from body and face across distances”. In: *Proceedings of International Conference on Pattern Recognition*. Cancun, Mexico, 2016.

[GS05] Alex Graves and Jürgen Schmidhuber. “Framewise phoneme classification with bidirectional LSTM and other neural network architectures”. In: *Neural Networks* 18.5 (2005), pp. 602–610.

[Guo+08] Guodong Guo, Yun Fu, Charles R Dyer and Thomas S Huang. “Image-based human age estimation by manifold learning and locally adjusted robust regression”. In: *Transactions on Image Processing* 17.7 (2008), pp. 1178–1188.

[Guo+09] Guodong Guo, Guowang Mu, Yun Fu and Thomas S Huang. “Human age estimation using bio-inspired features”. In: *Proceedings of Computer Vision and Pattern Recognition*. 2009, Miami, USA.

[Guo+16] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He and Jianfeng Gao. “Ms-celeb-1m: challenge of recognizing one million celebrities in the real world”. In: *Electronic Imaging* 2016.11 (2016), pp. 1–6.

[Guo12] Guodong Guo. “Human age estimation and sex classification”. In: (2012), pp. 101–131.

[Gur+16] Furkan Gurpinar, Heysem Kaya, Hamdi Dibeklioglu and Ali Salah. “Kernel ELM and CNN based facial age estimation”. In: *Proceedings of Computer Vision and Pattern Recognition Workshops*. Las Vegas, USA, 2016.

[GW12] Guodong Guo and Xiaolong Wang. “A study on human age estimation under facial expression changes”. In: *Proceedings of Computer Vision and Pattern Recognition*. Rhode Island, USA, 2012.

[GWP98] Srinivas Gutta, Harry Wechsler and P Jonathon Phillips. “Gender and ethnic classification of face images”. In: *Proceedings of Automatic Face and Gesture Recognition*. Nara, Japan, 1998.

[GYZ13] Xin Geng, Chao Yin and Zhi-Hua Zhou. “Facial age estimation by learning from label distributions”. In: *Transactions on Pattern Analysis and Machine Intelligence* 35.10 (2013), pp. 2401–2412.

[GZSM07] Xin Geng, Zhi-Hua Zhou and Kate Smith-Miles. "Automatic age estimation based on facial aging patterns". In: *Transactions of Pattern Analysis and Machine Intelligence* 29.12 (2007), pp. 2234–2240.

[Han+15] Hu Han, Charles Otto, Xiaoming Liu and Anil K Jain. "Demographic estimation from face images: Human vs. machine performance". In: *Transactions on pattern analysis and machine intelligence* 37.6 (2015), pp. 1148–1161.

[He+15] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun. "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification". In: *Proceedings of Computer Vision and Pattern Recognition*. Boston, USA, 2015.

[He+16] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun. "Deep Residual Learning for Image Recognition". In: *Proceedings of Computer Vision and Pattern Recognition*. Las Vegas, USA, 2016.

[HE16] Jonathan Ho and Stefano Ermon. "Generative adversarial imitation learning". In: *Proceedings of advances in Neural Information Processing Systems*. Barcelona, Spain, 2016.

[Hin+12] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath et al. "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups". In: *IEEE Signal Processing Magazine* 29.6 (2012), pp. 82–97.

[HKO04] Aapo Hyvärinen, Juha Karhunen and Erkki Oja. *Independent component analysis*. Vol. 46. 2004.

[HM82] James A Hanley and Barbara J McNeil. "The meaning and use of the area under a receiver operating characteristic (ROC) curve." In: *Radiology* 143.1 (1982), pp. 29–36.

[HMD12] Tri Huynh, Rui Min and Jean-Luc Dugelay. "An efficient LBP-based descriptor for facial depth images applied to gender recognition using RGB-D face data". In: *Proceedings of Asian Conference on Computer Vision*. Daejeon, Korea, 2012.

[Hoc+01] Sepp Hochreiter, Yoshua Bengio, Paolo Frasconi and Jürgen Schmidhuber. *Gradient flow in recurrent nets: the difficulty of learning long-term dependencies*. 2001.

[HOT06] Geoffrey E Hinton, Simon Osindero and Yee-Whye Teh. "A fast learning algorithm for deep belief nets". In: *Neural computation* 18.7 (2006), pp. 1527–1554.

[HP09] Abdenour Hadid and Matti Pietikäinen. "Manifold learning for gender classification from face sequences". In: *Advances in Biometrics* (2009), pp. 82–91.

[HS06] Geoffrey E Hinton and Ruslan R Salakhutdinov. "Reducing the dimensionality of data with neural networks". In: *science* 313.5786 (2006), pp. 504–507.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[HSA84] Geoffrey E Hinton, Terrence J Sejnowski and David H Ackley. *Boltzmann machines: Constraint satisfaction networks that learn*. Carnegie-Mellon University, Department of Computer Science Pittsburgh, PA, 1984.

[Hsu02] Feng-Hsiung Hsu. *Behind Deep Blue: Building the computer that defeated the world chess champion*. Princeton University Press, 2002.

[Hu+11] Si Ying Diana Hu, Brendan Jou, Aaron Jaech and Marios Savvides. "Fusion of region-based representations for gender identification". In: *Proceedings of International Joint Conference on Biometrics*. Washington DC, USA, 2011.

[Hua+07] Gary B Huang, Manu Ramesh, Tamara Berg and Erik Learned-Miller. *Labeled faces in the wild: A database for studying face recognition in unconstrained environments*. Tech. rep. University of Massachusetts, Amherst, 2007.

[Huo+16] Zengwei Huo, Xu Yang, Chao Xing, Ying Zhou, Peng Hou, Jiaqi Lv and Xin Geng. "Deep age distribution learning for apparent age estimation". In: *Proceedings of Computer Vision and Pattern Recognition Workshops*. Las Vegas, USA, 2016.

[HUP09] Xia Han, Hassan Ugail and Ian Palmer. "Gender classification based on 3D face geometry features using SVM". In: *Proceedings of International Conference on CyberWorlds*. Bradford, UK, 2009.

[Iba+14] Julian Ibarz, Ian Goodfellow, Sacha Arnoud, Vinay Shet and Yaroslav Bulatov. "Multi-digit Number Recognition from Street View Imagery using Deep Convolutional Neural Networks". In: *Proceedings of Internation Conference on Learning Representations*. Banff, Canada, 2014.

[IS13] Mehmet Yasar Iscan and Maryan Steyn. *The human skeleton in forensic medicine*. 2013.

[IS15] Sergey Ioffe and Christian Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". In: *Proceedings of International Conference on Machine Learning*. Lille, France, 2015.

[Iso+17] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou and Alexei A Efros. "Image-to-image translation with conditional adversarial networks". In: (2017).

[JC15] Sen Jia and Nello Cristianini. "Learning to classify gender from four million images". In: *PRL* 58 (2015), pp. 35–41.

[JDN04] Anil K Jain, Sarat C Dass and Karthik Nandakumar. "Soft biometric traits for personal recognition systems". In: *Biometric authentication*. 2004, pp. 731–738.

[JH04] Amit Jain and Jeffrey Huang. "Integrating independent components and linear discriminant analysis for gender classification". In: *Proceedings of Automatic Face and Gesture Recognition*. Seoul, South Korea, 2004.

[JKFF16] Justin Johnson, Andrej Karpathy and Li Fei-Fei. "Densecap: Fully convolutional localization networks for dense captioning". In: *Proceedings of Computer Vision and Pattern Recognition*. Las Vegas, USA, 2016.

[Joa98] Thorsten Joachims. *Making large-scale SVM learning practical*. Tech. rep. Technical Report, SFB 475: Komplexitätsreduktion in Multivariaten Datenstrukturen, Universität Dortmund, 1998.

[Jor98] Michael Irwin Jordan. *Learning in graphical models*. Vol. 89. 1998.

[JX+16] Felix Juefei-Xu, Eshan Verma, Parag Goel, Anisha Cherodian and Marios Savvides. "DeepGender: Occlusion and Low Resolution Robust Facial Gender Classification via Progressively Trained Convolutional Neural Networks with Attention". In: *Proceedings of Computer Vision and Pattern Recognition Workshops*. Las Vegas, USA, 2016.

[Jég+10] Hervé Jégou, Matthijs Douze, Cordelia Schmid and Patrick Pérez. "Aggregating local descriptors into a compact image representation". In: *Proceedings of Computer Vision and Pattern Recognition*. San Francisco, USA, 2010.

[Kar+14] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar and Li Fei-Fei. "Large-scale video classification with convolutional neural networks". In: *Proceedings of Computer Vision and Pattern Recognition*. Columbus, USA, 2014.

[KB14] Diederik Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *CoRR* abs/1412.6980 (2014).

[KE13] Pavel Korshunov and Touradj Ebrahimi. "Using face morphing to protect privacy". In: *Proceedings of Advanced Video and Signal Based Surveillance*. Krakow, Poland, 2013.

[KH11] Alex Krizhevsky and Geoffrey E Hinton. "Using very deep autoencoders for content-based image retrieval." In: *Proceedings of European Symposium on Artificial Neural Networks*. Bruges, Belgium, 2011.

[KMM05] Asifullah Khan, Abdul Majid and Anwar M Mirza. "Combination and optimization of classifiers in gender classification using genetic programming". In: *International Journal of Knowledge-based and Intelligent Engineering Systems* 9.1 (2005), pp. 1–11.

[Koc+96] Rolf M Koch, Markus H Gross, Friedrich R Carls, Daniel F von Büren, George Fankhauser and Yoav IH Parish. "Simulating facial surgery using finite element models". In: *Proceedings of Computer Graphics and Interactive Techniques*. New Orleans, USA, 1996.

[KSH12] Alex Krizhevsky, Ilya Sutskever and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Proceedings of advances in Neural Information Processing Systems*. Lake Tahoe, USA, 2012.

[KSSS14] Ira Kemelmacher-Shlizerman, Supasorn Suwajanakorn and Steven M Seitz. "Illumination-aware age progression". In: *Proceedings of Computer Vision and Pattern Recognition*. Columbus, USA, 2014.

[KW14] Diederik P Kingma and Max Welling. "Auto-encoding variational bayes". In: *Proceedings of International Conference on Learning Representations*. Banff, Canada, 2014.

[Lar+16] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle and Ole Winther. "Autoencoding beyond pixels using a learned similarity metric". In: *Proceedings of International Conference on Machine Learning*. New York, USA, 2016.

[Lay+12] Ryan Layne, Timothy M Hospedales, Shaogang Gong and Q Mary. "Person Re-identification by Attributes". In: *Proceedigns of British Machine Vision Conference*. Guildford, UK, 2012.

[LBH15] Yann LeCun, Yoshua Bengio and Geoffrey Hinton. “Deep learning”. In: *Nature* 521.7553 (2015), pp. 436–444.

[LCY14] Min Lin, Qiang Chen and Shuicheng Yan. “Network in network”. In: *Proceedings of International Conference on Learning Representations*. Banff, Canada, 2014.

[LDC04] Andreas Lanitis, Chrisina Draganova and Chris Christodoulou. “Comparing different classifiers for automatic age estimation”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 34.1 (2004), pp. 621–628.

[LeC+89] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard and Lawrence D Jackel. “Backpropagation applied to handwritten zip code recognition”. In: *Neural computation* 1.4 (1989), pp. 541–551.

[LeC+98] Yann LeCun, Léon Bottou, Yoshua Bengio and Patrick Haffner. “Gradient-based learning applied to document recognition”. In: *IEEE* 86.11 (1998), pp. 2278–2324.

[LeC85] Yann LeCun. “Une procédure d’apprentissage pour réseau a seuil asymmetrique (a learning scheme for asymmetric threshold networks)”. In: *Proceedings of Cognitiva 85, Paris, France*. 1985.

[Led+16] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang et al. “Photo-realistic single image super-resolution using a generative adversarial network”. In: *CoRR* abs/1609.04802 (2016).

[LH15] Gil Levi and Tal Hassner. “Age and gender classification using convolutional neural networks”. In: *Proceedings of Computer Vision and Pattern Recognition Workshops*. Boston, USA, 2015.

[LI00] SR Loth and MY Iscan. “ANTHROPOLOGY| Sex Determination”. In: *Encyclopedia of forensic sciences* (2000), pp. 252–260.

[Lia+16] Xiaodan Liang, Yunchao Wei, Xiaohui Shen, Zequn Jie, Jiashi Feng, Liang Lin and Shuicheng Yan. “Reversible recursive instance-level object segmentation”. In: *Proceedings of Conference on Computer Vision and Pattern Recognition*. Las Vegas, USA, 2016.

[Liu+15a] Xin Liu, Shaoxin Li, Meina Kan, Jie Zhang et al. “Agenet: Deeply learned regressor and classifier for robust apparent age estimation”. In: *Proceedings of International Conference on Computer Vision Workshops*. Santiago, Chile, 2015.

[Liu+15b] Ziwei Liu, Ping Luo, Xiaogang Wang and Xiaoou Tang. “Deep learning face attributes in the wild”. In: *Proceedings of International Conference on Computer Vision*. Santiago, Chile, 2015.

[Liu+15c] Ziwei Liu, Ping Luo, Xiaogang Wang and Xiaoou Tang. “Deep Learning Face Attributes in the Wild”. In: *Proceedings of International Conference on Computer Vision*. Santiago, Chile, 2015.

[Liu+17] Hao Liu, Jiwen Lu, Jianjiang Feng and Jie Zhou. “Group-aware deep feature learning for facial age estimation”. In: *Pattern Recognition* 66 (2017), pp. 82–94.

[LM+16] Erik Learned-Miller, Gary B Huang, Aruni RoyChowdhury, Haoxiang Li and Gang Hua. "Labeled faces in the wild: A survey". In: *Advances in Face Detection and Facial Image Analysis*. Springer, 2016, pp. 189–248.

[Low99] David G Lowe. "Object recognition from local scale-invariant features". In: *Proceedings of Computer Vision and Pattern Recognition*. Colorado, USA, 1999.

[LSD15] Jonathan Long, Evan Shelhamer and Trevor Darrell. "Fully convolutional networks for semantic segmentation". In: *Proceedings of Computer Vision and Pattern Recognition*. Boston, USA, 2015.

[LT10] Jiwen Lu and Yap-Peng Tan. "Gait-based human age estimation". In: *IEEE Transactions on Information Forensics and Security* 5.4 (2010), pp. 761–770.

[LTC02] Andreas Lanitis, Christopher J. Taylor and Timothy F Cootes. "Toward automatic simulation of aging effects on face images". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.4 (2002), pp. 442–455.

[Lu+15] Jiwen Lu, Junlin Hu, Venice Erin Liong, Xiuzhuang Zhou, Andrea Bottino, Ihtesham Ul Islam, Tiago Figueiredo Vieira, Xiaoqian Qin, Xiaoyang Tan, Songcan Chen et al. "The fg 2015 kinship verification in the wild evaluation". In: *Proceedings of Automatic Face and Gesture Recognition Workshops*. Ljubljana, Slovenia, 2015.

[Luc+16] Pauline Luc, Camille Couprie, Soumith Chintala and Jakob Verbeek. "Semantic segmentation using adversarial networks". In: *Proceedings of advances in Neural Information Processing Systems Workshops*. Barcelona, Spain, 2016.

[Luu+09] Khoa Luu, Karl Ricanek, Tien D Bui and Ching Y Suen. "Age estimation using active appearance models and support vector machine regression". In: *Proceedings of Biometrics: Theory, Applications and Systems*. Washington DC, USA, 2009.

[Luu+11] Khoa Luu, Keshav Seshadri, Marios Savvides, Tien D Bui and Ching Y Suen. "Contourlet appearance model for facial age estimation". In: *Proceedings of International Joint Conference on Biometrics*. Washington DC, USA, 2011.

[LW02] Chengjun Liu and Harry Wechsler. "Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition". In: *Transactions on Image Processing* 11.4 (2002), pp. 467–476.

[LYK15] Kuan-Hsien Liu, Shuicheng Yan and C-C Jay Kuo. "Age estimation via grouping and decision fusion". In: *Transactions on Image Forensics and Security* 10.11 (2015), pp. 2408–2423.

[MAP16] Jordi Mansanet, Alberto Albiol and Roberto Paredes. "Local deep neural networks for gender recognition". In: *Pattern Recognition Letters* 70 (2016), pp. 80–86.

[Mat+14] Markus Mathias, Rodrigo Benenson, Marco Pedersoli and Luc Van Gool. "Face detection without bells and whistles". In: *Proceedings of European Conference on Computer Vision*. Zurich, Switzerland, 2014.

[MCL16] Michael Mathieu, Camille Couprie and Yann LeCun. “Deep multi-scale video prediction beyond mean square error”. In: *Proceedings of International Conference on Learning Representations*. San Juan, Puerto Rico, 2016.

[MDH09] Abdel-rahman Mohamed, George Dahl and Geoffrey Hinton. “Deep belief networks for phone recognition”. In: *Proceedings of advances in Neural Information Processing Systems Workshops*. Vancouver, Canada, 2009.

[Mil95] George A Miller. “WordNet: a lexical database for English”. In: *Communications of the ACM* 38.11 (1995), pp. 39–41.

[MO14] Mehdi Mirza and Simon Osindero. “Conditional generative adversarial nets”. In: *Proceedings of advances in Neural Information Processing Systems*. Montreal, Canada, 2014.

[Moe+17] Ali Moeini, Hossein Moeini, Armon Matthew Safai and Karim Faez. “Regression Facial Attribute Classification via simultaneous dictionary learning”. In: *Pattern Recognition* 62 (2017), pp. 99–113.

[MP09] MARIANNA Madry-Pronobis. “Automatic gender recognition based on audiovisual cues”. PhD thesis. 2009.

[MP43] Warren S McCulloch and Walter Pitts. “A logical calculus of the ideas immanent in nervous activity”. In: *The bulletin of mathematical biophysics* 5.4 (1943), pp. 115–133.

[MP69] Marvin Minsky and Seymour Papert. “Perceptrons.” In: (1969).

[MRF15] Mateusz Malinowski, Marcus Rohrbach and Mario Fritz. “Ask your neurons: A neural-based approach to answering questions about images”. In: *Proceedings of International Conference on Computer Vision*. Santiago, Chile, 2015.

[MY02] Baback Moghaddam and Ming-Hsuan Yang. “Learning gender with support faces”. In: *Transactions on Pattern Analysis and Machine Intelligence* 24.5 (2002), pp. 707–711.

[Nes83] Yurii Nesterov. “A method of solving a convex programming problem with convergence rate O (1/k2)”. In: *Soviet Mathematics Doklady*. Vol. 27. 2. 1983, pp. 372–376.

[Nev+14] Natalia Neverova, Christian Wolf, Graham Taylor and Florian Nebout. “Multi-scale deep learning for gesture detection and localization”. In: *Proceedings of European Conference on Computer Vision Workshops*. Zurich, Switzerland, 2014.

[Niu+16] Zhenxing Niu, Mo Zhou, Le Wang, Xinbo Gao and Gang Hua. “Ordinal regression with multiple output cnn for age estimation”. In: *Proceedings of Computer Vision and Pattern Recognition*. Las Vegas, USA, 2016.

[NTG13] Choon-Boon Ng, Yong-Haur Tay and Bok-Min Goi. “A convolutional neural network for pedestrian gender recognition”. In: *Advances in Neural Networks*. 2013, pp. 558–564.

[OAE16] Gokhan Ozbulak, Yusuf Aytar and Hazim Kemal Ekenel. “How Transferable are CNN-based Features for Age and Gender Classification?” In: *Proceedings of BIOSIG*. Darmstadt, Germany, 2016.

[Oor+16] Aaron Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves et al. "Conditional image generation with pixelcnn decoders". In: *Proceedings of advances in Neural Information Processing Systems*. Barcelona, Spain, 2016.

[OOS16] Augustus Odena, Christopher Olah and Jonathon Shlens. "Conditional image synthesis with auxiliary classifier gans". In: *CoRR* abs/1610.09585 (2016).

[OPH96] Timo Ojala, Matti Pietikäinen and David Harwood. "A comparative study of texture measures with classification based on featured distributions". In: *Pattern recognition* 29.1 (1996), pp. 51–59.

[OR14] Asem A Othman and Arun Ross. "Privacy of Facial Soft Biometrics: Suppressing Gender But Retaining Identity." In: *Proceedings of European Conference on Computer Vision Workshops*. Zurich, Switzerland, 2014.

[Ore+97] Michael Oren, Constantine Papageorgiou, Pawan Sinha, Edgar Osuna and Tomaso Poggio. "Pedestrian detection using wavelet templates". In: *Proceedings of Computer Vision and Pattern Recognition*. San Juan, Puerto Rico, 1997.

[Pat+16] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell and Alexei A Efros. "Context encoders: Feature learning by inpainting". In: *Proceedings of Conference on Computer Vision and Pattern Recognition*. Las Vegas, USA, 2016.

[Pay10] Pascal Paysan. "Statistical modeling of facial aging based on 3D scans". PhD thesis. University of Basel, 2010.

[PBP92] Brunelli Poggio, R Brunelli and T Poggio. "HyberBF networks for gender classification". In: (1992).

[Per+10] Florent Perronnin, Yan Liu, Jorge Sánchez and Hervé Poirier. "Large-scale image retrieval with compressed fisher vectors". In: *Proceedings of Computer Vision and Pattern Recognition*. San Francisco, USA, 2010.

[Per+16] Guim Perarnau, Joost van de Weijer, Bogdan Raducanu and Jose M Álvarez. "Invertible Conditional GANs for image editing". In: *Proceedings of advances in Neural Information Processing Systems Workshops*. Barcelona, Spain, 2016.

[Pla+86] David C Plaut et al. "Experiments on Learning by Back Propagation." In: (1986).

[PVZ15] Omkar M Parkhi, Andrea Vedaldi and Andrew Zisserman. "Deep face recognition". In: *Proceedings of British Machine Vision Conference*. Swansea, UK, 2015.

[PY10] Sinno Jialin Pan and Qiang Yang. "A survey on transfer learning". In: *IEEE Transactions on Knowledge and Data Engineering* 22.10 (2010), pp. 1345–1359.

[RC06a] Narayanan Ramanathan and Rama Chellappa. "Face verification across age progression". In: *IEEE Transactions on Image Processing* 15.11 (2006), pp. 3349–3361.

[RC06b] Narayanan Ramanathan and Rama Chellappa. "Modeling age progression in young faces". In: *Proceedings of Computer Vision and Pattern Recognition*. New York, USA, 2006.

[Red+16] Joseph Redmon, Santosh Divvala, Ross Girshick and Ali Farhadi. "You only look once: Unified, real-time object detection". In: *Proceedings of Conference on Computer Vision and Pattern Recognition*. Las Vegas, USA, 2016.

[Ree+16] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele and Honglak Lee. "Generative adversarial text to image synthesis". In: *Proceedings of International Conference on Machine Learning*. New York, USA, 2016.

[Rho56] Henry Taylor Fowkes Rhodes. *Alphonse Bertillon, father of scientific detection*. 1956.

[RHW85] David E Rumelhart, Geoffrey E Hinton and Ronald J Williams. *Learning internal representations by error propagation*. Tech. rep. 1985.

[RIM17] Yaniv Romano, John Isidoro and Peyman Milanfar. "RAISR: rapid and accurate image super resolution". In: *IEEE Transactions on Computational Imaging* 3.1 (2017), pp. 110–125.

[RJT06] Karl Ricanek Jr and Tamirat Tesafaye. "Morph: A longitudinal image database of normal adult age-progression". In: *Proceedings of Automatic Face and Gesture Recognition*. Southampton, UK, 2006.

[RMC16] Alec Radford, Luke Metz and Soumith Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks". In: *Proceedings of International Conference on Learning Representations*. San Juan, Puerto Rico, 2016.

[RMG06] Sebastien Roux, Franck Mamalet and Christophe Garcia. "Embedded convolutional face finder". In: *Proceedings of International Conference on Multimedia and Expo*. Toronto, Canada, 2006.

[RNJ07] Arun A Ross, Karthik Nandakumar and Anil K Jain. *Handbook of biometrics*. 2007.

[Ros58] Frank Rosenblatt. "The perceptron: A probabilistic model for information storage and organization in the brain." In: *Psychological review* 65.6 (1958), p. 386.

[RP95] Duncan A Rowland and David I Perrett. "Manipulating facial appearance through shape and color". In: *Computer Graphics and Applications* 15.5 (1995), pp. 70–76.

[RP99] Maximilian Riesenhuber and Tomaso Poggio. "Hierarchical models of object recognition in cortex". In: *Nature neuroscience* 2.11 (1999), pp. 1019–1025.

[RS00] Sam T Roweis and Lawrence K Saul. "Nonlinear dimensionality reduction by locally linear embedding". In: *science* 290.5500 (2000), pp. 2323–2326.

[RTVG16] Rasmus Rothe, Radu Timofte and Luc Van Gool. "Deep expectation of real and apparent age from a single image without facial landmarks". In: *International Journal of Computer Vision* (2016), pp. 1–14.

[Rus+15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein et al. "Imagenet large scale visual recognition challenge". In: *International Journal of Computer Vision* 115.3 (2015), pp. 211–252.

[Sai+13] Tara N Sainath, Abdel-rahman Mohamed, Brian Kingsbury and Bhuvana Ramabhadran. "Deep convolutional neural networks for LVCSR". In: *Proceedings of International Conference on Acoustics, Speech and Signal Processing*. Vancouver, Canada, 2013.

[Sal+16] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford and Xi Chen. "Improved techniques for training gans". In: *Proceedings of advances in Neural Information Processing Systems*. Barcelona, Spain, 2016.

[Sam59] Arthur L Samuel. "Some studies in machine learning using the game of checkers". In: *IBM Journal of research and development* 3.3 (1959), pp. 210–229.

[See+08] Richard D Seely, Sina Samangooei, Middleton Lee, John N Carter and Mark S Nixon. "The university of southampton multi-biometric tunnel and introducing a novel 3d gait dataset". In: *Proceedings of Biometrics: Theory, Applications and Systems*. Virginia, USA, 2008.

[SG07] Zohra Saidane and Christophe Garcia. "Automatic scene text recognition using a convolutional neural network". In: *Proceedings of International Conference on Document Analysis and Recognition Workshops*. Kyoto, Japan, 2007.

[Sha11] Lior Shamir. "Automatic age estimation by hand photos". In: *Computer Science Letters* 3.1 (2011).

[Sha12] Caifeng Shan. "Learning local binary patterns for gender classification on real-world face images". In: *Pattern Recognition Letters* 33.4 (2012), pp. 431–437.

[She+11] Cheng-Ta Shen, Wan-Hua Lu, Sheng-Wen Shih and Hong-Yuan Mark Liao. "Exemplar-based age progression prediction in children faces". In: *Proceedings of International Symposium on Multimedia*. California, USA, 2011.

[Shi+12] D Shihfeng, Chien-Hsun Tu, Chih-Yao Chuang and Huei-Ting Lin. "Aging simulation using facial muscle model". In: *Proceedings of International Conference on Machine Learning and Cybernetics*. Xian, China, 2012.

[Shi13] Huang-Chia Shih. "Robust gender classification using a precise patch histogram". In: *Pattern Recognition* 46.2 (2013), pp. 519–528.

[Shu+15] Xiangbo Shu, Jinhui Tang, Hanjiang Lai, Luoqi Liu and Shuicheng Yan. "Personalized age progression with aging dictionary". In: *Proceedings of International Conference on Computer Vision*. Santiago, Chile, 2015.

[Shu+16] Xiangbo Shu, Guo-Sen Xie, Zechao Li and Jinhui Tang. "Age progression: current technologies and applications". In: *Neurocomputing* 208 (2016), pp. 249–261.

[Sil+16] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot et al. "Mastering the game of Go with deep neural networks and tree search". In: *Nature* 529.7587 (2016), pp. 484–489.

[SKP15] Florian Schroff, Dmitry Kalenichenko and James Philbin. "Facenet: A unified embedding for face recognition and clustering". In: *Proceedings of Computer Vision and Pattern Recognition*. Boston, USA, 2015.

[Smo86] Paul Smolensky. *Information processing in dynamical systems: Foundations of harmony theory*. Tech. rep. 1986.

[SR+14] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan and Stefan Carlsson. "CNN features off-the-shelf: an astounding baseline for recognition". In: *Proceedings of Computer Vision and Pattern Recognition Workshops*. Columbus, USA, 2014.

[SR+17] Philippe Pérez San-Roman, Jenny Benois-Pineau, Jean-Philippe Domenger, Aymar De Rugy, Florent Paclet and Daniel Cataert. "Saliency Driven Object recognition in egocentric videos with deep CNN: toward application in assistance to Neuroprostheses". In: *Computer Vision and Image Understanding* 164 (2017), pp. 82–91.

[Sri+14] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever and Ruslan Salakhutdinov. "Dropout: a simple way to prevent neural networks from overfitting." In: *Journal of Machine Learning Research* 15.1 (2014), pp. 1929–1958.

[Sun+06] Ning Sun, Wenming Zheng, Changyin Sun, Cairong Zou and Li Zhao. "Gender classification based on boosting local binary pattern". In: *Proceedings of Advances in Neural Networks* (2006).

[Suo+10] Jinli Suo, Song-Chun Zhu, Shiguang Shan and Xilin Chen. "A compositional and dynamic model for face aging". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.3 (2010), pp. 385–401.

[Suo+11] Jinli Suo, Liang Lin, Shiguang Shan, Xilin Chen and Wen Gao. "High-resolution face fusion for gender conversion". In: *IEEE Transactions on Systems, Man, and Cybernetics* 41.2 (2011), pp. 226–237.

[SVL14] Ilya Sutskever, Oriol Vinyals and Quoc V Le. "Sequence to sequence learning with neural networks". In: *Proceedings of Advances in Neural Information Processing Systems*. Montreal, Canada, 2014.

[SWL16] Leon Sixt, Benjamin Wild and Tim Landgraf. "Rendergan: Generating realistic labeled data". In: *Proceedings of International Conference on Learning Representations Workshops*. San Juan, Puerto Rico, 2016.

[SWT15] Yi Sun, Xiaogang Wang and Xiaoou Tang. "Deeply learned face representations are sparse, selective, and robust". In: *Proceedings of Computer Vision and Pattern Recognition*. Boston, USA, 2015.

[SZ15] Karen Simonyan and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition". In: *Proceedings of International Conference on Learning Representations*. San Diego, USA, 2015.

[Sze+15] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke and Andrew Rabinovich. "Going deeper with convolutions". In: *Proceedings of Computer Vision and Pattern Recognition*. Boston, USA, 2015.

[TA09] Matthew Toews and Tal Arbel. "Detection, localization, and sex classification of faces from arbitrary viewpoints and under occlusion". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31.9 (2009), pp. 1567–1581.

[Tai+14] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato and Lior Wolf. "Deepface: Closing the gap to human-level performance in face verification". In: *Proceedings of Computer Vision and Pattern Recognition*. Columbus, USA, 2014.

[Tan+17] Wei Ren Tan, Chee Seng Chan, Hernan Aguirre and Kiyoshi Tanaka. "ArtGAN: Artwork Synthesis with Conditional Categorial GANs". In: *CoRR* abs/1702.03410 (2017).

[TBP01] Bernard Tiddeman, Michael Burt and David Perrett. "Prototyping and transforming facial textures for perception research". In: *IEEE Computer Graphics and Applications* 21.5 (2001), pp. 42–50.

[Tho+07] Vince Thomas, Nitesh V Chawla, Kevin W Bowyer and Patrick J Flynn. "Learning to predict gender from iris images". In: *Proceedings of Biometrics: Theory, Applications, and Systems*. Washington, USA, 2007.

[TKC11] Yingli Tian, Takeo Kanade and Jeffrey F Cohn. "Facial expression recognition". In: *Handbook of face recognition*. 2011, pp. 487–519.

[TP13] Juan E Tapia and Claudio A Perez. "Gender classification based on fusion of different spatial scale features selected by mutual information from histogram of LBP, intensity, and shape". In: *Transactions on Image Forensics and Security* 8.3 (2013), pp. 488–499.

[TPW17] Yaniv Taigman, Adam Polyak and Lior Wolf. "Unsupervised cross-domain image generation". In: *Proceedings of International Conference on Learning Representations*. Toulon, France, 2017.

[UHK06] Kazuya Ueki, Teruhide Hayashida and Tetsunori Kobayashi. "Subspace-based age-group classification using facial images under various lighting conditions". In: *Proceedings of Automatic Face and Gesture Recognition*. Malaga, Spain, 2006.

[Uri+15] Michal Uricár, Vojtech Franc, Diego Thomas, Akihiro Sugimoto and Václav Hlavác. "Real-time multi-view facial landmark detector learned by the structured output SVM". In: *Proceedings of Automatic Face and Gesture Recognition*. Ljubljana, Slovenia, 2015.

[Uři+16a] Michal Uřičář, Vojtěch Franc, Diego Thomas, Akihiro Sugimoto and Václav Hlaváč. "Multi-view facial landmark detector learned by the Structured Output SVM". In: *Image and Vision Computing* 47 (2016), pp. 45–59.

[Uři+16b] Michal Uřičář, Radu Timofte, Rasmus Rothe, Jiří Matas and Luc Van Gool. "Structured output SVM prediction of apparent age, gender and smile from deep features". In: *Proceedings of Computer Vision and Pattern Recognition Workshops*. Las Vegas, USA, 2016.

[Vie+14] Tiago F Vieira, Andrea Bottino, Aldo Laurentini and Matteo De Simone. "Detecting siblings in image pairs". In: *The Visual Computer* 30.12 (2014), pp. 1333–1345.

[Vin+15] Oriol Vinyals, Alexander Toshev, Samy Bengio and Dumitru Erhan. “Show and tell: A neural image caption generator”. In: *Proceedings of Computer Vision and Pattern Recognition*. Boston, USA, 2015.

[VJ01] Paul Viola and Michael Jones. “Rapid object detection using a boosted cascade of simple features”. In: *Proceedings of Computer Vision and Pattern Recognition*. Hawaii, USA, 2001.

[Wan+10] Jian-Gang Wang, Jun Li, Wei-Yun Yau and Eric Sung. “Boosting dense SIFT descriptors and shape contexts of face images for gender recognition”. In: *Proceedings of Computer Vision and Pattern Recognition Workshops*. San Francisco, USA, 2010.

[Wan+16] Wei Wang, Zhen Cui, Yan Yan, Jiashi Feng, Shuicheng Yan, Xiangbo Shu and Nicu Sebe. “Recurrent face aging”. In: *Proceedings of Computer Vision and Pattern Recognition*. Las Vegas, USA, 2016.

[WEG87] Svante Wold, Kim Esbensen and Paul Geladi. “Principal component analysis”. In: *Chemometrics and intelligent laboratory systems* 2.1-3 (1987), pp. 37–52.

[Wer74] Paul Werbos. “Beyond regression: New tools for prediction and analysis in the behavioral sciences”. In: (1974).

[WGK15] Xiaolong Wang, Rui Guo and Chandra Kambhamettu. “Deeply-learned feature for age estimation”. In: *Proceedings of Winter Conference on Applications of Computer Vision*. Hawaii, USA, 2015.

[Wu+16] Qi Wu, Peng Wang, Chunhua Shen, Anthony Dick and Anton van den Hengel. “Ask me anything: Free-form visual question answering based on knowledge from external sources”. In: *Proceedings of Conference on Computer Vision and Pattern Recognition*. Las Vegas, USA, 2016.

[WZ95] Ronald J Williams and David Zipser. “Gradient-based learning algorithms for recurrent networks and their computational complexity”. In: *Backpropagation: Theory, architectures, and applications* 1 (1995), pp. 433–486.

[Xia+09] Bo Xiao, Xiaokang Yang, Hongyuan Zha, Yi Xu and Thomas Huang. “Metric learning for regression problems and human age estimation”. In: *Advances in Multimedia Information Processing* (2009), pp. 88–99.

[Xio+15] Hui Y Xiong, Babak Alipanahi, Leo J Lee, Hannes Bretschneider, Daniele Merico, Ryan KC Yuen, Yimin Hua, Serge Gueroussov, Hamed S Najafabadi, Timothy R Hughes et al. “The human splicing code reveals new insights into the genetic determinants of disease”. In: *Science* 347.6218 (2015).

[XLS08] Ziyi Xu, Li Lu and Pengfei Shi. “A hybrid approach to gender classification from face images”. In: *Proceedings of International Conference on Pattern Recognition*. Florida, USA, 2008.

[XLS12] Yiting Xie, Khoa Luu and Marios Savvides. “A robust approach to facial ethnicity classification on large scale face databases”. In: *Proceedings of Biometrics: Theory, Applications, and Systems*. Washington, USA, 2012.

[XSL08] Bin Xia, He Sun and Bao-Liang Lu. "Multi-view gender classification based on local Gabor binary mapping pattern and support vector machines". In: *Proceedings of International Joint Conference on Neural Networks*. Hong Kong, China, 2008.

[YA07] Zhiguang Yang and Haizhou Ai. "Demographic classification with local binary patterns". In: *Proceedings of International Conference on Biometrics*. Seoul, South Korea, 2007.

[Yan+15a] H-F Yang, Lin B-Y, Chang K-Y and Chen C-S. "Automatic Age Estimation from Face Images via Deep Ranking". In: *Proceedings of British Mashine Vision Conference*. Swansea, UK, 2015.

[Yan+15b] Xu Yang, Bin-Bin Gao, Chao Xing, Zeng-Wei Huo et al. "Deep label distribution learning for apparent age estimation". In: *Proceedings of International Conference on Computer Vision Workshops*. Santiago, Chile, 2015.

[Yan+16] Xinchen Yan, Jimei Yang, Kihyuk Sohn and Honglak Lee. "Attribute2image: Conditional image generation from visual attributes". In: *Proceedings of European Conference on Computer Vision*. Amsterdam, Netherlands, 2016.

[YBG15] Sonia Yousfi, Sid-Ahmed Berrani and Christophe Garcia. "Deep learning and recurrent connectionist-based approaches for Arabic text recognition in videos". In: *Proceedings of International Conference on Document Analysis and Recognition*. Nancy, France, 2015.

[Yeh+16] Raymond Yeh, Chen Chen, Teck Yian Lim, Mark Hasegawa-Johnson and Minh N Do. "Semantic Image Inpainting with Perceptual and Contextual Losses". In: *CoRR* abs/1607.07539 (2016).

[Yi+14] Dong Yi, Zhen Lei, Shengcai Liao and Stan Z Li. "Learning face representation from scratch". In: *CoRR* abs/1411.7923 (2014).

[YLL14] Dong Yi, Zhen Lei and Stan Z Li. "Age estimation by multi-scale convolutional network". In: *Proceedings of Asian Conference on Computer Vision*. Singapore, 2014.

[Zeb97] Leslie A Zebrowitz. *Reading faces: Window to the soul?* 1997.

[ZF14] Matthew D Zeiler and Rob Fergus. "Visualizing and understanding convolutional networks". In: *Proceedings of European Conference on Computer Vision*. Zurich, Switzerland, 2014.

[Zha+03] Wenyi Zhao, Rama Chellappa, P Jonathon Phillips and Azriel Rosenfeld. "Face recognition: A literature survey". In: *ACM computing surveys (CSUR)* 35.4 (2003), pp. 399–458.

[Zha+07] Lun Zhang, Rufeng Chu, Shiming Xiang, Shengcai Liao and Stan Z Li. "Face detection based on multi-block lbp representation". In: *Proceedings of International Conference on Biometrics*. Seoul, Korea, 2007.

[Zho+05] Shaohua Kevin Zhou, Bogdan Georgescu, Xiang Sean Zhou and Dorin Comaniciu. "Image based regression using boosting method". In: *Proceedings of International Conference on Computer Vision*. Beijing, China, 2005.

[Zho+14] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba and Aude Oliva. “Learning deep features for scene recognition using places database”. In: *Proceedings of advances in Neural Information Processing Systems*. Montreal, Canada, 2014.

[Zhu+15] Yu Zhu, Yan Li, Guowang Mu and Guodong Guo. “A Study on Apparent Age Estimation”. In: *Proceedings of International Conference on Computer Vision Workshops*. Santiago, Chile, 2015.

[Zhu+16] Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman and Alexei A Efros. “Generative visual manipulation on the natural image manifold”. In: *Proceedings of European Conference on Computer Vision*. Amsterdam, Netherlands, 2016.

[Zhu+17] Jun-Yan Zhu, Taesung Park, Phillip Isola and Alexei A Efros. “Unpaired image-to-image translation using cycle-consistent adversarial networks”. In: *Proceedings of International Conference on Computer Vision*. Venice, Italy, 2017.

[ZMZ11] Huiyu Zhou, Paul C Miller and Jianguo Zhang. “Age classification using Radon transform and entropy based scaling SVM.” In: *Proceedings of British Mashine Vision Conference*. Dundee, UK, 2011.

[ZSQ17] Zhifei Zhang, Yang Song and Hairong Qi. “Age Progression/Regression by Conditional Adversarial Autoencoder”. In: *Proceedings of Computer Vision and Pattern Recognition*. Honolulu, USA, 2017.

[ZY10] Yu Zhang and Dit-Yan Yeung. “Multi-task warped gaussian process for personalized age estimation”. In: *Proceedings of Computer Vision and Pattern Recognition*. San Francisco, USA, 2010.