Universitat
de les Illes Balears

DOCTORAL THESIS
2019

FACIAL DETECTION AND EXPRESSION RECOGNITION APPLIED TO SOCIAL ROBOTS

Silvia Ramis Guarinos

Universitat
de les Illes Balears

DOCTORAL THESIS
2019

Doctoral Programme in Information and Communications Technology

FACIAL DETECTION AND EXPRESSION RECOGNITION APPLIED TO SOCIAL ROBOTS

Silvia Ramis Guarinos

Supervisor: Francisco J. Perales López
Supervisor: Jose Maria Buades Rubio
Supervisor: Jose Luis Lisani Roca
Tutor: Javier Varona Gómez

Doctora per la Universitat de les Illes Balears

Silvia Ramis Guarinos
Facial Detection and Expression Recognition applied to Social Robots
July 2019
Supervisors: Dr. Francisco J. Perales López, Dr. Jose Maria Buades Rubio
and Dr. Jose Luis Lisani Roca

Universitat de les Illes Balears
Departament de Ciències Matemàtiques i Informàtica
UGIVIA Research group

Universitat
de les Illes Balears

I, Silvia Ramis Guarinos, declare that this thesis titled “*Facial Detection and Expression Recognition applied to Social Robots*” and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a Ph.D. degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this university or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.

For all intents and purposes, I hereby sign this document.

Signed: Silvia Ramis Guarinos

Palma, July 2019.

UIB

Universitat
de les Illes Balears

Dr. Francisco J. Perales López, Dr. Jose Maria Buades Rubio and Dr. Jose Luis Lisani Roca of the Universitat de les Illes Balears, declare that the thesis titled "*Facial Detection and Expression Recognition applied to Social Robots*", presented by Silvia Ramis Guarinos to obtain a doctoral degree, has been completed under our supervision and meets the requirements to opt for an International Doctorate.

For all intents and purposes, we hereby sign this document.

Perales

Signed: Dr. Francisco J. Perales López

Palma, July 2019.

BUADES

Signed: Dr. Jose Maria Buades Rubio

Palma, July 2019.

Jose Luis Lisani

Signed: Dr. Jose Luis Lisani Roca

Palma, July 2019.

Publications and contributions

Journals

Part of the results presented in this PhD dissertation has been published. On the one hand, the following journal articles arise from the main contributions of the work herein presented.

1. Lisani, J. L., Ramis, S., & Perales, F. J. (2017). A Contrario Detection of Faces: A Case Example. *Siam Journal On Imaging Sciences*. 10 - 4, pp. 2091 - 2118. (USA). ISSN 1936-4954.

2. Lisani, J. L., & Ramis, S. (2019). A Contrario Detection of Faces with a Short Cascade of Classifiers (submitted, in review). *IPOL Journal · Image Processing On Line*.

On the other hand, the PhD candidate has been involved in related projects that led to additional journal articles and that, thus, represent a relevant part of her research activity.

3. Baldassarri, S., Passerino, L., Ramis, S., Perales, F. J., & Riquelme, I. (2019). Towards emotional interactive videogames for children with Autism Spectrum Disorder (accepted). *Universal Access In The Information Society*. (Germany). ISSN 1615-5289.

4. Perales, F. J, Riera, L., Ramis, S., & Guerrero, A. (2019). Evaluation of a VR system for Pain Management using binaural acoustic stimulation. *Multimedia Tools and Applications*. (Holland). ISSN 1380-7501.

Proceedings and Book Chapters

The Phd candidate has also published in several proceeding and book chapters. On the one hand, the following article includes part of the results from the present work.

1. Ramis, S., Perales, F. J., Buades, J. M., & Guerrero, A. (2019). Interacción basada en robots sociales para la evaluación de expresiones faciales. *XX International Conference on Human-Computer Interaction. Donostia-SanSebastián*. Basque Country, June 25-28, 2019. (Spain).

On the other hand, although the next publications do not include results presented in this dissertation, they do include results produced by the PhD candidate in the context of works that the majority of which are closely related to it.

2. Baldassarri, S., Passerino, L., Ramis, S., Riquelme, I., & Perales, F. J. (2018). Videogame-based experiences for improving communication and attention in children with ASD. *XIX International Conference on HCI*. Palma, September 12-14, 2018. (Spain).

3. Perales, F. J., Sánchez, M., Riera, L., & Ramis, S. (2018). A Pilot Study: VR and Binaural Sounds for Mood Management. *International Conference Information Visualization (IV2018)*. Università degli Studi di Salerno, Salerno, July 10-13, 2018. (Italy).

4. Perales, F. J., Sánchez, M., Ramis, S., & Riera, L. (2018). A Virtual Reality system for Pain Management using acoustic stimulation and electrodermal evaluation. *Cognitive Area Networks*, vol. 5, nº1, June 2018, © Asociación Nicolo, ISSN: 2341-4243. (Spain).

5. Ramis, S., Perales, F. J., Campins, M., & Riquelme, I. (2017). Un videojuego serio para el estudio de expresiones faciales en personas con Autismo. *Cognitive Area Networks, 9º Simposio CEA de Bioingeniería, Interfaces Cerebro-Máquina. Neurotecnologías para la Asistencia y la Rehabilitación.* July 6-7, 2017. Institut Guttmann, Barcelona. ISSN 2341-4243. (Spain).

6. Bibiloni, T., Ramis, S., Oliver, A., & Perales, F. J. (2016). An Augmented Reality and 360-degreeVideo System to Access Audiovisual Content through Mobile Devices for Touristic Applications. *Applications and Usability of Interactive TV*. 605 -1, pp. 44 -58. Cham (Switzerland): Springer International Publishing Switzerland. ISBN 978-3-319-38906-6.

7. Ramis, S., Perales, F. J., Manresa, C., & Bibiloni, A. (2015). Usability Study of Gestures to Control a Smart TV. *Communications in Computer and Information Science*. (Germany): Springer Series. ISSN 1865-0929.

8. Ramis, S., Perales, F. J., & Bibiloni, T. (2015). Nuevas Interfaces de Acceso al Repositorio Audiovisual. *VI Interactive Digital TV Congress IV Iberoamerican Conference on Applications and Usability of Interactive TV*. (Spain).

9. Ramis, S., Perales, F. J., Manresa, C., & Bibiloni, A. (2014). Estudio de la usabilidad de gestos para el control de un Smart TV. *jAUTI 2014 III Jornadas de Aplicaciones y Usabilidad de la TVDi III Workshop TVDi Webmedia 2014*. 1 -1, pp. 152 -160. La Plata (Brazil). ISBN 9789503411889.

10. Ramis, S., Perales, F. J., & Bibiloni, T. (2014). Reconocimiento Facial e Identificación de Textos en Videos Interactivos. *jAUTI 2013 -II Jornadas Iberoamericanas de Difusión y Capacitación sobre Aplicaciones y Usabilidad de la TV Digital Interactiva*. pp. 210 -223. Córdoba (Spain). ISBN 978-84-697-0302-1.

11. Buades, J. M., González-Hidalgo, M., Perales, F. J., Ramis-Guarinos, S., Oliver, A., & Blanch, V. (2012). A New Parallelizable Deformation Method -Automatic Comparision between Foot and Last. *ICPRAM 2012-Proceedings of the 1st International Conference on Pattern Recognition Applications and Methods*, vol 1, Vilamoura, Algarve, Portugal, 6-8 February, 2012. 1, pp. 363 -369. Algarve (Portugal): SciTePress-Science and Technology Publications. ISBN 978-989-8425-98-0.

12. Buades, J. M., González-Hidalgo, M., Perales, F. J., Ramis-Guarinos, S., Oliver, A., & Montiel, E. (2012). A Fast Geometric Deformation Method to Adapt a Foot to a Platform. *Deformation Models: Tracking, Animation and Applications*. 7, pp. 121 -143. (Holland): Springer Netherlands. ISBN 978-94-007-5445-4.

Internships

1. University of Cagliari. Sardinia Italia. From 6th of September until 6th of December, 2017. (Grant of a pre-doctoral mobility aid for short stays, from the *Conselleria d'Innovació, Recerca i Turisme*).

Projects

1. National project "Juegos serios multimodales y robots sociales para la valoración de la atención". Universitat de les Illes Balears, Spain. IP: Francisco José Perales López. Financed by the *Ministerio de Economía y Competitividad (MINECO)*. TIN2015-67149-C3-2-R (01/01/2016- 31/12/2019).
2. OCDS project "Diseño de experiencias interactivas dirigidas al bienestar de personas con necesidades especiales". Universitat de les Illes Balears, Spain. IP: Cristina Manresa Yee. Financed by the *Oficina de Cooperació al Desenvolupament i Solidaritat (OCDS) de la UIB*. OCDS-CUD2016/13 (01/10/2016- 30/09/2018).

Also, the candidate has benefited from the fellowship FPI/1828/2015 of the *Conselleria d'Educació, Cultura i Universitats* of the *Govern de les Illes Balears* under an operational program co-financed by the European Social Fund.

Abstract

Facial expression is a non-verbal language that plays an important role in the communication, behaviour and interaction among humans. Recently, there has been a growing interest in the recognition of facial expressions in the field of Human-Robot Interaction (HRI). This interaction between robots and persons finds applications in different areas such as video surveillance, health care, road safety, etc.

This research work has progressed in three lines designed to improve this interaction: face detection, facial expression recognition, and their integration into a human-robot interaction system implemented in a social robot.

Face detection is the first step in a system in order to interact with a person. Many algorithms have been proposed for face detection. In this thesis a new method inspired on the classical Viola-Jones algorithm but using an *a contrario* statistical model in the detection step is presented. This method improves the accuracy of the original method, at a much lower computational cost.

Facial expression classification is performed with a convolutional neuronal network, after a pre-processing of the input face images and using different datasets for training and testing. The developed network has achieved a success rate close to that of humans. In addition, the human capacity to recognize expressions has been evaluated and the results have been compared with the ones obtained with the neural network.

Finally, an application with a social robot has been designed for the evaluation and validation of the proposed system in a real environment. The robot interacts with the user through a dynamic game where the player performs a series of facial expressions and the robot acts in response to the emotion expressed by the player.

Resumen

El reconocimiento de expresiones faciales es un lenguaje no verbal que determina un papel importante en la comunicación, comportamiento e interacción del ser humano. Recientemente ha surgido un gran interés en realizar reconocimiento de expresiones faciales en el ámbito de la Interacción Hombre-Robot (IHR). Dicha interacción entre robot y persona está orientada a diferentes ámbitos como vídeo-vigilancia, cuidados sanitarios, seguridad vial, detección de engaños, etc.

En este trabajo de investigación se ha avanzado en tres líneas encaminadas a mejorar dicha interacción: detección de caras, reconocimiento de la expresión facial, y un sistema de interacción hombre-robot implementado en el robot social.

La detección de caras es el primer eslabón de un sistema capaz de interactuar con una persona. Referente a este tópico existen numerosos algoritmos capaces de detectar el rostro. En esta tesis se presenta un nuevo método basado en el trabajo propuesto por Viola-Jones pero utilizando un modelo *a contrario*, el cual mejora la precisión de una cascada clásica, a un coste computacional mucho menor.

La clasificación de la expresión facial se lleva a cabo con una red neuronal convolucional, aplicando un pre-procesamiento y haciendo uso de diferentes bases de datos. Con ello se ha conseguido una tasa de acierto cercana a la del ser humano. Además, se ha evaluado la capacidad humana para reconocer las expresiones y se han contrastado los resultados con la red neuronal.

Por último, se ha diseñado una aplicación con el robot social para la evaluación y validación del sistema propuesto en un entorno real. El robot interactúa con el usuario a través de una dinámica de juego donde el jugador debe ir realizando expresiones faciales y el robot actúa en consecuencia a la emoción que ha expresado el jugador.

Resum

El reconeixement d'expressions facials és un llenguatge no verbal que té un paper important en la comunicació, comportament i interacció de l'ésser humà. Recentment ha sorgit un gran interès a realitzar reconeixement d'expressions facials a l'àmbit de la Interacció Home-Robot (IHR). Aquesta interacció entre el robot i la persona està orientada a diferents àmbits com la vídeo-vigilància, cures sanitàries, seguretat viària, detecció d'enganys, etc.

En aquest treball de recerca s'ha avançat en tres línies destinades a millorar la interacció: la detecció de cares, el reconeixement de l'expressió facial, i un sistema d'interacció home-robot implementat en el robot social.

La detecció de cares és el primer pas d'un sistema capaç d'interactuar amb una persona. Referent a aquest tema, hi ha nombrosos algoritmes capaços de detectar la cara. En aquesta tesi es presenta un nou mètode basat en el treball proposat per Viola-Jones però utilitzant un model "a contrario", el qual millora la precisió d’una cascada clàssica, a un cost computacional molt menor.

La classificació d'expressió facial es duu a terme amb una xarxa neuronal convolucional, aplicant un pre-processament i fent ús de diferents bases de dades. D’aquesta manera s'ha aconseguit una taxa d'encert propera a la de l'ésser humà. A més, s'ha avaluat la capacitat humana en reconèixer expressions facials i s'han contrastat aquests resultats amb la xarxa neuronal.

Finalment, s'ha dissenyat una aplicació amb el robot social per a l'avaluació i validació del sistema proposat en un entorn real. El robot interactua amb l'usuari a través d'una dinàmica de joc, on el jugador ha d'anar realitzant expressions facials i el robot actua en conseqüència a l'emoció que s’ha expressat.

Agraïments

Voldria donar les gràcies a totes aquelles persones que han estat al meu costat durant la Tesi i al llarg de la meva vida. Primerament donar les gràcies als meus directors Paco, Jose María i José Luis, per tota la seva dedicació, consell i ànims que m'han donat. Al grup UGIVIA on m'he sentida acollida des del primer moment.

A tots els meus companys de laboratori que sense ells, el món no seria el mateix, ni els berenars. Especialment a na Xisca i en Pedro, qui varen començar aquest viatge amb jo. A na Cris i en Ramon per tot el seu suport, carinyo i ànims. Vull també donar les gràcies als meus companys de departament, amb qui també he gaudit de molts bons moments i he trobat molt bones amistats. També donar les gràcies als companys de la *Università degli Studi di Cagliari*, per la seva hospitalitat durant la meva estada a l'illa.

I finalmet, donar les gràcies a la meva família, per tota la paciència que han tingut i per animar-me sempre amb tot el que faig. També vull donar gràcies als meus amics de sempre, que m'han donat ànims en els moments baixos. En especial a na Marta, qui ha estat al meu costat quan més ho he necessitat.

Contents

A tots als meus companys, amics i família.

Chapter 1

Introduction

In the last decade, the technology has achieved big advances in many fields, but especially in the field of artificial intelligence. Artificial Intelligence is the discipline that tries to simulate human intelligence processes. Within this field, the recognition of facial expressions entails a great challenge for many researchers, since the same expression among different people can vary according to ethnicity, age or gender. Even an expression of the same person can be interpreted in different ways depending on environment parameters (brightness, background and posture).

Facial expression is a non-verbal language which plays an important role in communication, behaviour and understanding among people. A facial expression involves a physical component of morphological changes in a face [8]. These changes in the face convey the emotional state of an individual and give us social information that we can use in many fields as human-computer interaction (HCI), health-care, surveillance, driver safety, deceit detection, etc. [29]. For example, we can apply the facial expression recognition in order to measure the level of satisfaction about a commercial product. In this way the seller could conduct a marketing study in real time. Another interesting application is to recognize when a driver is falling asleep or is angry. Both situations can lead to a traffic accident and the facial expression recognition could help to prevent these situations.

Recently, it has emerged a growing interest in incorporating facial expression recognition capabilities in social robots, since the emotions play an important role in human-robot interaction. Human-Robot Interaction (HRI) is a multidisciplinary field with contributions from HCI, artificial intelligence, robotics, natural language understanding, design, and social sciences [25].

For a good interaction between humans and robots, robots must be able to recognize, interpret and respond effectively to social signals from a human. A person's affection is a complex combination of emotions, moods, interpersonal postures, attitudes and personality traits that influence the behaviour of other persons [84]. A robot that is able to interpret emotions will have an improved capacity to take decisions and help humans [78]. These robots would promote more effective and attractive interactions with users and lead to better acceptance by users [90].

A recent survey [66] classifies and defines the social interactions between a human and a robot in several ways: collaborative HRI, assistive HRI, mimicry HRI, and general HRI (for example, multipurpose). The collaborative HRI involves a robot and a person working together to complete a common task. The robot must be able to identify the emotional state of a person to improve team performance. The assistive HRI includes robots that provide physical, social and / or cognitive assistance. For example, assistive robots are used in autism therapy [83]. Another field where assistive robots are used is with the elderly [70]. The mimicry HRI consists of a robot or person that imitates the verbal and/or non-verbal behaviours of the other. Finally, the general or multipurpose HRI are robots designed to involve people who use bidirectional communication for several applications.

Given the growing interest in Human-Robot Interaction [87] and the importance of facial expression recognition in this field, we have created an advanced interaction system using a social robot. The objectives and main contributions of this thesis are described in Section 1.1. The organization of the document is described in Section 1.2.

1.1 Objectives and Contributions of this Thesis

The main goal of this Thesis is to design, develop and validate a system which is able to detect the face of a person and recognize his/her facial expression in the wild, since in the real world an application must be able to work well in multiple scenarios. In Figure 1.1, we show the general scheme of this work, which has three main objectives.

Objective 1. *Face detection.*

Face detection is a process which consists in locating the faces in digital images. In Human-Computer Interaction, Computer Vison and, more recently, Human-robot Interaction, this process is an initial step in order to develop applications related with these fields. This step has a critical role, since if the face detector fails, the whole system fails.

Although this field has been widely studied in the last two decades, most of the improvements proposed in the literature have been focused on the training step of the face detection algorithm, but little attention has been paid to the detection step. We will delve more deeply in this step and we will show that its improvement allows a faster detection of faces and thus a more fluid interaction between human and machine.

Objective 2. *Facial expression recognition.*

Automatic facial expression recognition is still a very difficult task. A person’s face can appear differently depending on brightness conditions, background and posture. Even humans have difficulties in identifying facial expressions when these factors are modified, as we shall show in our experiments. Recent deep learning-based approaches have been proposed to improve the overall performance recognition of the six basic expressions (happiness, sadness, anger, surprise, fear and disgust). However, these models exhibit significant limitations when faces are captured using settings different from the ones used to obtain the training images ("cross-datasets" problem), leading to a performance drop of up to 49%.

In this thesis we propose a Convolutional Neural Network (CNN) for the classification of human emotions and perform a thorough analysis of its performance for several variations of training and test sets, using several datasets. Two of these datasets have been created exclusively for this thesis. We show that our results compare favorably to the ones obtained with other published methods.

On the other hand, it is not yet clear how much the facial image pre-processing step can impact on the final performance. It is also not clear whether the obtained results are correlated with those obtained by humans. All these points are investigated in this thesis.

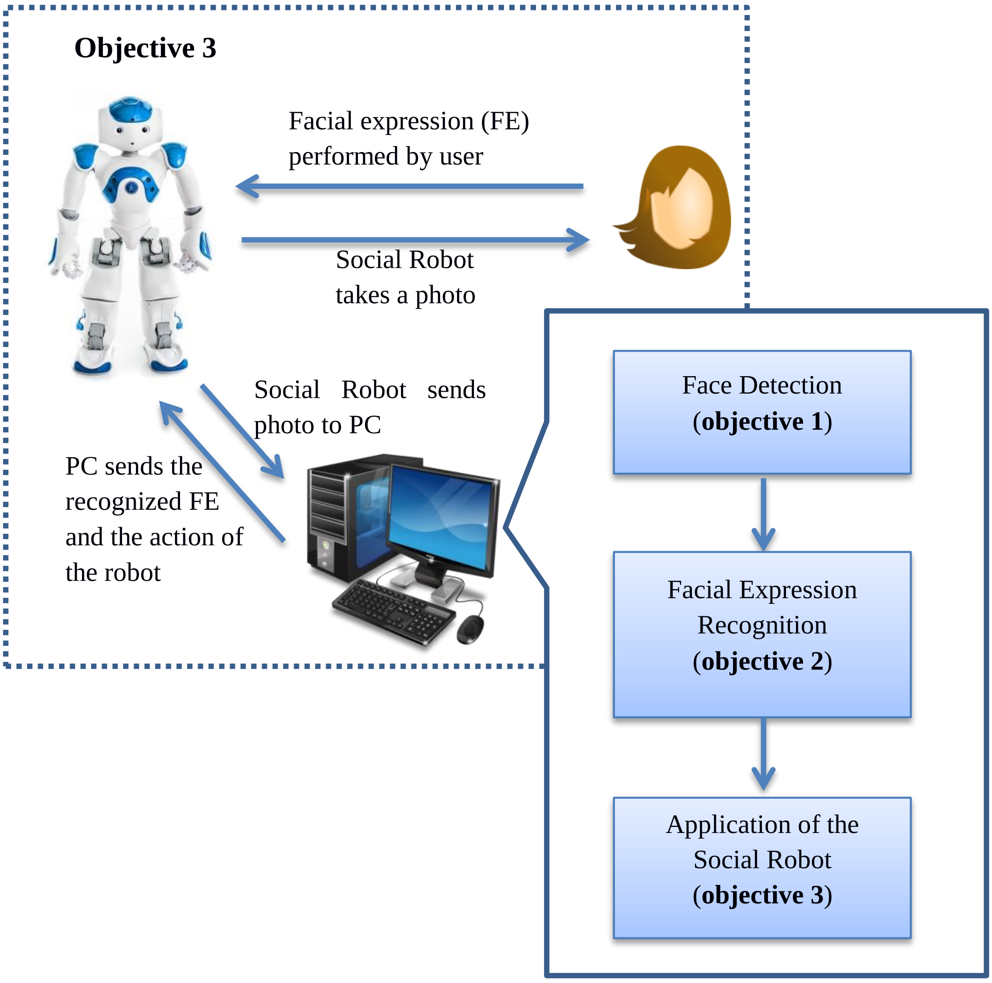

Objective 3
Facial expression (FE) performed by user
Social Robot takes a photo
Social Robot sends photo to PC
PC sends the recognized FE and the action of the robot
Face Detection (objective 1)
Facial Expression Recognition (objective 2)
Application of the Social Robot (objective 3)

Figure 1.1. General scheme of the proposed system. Each step of the system is related with an objective of this Thesis.

Objective 3. An application on a Social Robot to validate the system.

Social Robots are designed to interact with humans in a natural way and they must be able to work well with all kinds of users and situations. For this reason, facial expression recognition plays an important role in social robots.

This third step of the thesis consists in the evaluation and validation of the developed face detector and expression recognition in a real environment using a social robot. The robot will interact with a set of non-expert users, since it must work well while playing and interacting with them. This step will also permit to evaluate the interaction and the attention of each participant in designed application.

1.2 Organization of the Thesis

This thesis document is organized as follows. Chapter 2 introduces the basic concepts and reviews the more relevant literature related to the three objectives described in Section 1.1.

Chapter 3 presents our novel approach to face detection, based on an *a contrario* model of the face detection step. We show that an *a contrario* formulation can be adapted to the face detection method described by Viola and Jones in their seminal work. We propose an alternative to the cascade of classifiers proposed by the authors by introducing a stochastic *a contrario* model for the detections of a single classifier, from which adaptive detection thresholds may be inferred. The result is a single classifier whose detection rates are similar to those of a cascade of classifiers. Moreover, we show how a very short cascade of classifiers can be constructed, which improves the accuracy of a classical cascade, at a much lower computational cost.

Chapter 4 introduces a pre-processing algorithm of facial images, and a Convolutional Neural Network-based model for facial expression recognition. We perform a set of experiments which include~~s~~ widely known benchmark datasets, in addition to two newly created datasets. One of them is labeled, for the first time to our knowledge, with age, gender and facial expression data in order to facilitate the work in multimodal systems which combine these three types of information. The other dataset is a set of images in the wild to test the system. Results using the proposed system show an improvement in cross-datasets facial expression recognition, in addition to showing competitive results with respect to existing deep-learning approaches in the literature. Finally, we asked 253 participants to classify the emotions in a set of test images. Results show a correlation between the results obtained with the participants and the deep neural networks. That is, the same types of facial expressions were misclassified by both.

Chapter 5 presents an advanced interaction system based on a social robot which allows users to replicate and learn in a playful way the basic facial expressions. The Convolutional Neural Network (CNN) from Chapter 4 has been used in the robot application. The system is able to evaluate the facial expression of the user in front of the robot. The evaluation has been performed with 29 non-expert participants. Also, this experiment allowed evaluating the interaction, the attention and the difficulty to express an emotion through a final interview with each participant. This information is relevant to us since one application of the developed system is to encourage attention and motivation of users, especially people with special needs. Finally, the results obtained by the CNN were also compared with the ground truth provided by

10 experts in facial expression recognition, in order to validate the system.

We conclude this thesis summarizing the main contributions of the developed work and with suggestions for future research lines where the proposed system can prove its utility.

Chapter 2

Related Work

In this Chapter we review the more relevant literature related to the three problems tackled in this thesis. In Section 2.1 a summary of face detection methods is provided. Facial Expression Recognition methods are reviewed in Section 2.2. And, finally, Section 2.3 gives an overview of recent works on human-robot interaction.

2.1 Face detection

Face detection techniques have been development since the seventies when the first algorithms were implemented. These algorithms were very sensitive to image changes and unreliable. Therefore, this research was deserted, since the low storage capacity and the low computation power of the computers did not allow advancing in this field until the nineties, with the advent of faster processors. But it not was until 2001 when the researchers Viola and Jones proposed an algorithm for frontal face detection that settled the basis of most current face detection methods. The authors utilized the Integral Image technique, Haar-like features, a cascade classifier and the AdaBoost algorithm to construct an accurate classifier combining plenty of weak classifiers. Many improvements have been proposed from this work. In [55] it was introduced a novel set of rotated Haar-like features in order to obtain a rapid object detection scheme. The authors showed that the overall performance could be improved by about 23.8%. In [51] it was proposed a new boosting algorithm, called FloatBoost, to eliminate weak classifiers which cause higher error rates. In this way they were able to train a cascade with fewer weak classifiers and with lower error rates than AdaBoost. In [39] the algorithm proposed by Viola-Jones [99] was extended to handle profile views and rotated faces. In [107] the authors proposed a multiview face detection method invariant in to rotations using Real AdaBoost algorithm. Real AdaBoost is an algorithm used to boost the weak classifiers and construct a nesting-structured face detector. The work in [54] introduced a novel set of rotated Haar-like features and presented an analysis of different boosting algorithms (Discrete, Real and Gentle AdaBoost).

Recently, other methodologies have been proposed in order to improve the performance, such as [48, 52, 80]. The work in [48] combined Histograms of Orientated Gradient (HOG) features with linear Support Vector Machine (SVM). In [52] it was proposed a cascade architecture using convolutional neural networks (CNNs) which have a very powerful discriminative capability and a high performance. In [80] they proposed joint training to achieve end-to-end optimization for CNN cascade.

On the other hand, works as [55, 88, 108] have been proposed as alternative ways of building the cascade of classifiers. In [88] the authors proposed an algorithm called WaldBoost. The researchers integrated AdaBoost for the measurement selection and the Wald's optimal sequential probability ratio test. In [108] it was proposed a method called "Dynamic Cascade" which was used to train with massive data sets and obtain an efficient face detector. However, other works such as [12] combined the face alignment with the detection. They observed that aligned face shapes obtained better features to classify the face. Recently, in [65] the researchers used an integral channel detector instead of using the integral image proposed in the Viola-Jones detector to improve the performance in face detection. We can find a comprehensive survey about face detection in [111], where it is reviewed the state of the art from the Viola-Jones Detector [99] to recent advances.

Majority of the works in this field have focused on the training step of the algorithm, but have paid little attention to the detection step. Jain and Learned-Miller proposed the only method in the literature that deals with the detection of faces using a pre-trained cascade [35]. These authors proposed to quickly adapt a classifier to a new set of test data without retraining the classifier or examining the original optimization criterion. Similar to [35], we propose in Chapter 3 an improvement of the original Viola-Jones method that focuses on the detection step, but using an *a contrario* approach (described in Chapter 3).

2.2 Facial Expression Recognition

Automatic facial expressions recognition is now a main area of interest within various fields such as computer science, medicine, and psychology. It is used to improve the human-computer interaction (HCI) [29] or more recently in human-robot interaction (HRI).

Several techniques have been proposed for facial expression recognition in the last decades. In [85], the researchers used techniques such as Bayesian networks, SVMs, and decision trees to evaluate the several promising machine learning algorithms for emotion detection. In [95] the facial expression classification was made with a Support Vector Machine. In [79], the authors investigate Gauss–Laguerre wavelets, which have rich frequency extraction capabilities, to extract texture information of various facial expressions. For each input image, the face area is localized first. Then, the features are extracted based on GL filters, and, finally, the KNN classification is used for expression recognition. In [86] the authors utilized principal component analysis (PCA) and independent component analysis (ICA) for global and local feature extraction, and a hierarchical classifier (HMM) to recognize the facial expression. In [76], Gabor feature extraction techniques were employed to extract thousands of facial features. An AdaBoost-based hypothesis is used to select a few hundreds of the numerous extracted features to speed up classification, and these are fed into a well-designed 3-layer neural network classifier trained by a back-propagation algorithm. In [97] it was proposed an algorithm for facial expression recognition by integrating curvelet transform and online sequential extreme learning machine (OSELM) with a radial basis function (RBF) hidden node having optimal network architecture.

More recently, deep learning methods have contributed to improve facial expression recognition, with works like [36, 60, 37, 82, 103]. In [36] the authors proposed a model based on single Deep Convolutional Neural Networks (DNNs), which contain convolution layers and deep residual blocks. A combination of CNN and a specific image pre-processing step for the task of emotion detection is proposed in [60], and a Hybrid Convolution-Recurrent Neural Network method for facial expression recognition (FER) in images is presented in [37]. In [82] it was evaluated the performance of Inception and VGG architectures, which are pre-trained for object recognition, and these are compared with VGG-Face, which is pre-trained for face recognition. In [103], an ensemble of convolutional neural networks is presented with probability-based fusion for facial expression recognition, where the architecture of each CNN is adapted by using the convolutional rectified linear layer as the first layer and multiple hidden maxout layers. In spite of achieving a significant progress in facial expression recognition, the majority of papers are focused on getting a method to improve current results in one or several datasets separately, but they do not solve the problem of cross-dataset evaluation. Some recent papers, as [67, 112] have studied this problem. In [67] it was proposed a deep neural network architecture to address the face expression recognition (FER) problem across multiple well-known standard face datasets. The authors evaluated the accuracy of the proposed deep neural network architecture in two different experiments: subject-independent and cross-dataset evaluation. In [112] the performance influence of fine-tuning with the cross-dataset approach was investigated. In order to perform this study, the VGGFace Deep Convolutional Network model (pre-trained for face recognition) was fine-tuned to recognize facial expressions. The cross-dataset experiments were organized so that one of the datasets was separated as test set and the others as training set, and each experiment was ran multiple times to ensure the robustness of the results.

Starting from these last emergent studies, we investigate to which extent the use of multiple sources in the CNN's training phase helps during the test phase (see Chapter 4). In this Chapter we use a combination of a convolutional neuronal network with a specific image preprocessing. We employ five different data sets, which are combined to train the CNN, in order to provide better performance under cross-dataset tests. In addition we verify how trustworthy our results are by comparing human and machine performance.

2.3 Human – Robot Interaction

The study of facial expression recognition is a very active field in the area of computer vision [47]. Computer Vision allows acquiring, process, analyzing and understanding the images taken from one or more cameras, both in real time and offline. Often computer vision and Human-Computer Interaction (HCI) or Human-Robot Interaction go hand in hand. HCI is a multidisciplinary field of study focusing on the design of computer technology and, especially in the interaction between humans and computers [10]. Human-Robot Interaction (HRI) is a field of study dedicated to understanding, designing, and evaluating robotic systems for use by or with humans [31]. It is a relatively new field compared with HCI. Therefore many techniques used in HCI are also used in HRI [31].

Social Robots have been the subject of a growing interest in the last decade. A social robot must be able to express and/or recognize emotions, communication with dialogue, use natural gestures, have personality and stablish social relationships. The humans prefer to interact with machines in the same way that they interact with other persons. These robots can be used as research platforms, toys, educational tools or as therapeutic aids [22].

An area of interest in social interaction is that of "robot as a persuasive machine" [21], that is, the robot can change the behaviour, feelings or attitudes of humans. An example would be to use the robot as a mediator in human-human interaction, as in the therapy of autism [104]. Another area is "the robot as an avatar" [77]. For example, a robot can be used to communicate, and must act socially to transmit information effectively.

In all these areas, emotions play an important role in human behaviour, communication and interaction. Emotions are complex and are often closely related to the social context [4]. In recent years, emotions have been used more and more in this field, as we can see in papers such as [105, 63, 96, 49, 10, 93].

In [105], the authors propose a system with three main steps; first an adaptive skin color extraction, second the localization of the face and facial parts, such as eyes and mouth. Third, they propose to learn an objective function from training data. Experimental evaluation got a recognition rate of 70% using the Cohn–Kanade facial expression dataset, and 67% in a robot scenario. In [63] the authors combine a method for facial expression recognition based on Active Appearance Models (AAM) with Eigen-faces dynamic face recognition. This method achieved a recognition rate of positive facial expressions (happy, surprise and anger) of about 85%. And a recognition rate of negative facial expressions (disgust, sadness and fear) of about 65%. The authors did not implement the system in a social robot, but they propose it as a future work.

On the other hand, in [96] it was presented a novel approach to imitate facial expressions, since imitating the facial expressions of another person is a significant signal within interpersonal communication. Other papers such as [49] presented an ethnographic study with 40 children from an elementary school. The participants interacted with a social robot, which was able to recognize and respond empathetically to some of the affective states of the children. The results suggested that the robot's empathic behaviour affected children in a positive way. Recently, another study [93] proposed a model for adaptive emotion expression using the NAO robot. The NAO robot was able to express these emotions through its voice, posture, full-body postures, eye colour and gestures. The experiment was performed with 18 children and two NAO robots. One of the robots was an affective robot and the other a non-affective robot. The results showed that children react more expressively and more positively to an affective robot than to a robot that does not display emotions.

All of the above mentioned researches demonstrate that Facial expression recognition plays an important role in recognizing and understanding human emotion by robots. In order to develop an advanced interaction system using a social robot, we focus this work in creating a system which is able to recognize facial expression recognition in the wild.

Chapter 3

Face Detection using an *a contrario* approach

A fundamental first step in a Human-Robot Interaction system is face detection. Although many face detection methods have been proposed in the last two decades, most of them focus on improving the training step of the method, but little attention has been paid to the detection step. In this Chapter we describe an statistical approach that permits to automatically adjust the detection threshold of the detector, providing good performance with a reduced computational cost.

Section 3.1 introduces the problem and the most relevant previous works. Section 3.2 is devoted to the analysis of the strong classifiers proposed by Viola and Jones. We propose a stochastic model for the values of the classifier corresponding to nonface detections, which we shall use for the *a contrario* detection of faces, as described in Section 3.3. The experiments in Section 3.4 show that a single classifier with 200 features and adaptive detection thresholds computed using the *a contrario* approach is able to compete with a full cascade. The experimental results have been obtained using three standard faces datasets: IMM, BioID, and FDDB. This last dataset contains faces taken under unconstrained capture conditions (so-called in-the-wild). In Section 3.5 we propose the use of a very short cascade of classifiers (just four stages) in combination with the adaptive threshold principle to improve the computation speed of our method. The experimental results with the short cascade, using the same datasets as in the previous Section, are presented in Section 3.6. Finally, some conclusions and future research lines are exposed in Section 3.7.

3.1 Introduction

The *a contrario* framework provides a statistical formulation of a perception principle that states that an observed structure should be considered perceptually meaningful only if it is rarely encountered in a random image. This general principle is sometimes called the non-accidentalness principle [106, 61]. In [18, 19] Desolneux, Moisan, and Morel lay the basis of the *a contrario* methodology, which permitted translation of this principle into an efficient tool.

It consists in, first, defining a noise model (also called the background or a contrario model) and then testing against it the existence of the observed structure. If the expected number of occurrences of such a structure in the model is very low, then the structure is deemed meaningful.

This framework has been used successfully to detect contours and lines in images [16, 100, 101], modes in one-dimensional histograms [17, 15], moving objects in video [58], changes in satellite images [57], etc. What we propose in this Chapter is to apply, for the first time to our knowledge, this approach to the detection of faces.

In 2001 Viola and Jones [99] proposed an algorithm for face detection that settled the basis of most current face detection methods. Since this seminal work many improvements have been proposed. In order to increase the performance of the detector for multiview face detection, the original set of Haar-like features was extended using rotated features [55], rectangular features [50], or diagonal filters [39]. In addition, other types of features were proposed to describe face appearance: local binary patterns and its variants [74, 1, 40], histograms of oriented gradients [13], integral channel features [65], etc.

The original AdaBoost learning algorithm was also replaced by alternative boosting techniques: RealBoost [107], GentleBoost [54], and FloatBoost [50]. Recently, more powerful and discriminative methodologies such as support vector machines (SVM) [48] and deep neural networks [46] have also been used to train the detectors. Moreover, several alternative ways of building the cascade of classifiers have been proposed, using different methods to determine the rejection thresholds [55, 88] or integrating knowledge from previous stages [108]. Also a number of detector structures have been used to extend the cascade to multipose/multiview face detection: parallel cascade [107], detector-pyramid [50], and decision trees [39, 23].

All of the above mentioned improvements have focused on the training step of the algorithm but little attention has been paid to the detection step. To our knowledge, the only method in the literature dealing with the detection of faces using a pretrained cascade was proposed by Jain and Learned-Miller in 2011 [35]. These authors propose to adapt the detection thresholds to the image contents in such a way that *reliable* face detections can be used to detect other *difficult-to-detect* faces in the same scene. Similarly to [35], we propose in this Chapter an improvement of the original Viola–Jones method that focuses in the detection step, but using an *a contrario* approach. We show that it is possible to improve the performance of the detector (i.e., increase the detection rates, keeping low the number of false detections and at a reduced computational cost) without the need of a long cascade of classifiers. We propose to replace the fixed detection thresholds of the classifiers, learned in the training step, by adaptive thresholds particular to each input image. Contrary to [35] we do not propose a threshold function but instead propose a constant threshold estimated from the set of detections values computed for the whole image.

3.2 Analysis of a single classifier

A face classifier is a mathematical function that takes as input a portion of an image (typically a rectangular subimage) and gives as output a numerical value (typically 1 or 0) indicating whether the subimage contains a face.

Viola and Jones [99] defined a series of subimage features (Haar-like features; see Figure 3.1) and used a learning set of frontal faces to train, with the AdaBoost algorithm, a classifier that combined K of these features. This *strong* classifier (in opposition to the *weak* classifier that uses a single feature) is defined as

$$h(x) = \begin{cases} 1, & \vartheta_{det}(x) \geq T, \\ 0, & otherwise \end{cases} \tag{3.1}$$

with

$$\vartheta_{det}(\mathrm{x}) = \sum_{k=1}^{\mathrm{k}} \alpha_k h_k(x) \tag{3.2}$$

and

$$T = \frac{1}{2} \sum_{k=1}^{\mathrm{k}} \alpha_k \tag{3.3}$$

where x is a subimage, K is the number of features of the classifier, $h_k(x)$ is the weak classifier[1] associated with feature k, and α_k is the *weight* of h_k in the final strong classifier. The detection threshold T is fixed and depends on the α_k values learned from the training set of images[2].

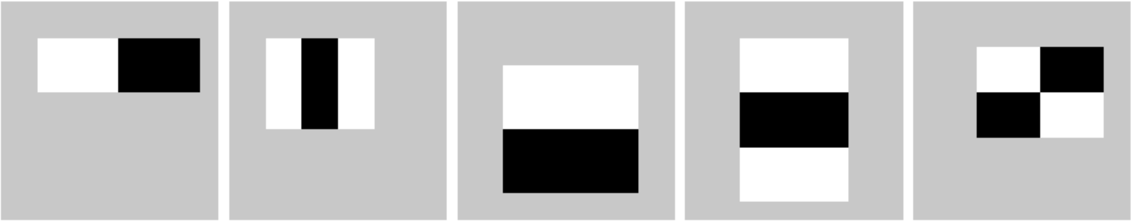


Figure 3.1. Haar-like feature masks used by the Viola–Jones detection method (Figures from [102]).

Usually, the detection value ϑ_{det} of the classifier is disregarded, since we are just interested in its binary response (1 for faces, 0 for nonfaces). In our study we take a different approach. We have analyzed the distribution of detection values (the set of detection values associated to all the tested subwindows in a particular image[3]) and several strong classifiers with different numbers of features. These classifiers have all been trained using the same set of frontal faces and Haar-like features used by Viola and Jones in their original paper [99]. It is important to

[1] $h_k(x) = 1$ if its associated feature, computed at subimage x, is above/below a learned threshold; otherwise $h_k(x) = 0$. The feature value is computed as the sum of intensity values in the "white" feature mask minus the sum of intensity values in the "black" feature mask. The masks associated to each feature are displayed in Figure 3.1.

[2] It must be remarked, however, that this dependence on α_k could be prevented by using values 1 and 1 in the weak classifiers. In this case, the threshold could be fixed to $T = 0$.

[3] In our implementation we have tested all the subwindows of sizes ranging from 20×20 to 220×220 pixels.

remark that flat image regions are not considered in the tests, i.e., detection values for subwindows whose standard deviation in intensity is small (in our implementation below 20) are not computed. The reason is that the standard deviation of each subwindow is normalized to a fixed value (50 in our implementation) before applying the detection masks, and if the initial standard deviation of the subwindow is too small the resulting normalized image displays visual artifacts caused by the excessive amplification of noise.

A fundamental requirement of the a contrario approach is the definition of a stochastic model (or noise model) for the data where the sought structure is not present and can be observed only by accident. This stochastic model is particular to each application of the framework. For example, for the detection of smooth contours [101] a noisy soft gradient model is proposed, or for the detection of modes in one-dimensional histograms a flat model can be used. In the case of face detection this stochastic model may be inferred by analyzing the response of the classifier to images that do not contain faces. In Figure 3.2 we display the distribution of detection values for classifiers with increasing number of features (10, 20, 40, 80, and 200) for two images without faces. The image on the left is a pure Gaussian noise image with standard deviation σ = 30. The image on the right is a natural image. In both cases we observe that, as the number of features increases, the distribution of detection values tends to a normal distribution.

We would like to quantify how well these distributions fit the Gaussian density function; however, since the number of samples is huge (536402 values for the first image and 5170933 values for the second one) typical normality tests (Kolmogorov–Smirnov, Anderson–Darling, Shapiro–Wilk, etc.) reject the normality assumption. This is due to the fact that for large amounts of data even small deviations from normality lead to a negative result of the test. An alternative is to use graphical plots to visually compare the distributions of values to a Gaussian distribution, the so-called normal quantile-quantile (Q-Q) plots[4]. Figure 3.3, left, shows the plot corresponding to the distribution of values in Figure 3.2, bottom right (200 features classifier). Observe that the values follow closely those of a Gaussian distribution.

Thus it seems that a Gaussian distribution could be used as a background model against which to test the existence of faces in the image, provided that the number of features in the detector is large enough. However, as the previous figure shows, the parameters of this Gaussian, namely, its mean and standard deviation depend on the image content. How can we infer these parameters given any input image, independently of the presence or absence of faces? Fortunately, even when the image does contain faces, the vast majority of the contributions to the histogram correspond to nonfaces subwindows. Therefore, we still obtain a Gaussian distribution of detection values, and its parameters may be computed from the image itself, as shown in Figure 3.4.

In order to test how general this Gaussian assumption is we have computed the distribution

[4] In the normal Q-Q plot the x-axis corresponds to values from a standard normal distribution $Z \sim N(0, 1)$. The y-axis corresponds to values from the normalized input distribution $\hat{\gamma} = \frac{\gamma - \mu_\gamma}{\sigma}$, where μY and σY are the mean and standard deviation of the input distribution Y . Each point (x, y) in the plot is computed as $(x, F_{\hat{}}^{-1}(FZ(x)))$, where F_* denotes the cumulative distribution function of $*$. The line y=x is also displayed for reference, being the ideal plot in the case of $\hat{Y}$ being a perfect standard normal distribution.

of detection values (for the 200-features classifier) for the 2845 images in the FDDB dataset [34]. Some of these images are displayed later in Figure 3.12. The normal Q-Q plot in Figure 3.3, right, shows the limits for the Q-Q plots of these images: all the plots are within the two red lines, while 99% of the plots are within the blue lines. The plot shows that, for all the images, most of the detection values[5] follow closely a normal distribution, which seems to confirm our assumption.

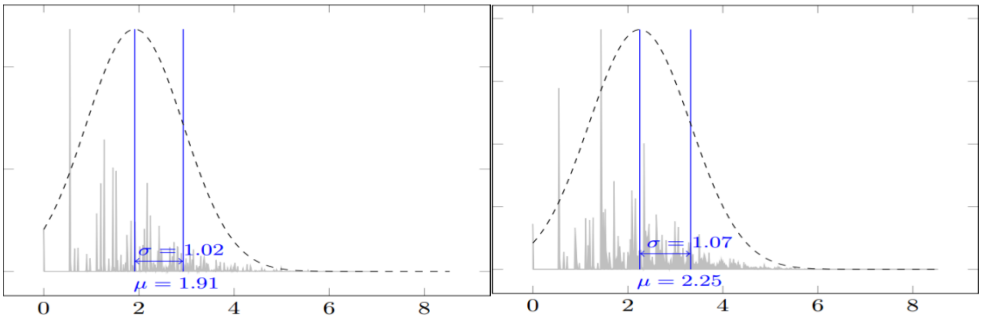
An alternative way to assess the Gaussianity of the distributions of detection values is by using the central limit theorem. Indeed, from (3.2) we can consider the detection value of a strong classifier with K features as a random variable of the form

$$V = Y_1 + Y_2 + \cdots + Y_K \qquad (3.4)$$

with $Y_k = \alpha_k X_k$, where α_k is the constant weight for the kth weak classifier and X_k is a random variable associated to this classifier. Note that X_k is a Bernoulli random variable, since it takes binary values 1 or 0 with unknown probabilities p_1^k and $p_0^k = 1 - p_1^k$. In its classical formulation the central limit theorem states that the sum of a large number of identically distributed independent random variables follows, approximately, a Gaussian law. More specifically, when $K \longrightarrow \infty$

$$\frac{V - E(V)}{\sqrt{VarV}} \xrightarrow{d} N(0,1)$$

where $\overset{\mathrm{d}}{\rightarrow}$ denotes convergence in distribution and N(0,1) is a normal random variable with mean 0 and variance 1.



[5] Recall that 95.45% of the values of a standard normal distribution are in the range (−2, 2); 99.73% of the values are in (−3, 3).

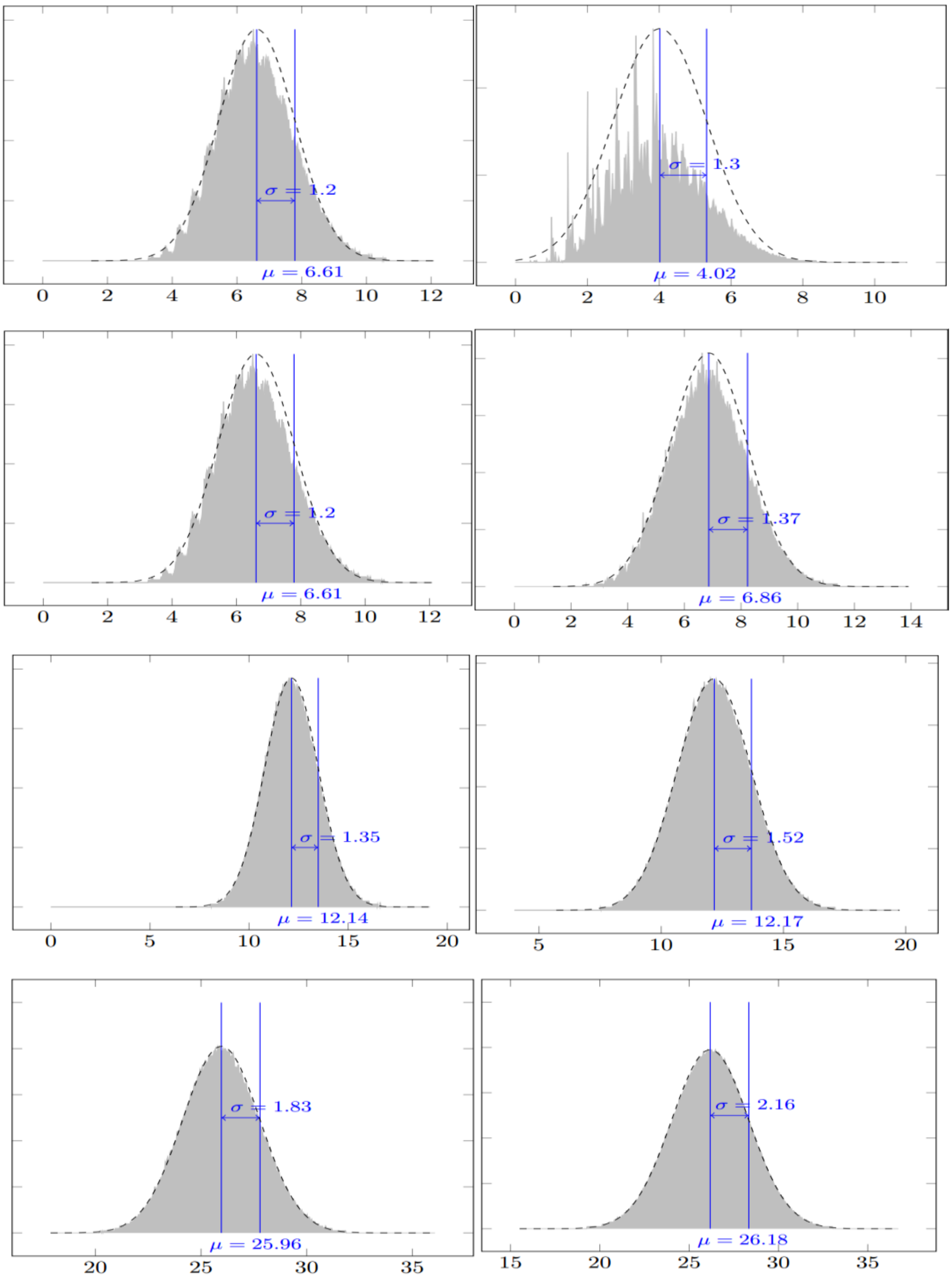


Figure 3.2. Distribution of detection values for classifiers with increasing numbers of features. From top to bottom: original image and histograms for classifiers with 10, 20, 40, 80, and 200 features. The mean μ and standard deviation σ of each distribution are shown, and the Gaussian function with the same mean and variance parameters is superimposed. For the left image a total of 536, 402 subwindows were checked by each classifier. For the right image 5170933 subwindows were checked.

In this case, we can assume that the Y_k are independent since they come from different weak classifiers, but the identical distribution of their values cannot be justified a priori. However, Lindeberg [56] proved that if the following condition is met, then the result of the central limit theorem holds, even if the variables are not identically distributed

$$\lim_{k\to\infty}\frac{1}{s_k^2}\sum_{i=1}^{k}E\left[(Y_i-\mu_i)^2\cdot\mathbf{1}_{\{|Y_i-\mu_i|\geq\varepsilon s_k\}}\right]=0,\quad \forall_{\varepsilon}>0, \qquad (3.5)$$

where E[·] denotes the expectation of a random variable, $\mathbf{1}_{\{\cdot\}}$ is the indicator function, µi = E[Y$_i$] and $s_k^2=\sum_{i=1}^{k}Var(Y_i)$.

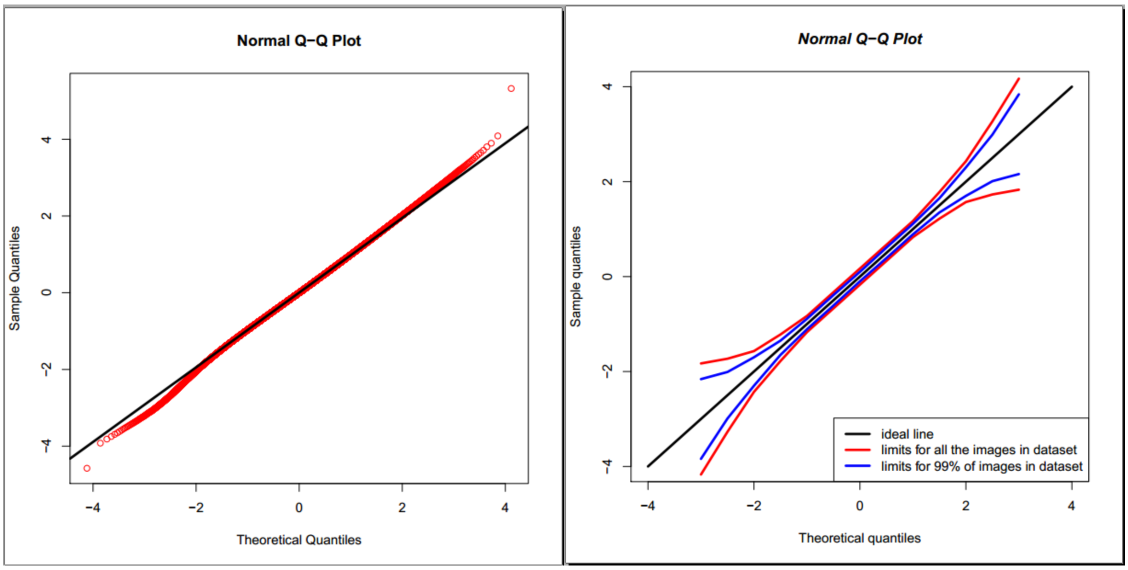


Figure 3.3. Left, normal Q-Q plot for the distribution of values in Figure 3.2, bottom right. Right, limits of the normal Q-Q plots for the 2845 images in the FDDB dataset [34].

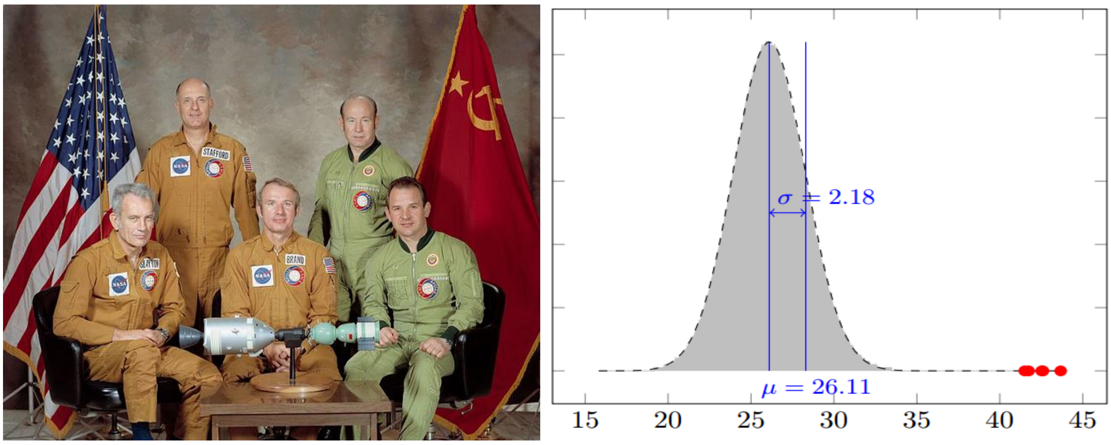


Figure 3.4. Left, original image. Right, distribution of detection values for a 200-features classifier. The red dots indicate the detection values for the subwindows actually containing a face. A total of 3426685 subwindows were checked by the classifier.

The following condition (Lyapunov's condition [3]) implies (3.5):

$$\lim_{k\to\infty} \frac{1}{s_k^{2+\delta}} \sum_{i=1}^{k} E\left[|Y_i - \mu_i|^{2+\delta}\right] = 0, \quad \delta > 0. \tag{3.6}$$

Indeed

$$\begin{aligned} E\left[|Y_i - \mu_i|^{2+\delta}\right] &\geq \; E\left[|Y_i - \mu_i|^{2+\delta} \cdot \mathbf{1}_{\{|Y_i - \mu_i| \geq \varepsilon s_k\}}\right] \\ &= E\left[|Y_i - \mu_i|^{2} \cdot |Y_i - \mu_i|^{\delta} \cdot \mathbf{1}_{\{|Y_i - \mu_i| \geq \varepsilon s_k\}}\right] \\ &\geq \varepsilon^{\delta} s_k^{\delta} E\left[(Y_i - \mu_i)^{2} \cdot \mathbf{1}_{\{|Y_i - \mu_i| \geq \varepsilon s_k\}}\right] \end{aligned}$$

Therefore

$$\frac{1}{s_k^2} \sum_{i=1}^{k} E\left[(Y_i - \mu_i)^2 \cdot \mathbf{1}_{\{|Y_i - \mu_i| \geq \varepsilon s_k\}}\right] \leq \frac{1}{s_k^2} \sum_{i=1}^{k} \frac{E\left[|Y_i - \mu_i|^{2+\delta}\right]}{\varepsilon^{\delta} s_k^{\delta}} = \frac{1}{\varepsilon^{\delta}} \frac{1}{s_k^{2+\delta}} \sum_{i=1}^{k} E\left[|Y_i - \mu_i|^{2+\delta}\right].$$

If condition (3.6) is met, then the previous inequality implies that the Lindeberg condition (equation (3.5)) is also met and therefore the central limit theorem holds for *V*.

In Figure 3.5 we show, empirically, that Lyapunov's condition is satisfied for $\delta = 1$ when *K* increases. The figure displays the average value of

$$\boldsymbol{r_k} = \frac{\mathbf{1}}{\boldsymbol{s_k^3}} \sum_{\boldsymbol{i=1}}^{\boldsymbol{k}} \boldsymbol{E}\left[|\boldsymbol{Y_i} - \boldsymbol{\mu_i}|^{\mathbf{3}}\right]. \tag{3.7}$$

over all the images in the FDDB dataset, for increasing values of *K*. We observe a decreasing trend in the value, which agrees with the Gaussianity hypothesis.

3.3 Improving the performance of a single classifier.

In the preceding Section we have shown, empirically, that the distribution of detection values for a single strong classifier tends to a Gaussian law when the number of features used by the classifier is large. Moreover, the parameters of this law (its mean and standard deviation) are different for each image. This empirical observation contradicts the way the detection threshold is chosen in the classical Viola–Jones detection scheme: the same threshold, fixed in the learning stage and computed with (3.3), is used for all the images. Note that this fixed threshold is optimal globally, though a local adjustment could improve the detector's performance. What we propose is to adapt the threshold to the particular distribution of detection values associated to each image.

Before detailing the method to adaptively select the detection threshold let us remark that the true positives of the detection process (i.e., the subimages containing the actual faces to be detected) have, in general, a very high detection value. This is to be expected provided that the classifier is discriminant enough (i.e., it is formed by a large number of weak classifiers). Figures 3.4 and 3.6 display the histograms of detection values for two images containing faces and for classifiers with 200 features and 80 features, respectively. The red dots indicate the detection values for the faces in the image. Observe that they are located in the far right end of the distribution. Moreover, in Figure 3.6 the position of the default detection threshold T computed with formula (3.3) is also displayed. It is clear from this figure that the use of the default detection threshold would produce a large number of false positives. We describe in the following paragraphs a method which permits us to reduce the number of false positives of a single classifier by computing a detection threshold adapted to the distribution of detection values.

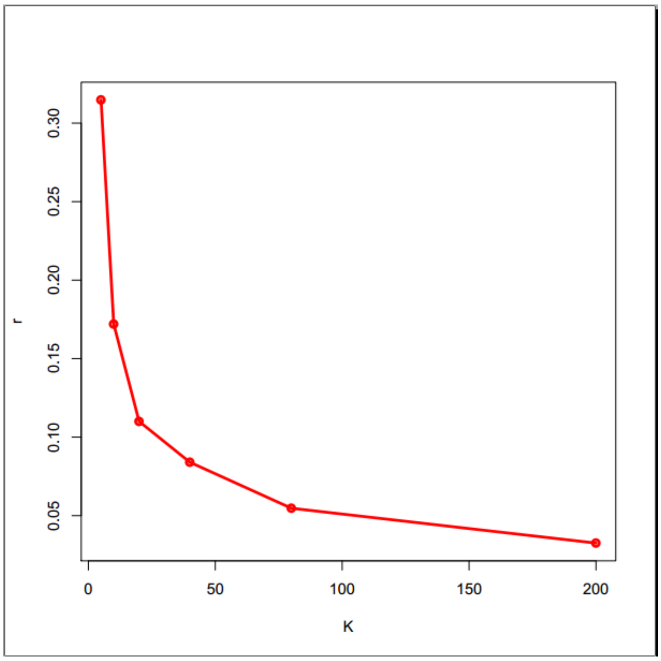


Figure 3.5. Evolution of the values of r_K (3.7) (average over all the images in the FDDB dataset) for increasing values of K (5, 10, 20, 40, 80, 200).

Following the *a contrario* detection principle we test the presence of a face in a subwindow against a noise or *a contrario* model where the face is not present. This is equivalent to performing the following hypothesis test:

H_0 (null hypothesis): the subimage ***does not contain*** a face

H_1 (alternative hypothesis): the subimage ***contains*** a face

The acceptation/rejection of H_0 depends on a rejection threshold θ and the *level of significance* α of the test is defined as

$$\alpha = P(rejecting\ H_0|H_0\ is\ true) = P(\vartheta_{det} > \theta|H_0\ is\ true) =$$

$$= P\left(\begin{array}{c|c} accepting & the\ subimage \\ subimage & does\ not\ contain \\ as\ face & a\ face \end{array}\right) = P(False\ positive)$$

where $\boldsymbol{\vartheta_{det}}$ is the detection value associated to the subimage, computed from (3.2).

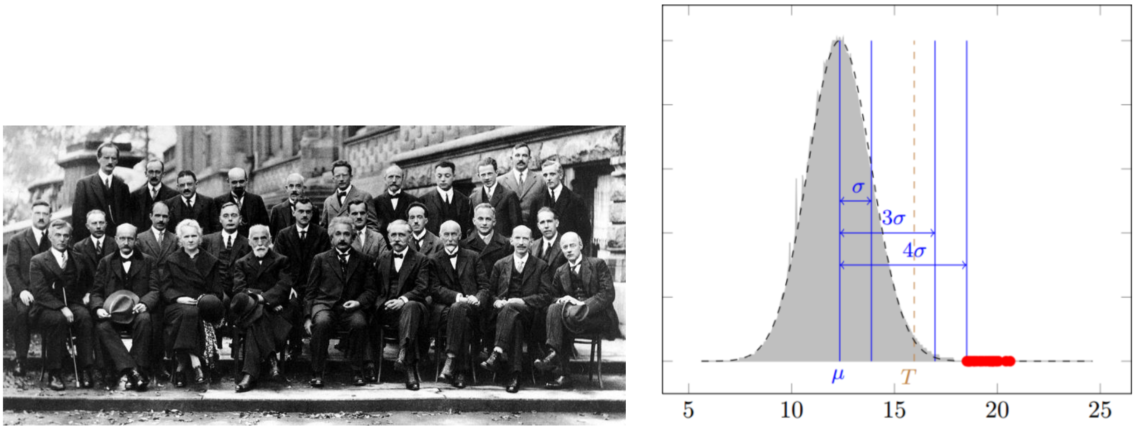


Figure 3.6. Input image and its histogram of detection values for an 80-features classifier. The red dots indicate the detection values for the subwindows actually containing a face. T is the default detection threshold of the classifier.

By applying the observations from the previous Section, we may assume a Gaussian distribution of the detection values for the null hypothesis (i.e., the distribution of detection values for the nonfaces subwindows is Gaussian). This allows us to compute the level of significance in closed form. The mean μ and standard deviation σ of this Gaussian can be estimated from the empirical values of the histogram. We are assuming here that just a small fraction of the subwindows in any image, if any, corresponds to actual faces. Therefore, the actual distribution of detection values for the whole image corresponds, roughly, to the distribution of values under the null hypothesis.

We first write the rejection threshold θ as a function of μ and σ: $\theta = \theta_s = \mu + s\sigma$, where s is a parameter. Then α can be expressed in terms of s:

$$\begin{aligned} \alpha &= P(False\ positive) \\ &= P\left(\vartheta_{det} > \theta_s \middle| H_0 \sim N(\mu, \sigma^2)\right) \\ &= \frac{1}{2} erfc\left(\frac{\theta_s - \mu}{\sqrt{2}\sigma}\right) = \frac{1}{2} erfc\left(\frac{s}{\sqrt{2}}\right) \end{aligned} \qquad (3.8)$$

where N denotes the Gaussian probability density function and erfc is the complementary error function. Note that θs is an adaptive threshold, since it depends on the detection statistics (μ and σ) of the input image.

3.3.1 Setting the detection threshold of the detector.

Equation (3.8) permits us to control the probability of false positives of a strong classifier. Table 3.1 displays this probability for the histogram in Figure 3.2, bottom right, and for different values of θ_s.

θ	***P* (False positive) (3.8)**
T (default, equation (3.3))	0.0024
$\theta_{4.0} = \mu + 4\sigma$	$3.16 \cdot 10^{-5}$
$\theta_{5.0} = \mu + 5\sigma$	$2.87 \cdot 10^{-7}$

Table 3.1. Probability of false positives computed with (3.8) for the histogram in Figure 3.2, bottom right, and for different values of the parameter s. The probability of a false positive for the default detection threshold is also displayed.

A question that arises is which optimum value of the parameter *s* guarantees such a low value of probability that no false positives are observed in the image. To answer this question we need first to establish the relation between the probability of false positives and the actual number of observed false positives. This relation is straightforward: if the number of tested subwindows in the image is *N*, then the *expected number of false positives*, *NFP*, can be computed as

$$NFP = N \times P(\text{False positive}). \qquad (3.9)$$

Table 3.2 displays the expected number of false positives for different values of the detection threshold for the image in Figure 3.2, top right, and a 200-features classifier. For this example N = 5170933 and the values of P (False positive) are the ones in Table 3.1. The observed number of false positives is also shown. Note the similarity between the expected and actual values, which confirms the validity of the Gaussian model in this case.

A criterion for the selection of the detection threshold is to compute the value of θ_s that guarantees a value of *NFP* below some predefined upper bound $NFP_{\max}$. Combining (3.9) and (3.8) we obtain the value of the detection threshold as

$$\theta = \mu + \sqrt{2} erfc^{-1}\left(\frac{2}{N} NFP_{max}\right)\sigma \qquad (3.10)$$

Figure 3.7 shows the result of applying this criterion to the image in Figure 3.6, left, using a detector of 200 features, for increasing values of $NFP_{\max}$ (1, 5, 10). Note that in the figure no postprocessing was used to display the results, and all the subwindows above the estimated detection threshold are displayed. In the next Section we shall discuss how to group together similar detections and display a single detection rectangle per face. As expected, when $NFP_{\max}$ is set to 1 no false positives are detected, but some faces are missed by the detector. As $NFP_{\max}$ increases more false positives appear but also more faces are detected.

Θ	***NFP* (estimated)**	***NFP* (observed)**
T (default, equation (3.3))	12416.35	12101
$\theta_{4.0} = \mu + 4\sigma$	163.76	183
$\theta_{5.0} = \mu + 5\sigma$	1.48	2

Table 3.2. Estimated and observed number of false positives for a 200-features classifier applied to the image in Figure 3.2, top right.

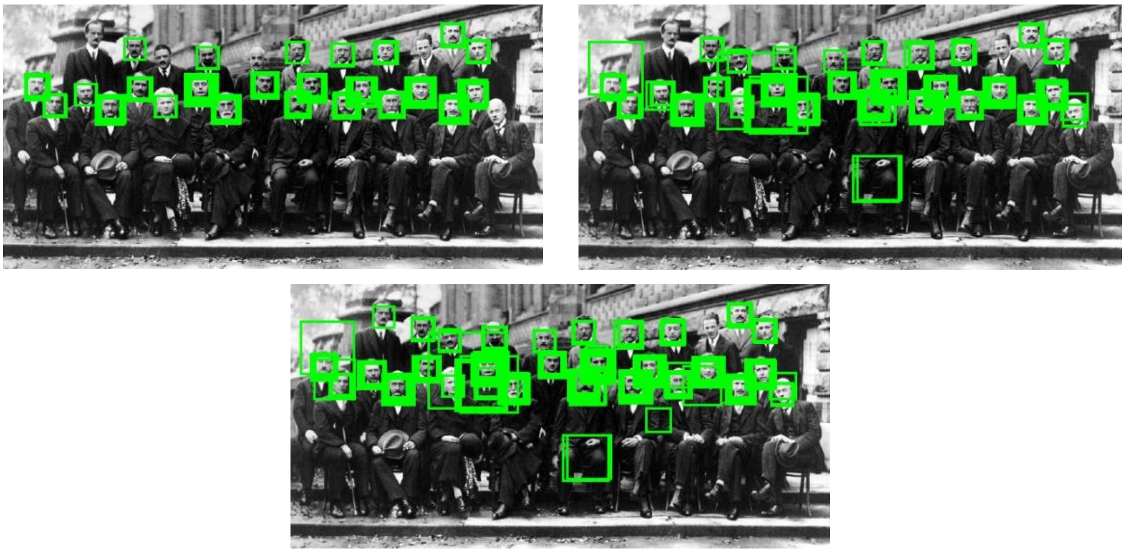


Figure 3.7. From left to right and from top to bottom: detections with NFPmax 1, 5, and 10, using a 200 features detector.

3.4 Experiments with a single classifier.

In this Section we analyze the performance of a single strong classifier whose detection threshold is adapted to the detection statistics of the input image, as described in the previous Section. The classifier uses 200 features and it has been trained with AdaBoost using the same set of 24x24 frontal faces and Haar-like features proposed by Viola and Jones in their original paper. We first discuss how to simplify the detection results and then we test the classifier on different standard datasets.

3.4.1 Simplifying the classifier output.

As shown in Figure 3.7 the raw output of the classifier is a set of squares that indicate the position of a detected face. Usually, for each true detection, many squares of similar sizes are found, centered around the detected face and forming a thick frame around it. Moreover, false detections do not usually exhibit such thick frames, meaning that the detection result is not very stable in this region of the image. All face detection methods apply some kind of postprocessing to these raw results in order to get just one representative for each group of similar detections. We propose two types of postprocessing:

1. First, detections are grouped according to the following principle: detections d_1 and d_2 are grouped together if

$$\frac{A(d_1 \cap d_2)}{\min(A(d_1), A(d_2))} > 0.5, \qquad (3.11)$$

where d_i denotes a square detection (region of the image that tested positive for the detector) and *A*() is the area operator. For each group of detections only the one with the highest detection value is kept. Figure 3.8 shows an example of the simplification provided by this grouping principle. Note that (3.11) permits us to group together detections of very different sizes, provided that they share at least 50% of their area (see Figure 3.8, center). This may seem weird, but underlying this criterion is the idea that two faces cannot be contained inside the same detection window; therefore only the one with the highest detection value is kept and the other is rejected. An alternative grouping criterion could be to group detections satisfying the 50% intersection condition but having similar sizes. But this would produce results such as the one displayed in Figure 3.8, right.

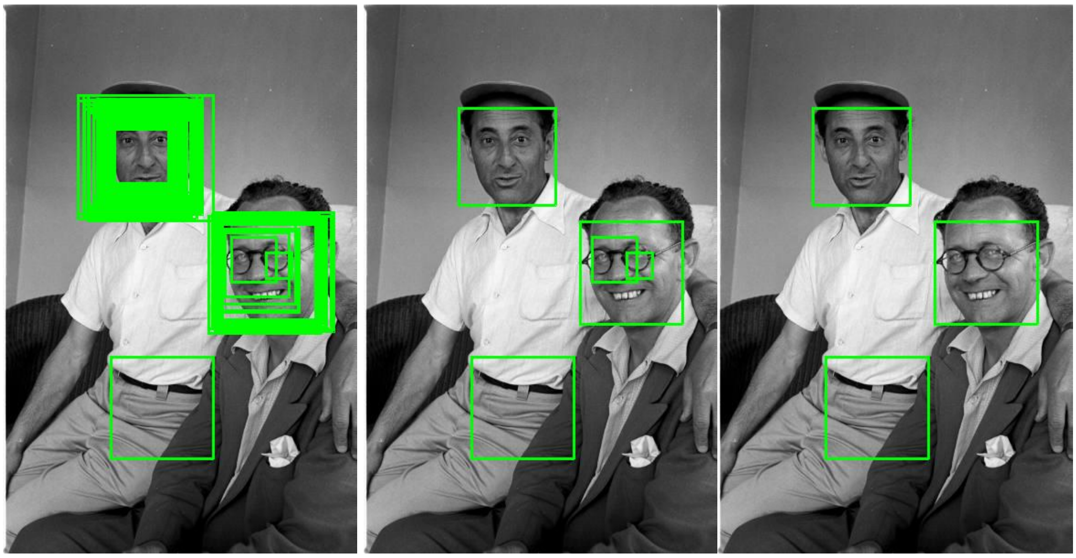


Figure 3.8. Left, raw detection results of the 200-features classifier with NFPmax = 0.5. Center, result of the proposed grouping and simplification method. Right, result of alternative grouping method.

2. Second, only stable detections are kept. Typically, the stability of a detection is measured in terms of the thickness of the group of detections it represents (i.e., the number of detections in the group). This criterion adds an additional parameter to the detector. For instance, in the OpenCV implementation of the Viola–Jones detector[6], by default, a minimum of three detections are required to keep the group. In [102] a similar

[6] OpenCV documentation is available at http://www.emgu.com/wiki/files/1.3.0.0/html/55a16889-537c- 534f-f2fa-fbbe60e1d8d4.htm

criterion is used, but the minimum required number of detections depends on the size of the detection windows. We get rid of this parameter by assessing the stability of the detection in a different way: we take advantage of the lack of left-right symmetry of many of the features of the 200-features strong detector and keep only the raw detections testing positive in a mirror version (horizontal flip) of the input image. This is illustrated in Figure 3.9. The raw detections surviving this test are then grouped together and simplified as in the previous post-processing method.

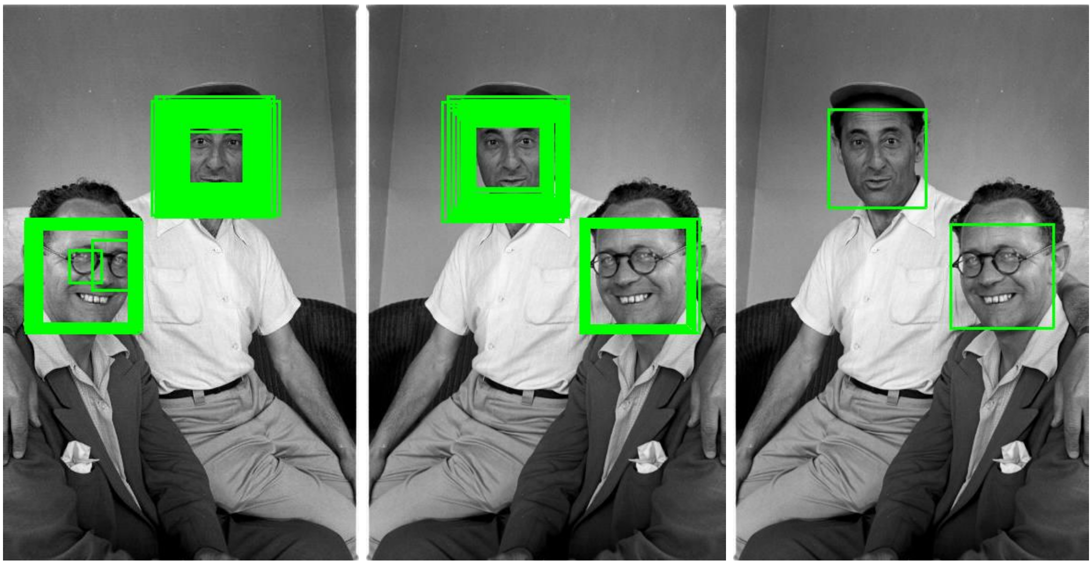


Figure 3.9. Left, raw detection results of the 200-features classifier with $\mathrm{NFP}_{\max} = 0.5$ applied to a mirror version (horizontal flip) of the same image used in Figure 3.8. Observe that the detection results are slightly different than the ones displayed in Figure 3.8-letf. Center, intersection of detection results on original and mirror image. Right, final result after detections grouping and simplification.

We shall call the result of the first type of post-processing method the rawS result and the result of the second type the mirrorS result. In the following Sections we will evaluate the performance of the face detector for each kind of result.

3.4.2 Detecting faces with a single classifier in "easy" datasets.

The IMM [72] and BioID [5] datasets are composed, respectively, of 240 and 1521 images, each one containing a single face. The faces correspond to 40 individuals in the first dataset and 23 in the second one, are placed in front of the camera, and display different facial expressions and small variations in position with respect to the frontal view. The IMM dataset contains both color and gray images, while BioID contains just gray images. Moreover, none of the persons in the first dataset wear glasses, while some of them do in the second. The illumination conditions also may change between different snapshots of the same person. Some examples of each dataset are shown in Figure 3.10.

Both datasets are manually annotated. BioID provides information on the position of the eyes, while IMM contains 58 landmarks per image: eyebrows, eyes, nose, mouth, and jaw. In this last case we have just used the eyes information, from which we have computed the center of each eye.

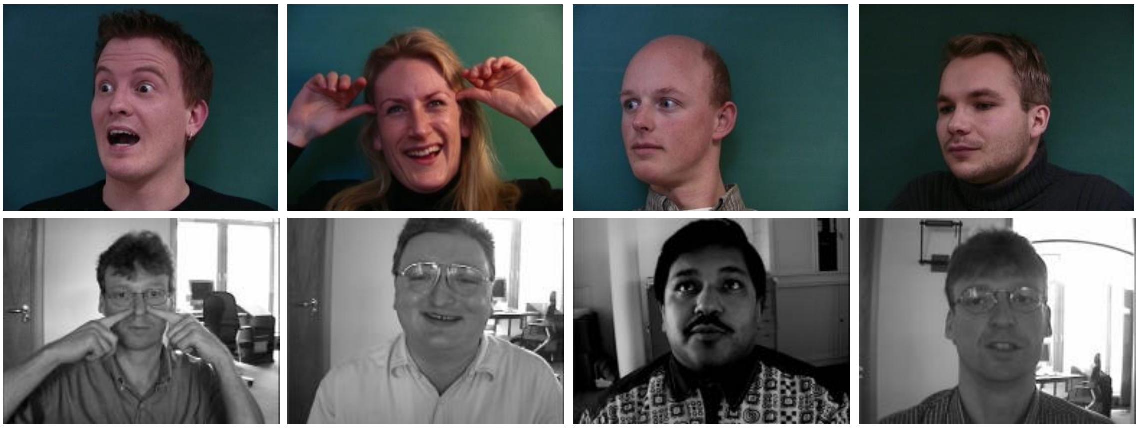


Figure 3.10. Top, examples of images in the IMM dataset. Bottom, examples of images in the BioID dataset.

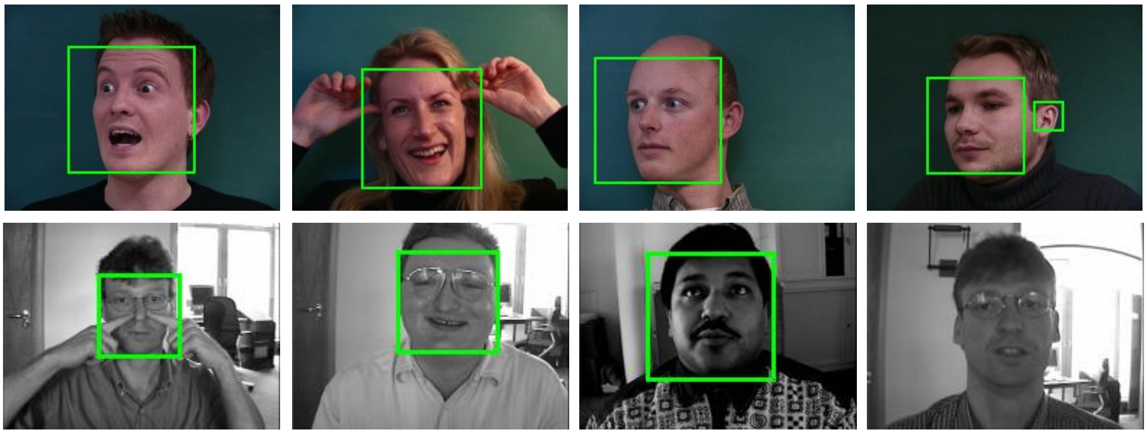


Figure 3.11. Top, examples of detection results for images in the IMM dataset Bottom, examples of detection results for images in the BioID dataset A single 200-features classifier was used for the detection, with NFPmax = 0.5. The last column displays a false positive for the IMM dataset and a missed detection for the BioID dataset. For these images the results of the rawS and mirrorS versions of the algorithm are the same.

We have run our face detector, consisting of a single strong classifier using 200 features, on both datasets, for different values of the NFPmax parameter. A detection is considered positive if (i) it contains both eyes, (ii) they are located above the center of the detection subwindow, and (iii) the size of the subwindow is less than five times the distance between the eyes. Using this detection criterion we compute, for each value of the parameter, the true positive rate[7] (R) and the number of false positives (FP) and we display the results in Table 3.3. The table shows the results of the two types of post-processing methods proposed in Section 3.4.1, namely, rawS and mirrorS. Note that the rawS results can be considered as the baseline results of the proposed method (i.e., results due to the adaptive selection of the detection threshold, not to the post-processing), since just a mere simplification of the raw detections is performed, while the mirrorS results include a further post-processing.

NFPmax	IMM				BioID			
	RawS		MirrorS		RawS		MirrorS	
	R	FP	R	FP	R	FP	R	FP
0.001	0.9833	0	0.9083	0	0.8008	3	0.5943	0
0.005	0.9958	0	0.9950	0	0.8843	7	0.7068	1
0.01	1	0	0.9625	0	0.9139	14	0.7567	1
0.05	1	1	0.9750	0	0.9704	75	0.8606	1
0.1	1	4	0.9875	0	0.9783	129	0.8955	1
0.5	1	10	1	2	0.9895	386	0.9632	13
1	1	15	1	3	0.9908	594	0.9757	32
2	1	18	1	4	0.9934	859	0.9862	68
4	1	24	1	6	0.9934	1149	0.9934	176
	R		FP		R		FP	
OpenCV [42]	1		7		0.9678		69	
31-stages [102]	1		2		0.9598		10	

Table 3.3. Results on IMM and BioID datasets of the 200-features detector with adaptive detection threshold and results for other implementations of the Viola–Jones detector.

For comparison, the results of two implementations of the Viola–Jones cascade are also shown in the table: the OpenCV implementation[8] and a recently published 31-stages implementation [102]. In both cases we used the default parameters (basically the minimum number of subwindows required to keep a group of detections, which is set to three in both implementations). It must be noted that the performance of the different implementations of the Viola–Jones cascade depends on multiple factors: datasets used for training, criteria in the cascade design (number of stages, detection rates at each stage), postprocessing of the results, etc. We just display the results of the implementations in [42] and [102] as a reference of what can be achieved with a classical cascade of classifiers.

These results show that for a similar number of false positives, our single-stage detector is able to perform as well as a long cascade of classifiers. In particular, for NFPmax = 0.05 the

[7] The true positive rate or Recall is defined as the ratio between the number of true positive detections (TP) and the number of faces in the dataset: $\mathrm{TPR} = \mathrm{R} = \frac{\mathrm{TP}}{\text{number of faces}}$.

[8] This implementation is documented in http://docs.opencv.org/2.4/modules/objdetect/doc/cascade classification.html. We used the cascade available from http://alereimondo.no-ip.org/OpenCV/34, frontal Face10.zip, haarcascade frontalface default.xml. The implemented cascade, consisting of 20 stages, is an improvement of the original Viola–Jones cascade due to Lienhart and Maydt [55].

rawS version of the algorithm performs slightly better than the OpenCV cascade on both datasets. Compared to the 31-stages cascade our detector gets similar results on IMM but slightly worse results on BioID. Moreover, we observe that the mirrorS version of the algorithm always produces a much smaller number of false positives than the rawS version, given the same NFPmax parameter value, at the expense of a lower true detection rate. However, as NFPmax increases the difference in detection rates of both versions gets smaller. In general, the mirrorS version of the algorithm produces better results than the rawS version. In particular, the results of mirrorS with NFPmax = 0.5 are comparable to those of the 31-stages cascade and better than the ones from the OpenCV implementation, for both datasets. Figure 3.11 displays the detection results, with NFPmax = 0.5, for the images in Figure 3.10. For these images the results of the rawS and mirrorS versions of the algorithm are the same.

3.4.3 Detecting faces in-the-wild with a single classifier.

In this Section we test the single 200-features classifier on the Face Detection Data Set and Benchmark (FDDB) [34]. This dataset is composed of 2845 images (gray and color) containing 5171 faces in arbitrary positions (including lateral views and partial occlusions), so-called in-the-wild. Some examples of the images in this dataset are shown in Figure 3.12.

Since our detector was trained using frontal faces it is out of the question to achieve state of the art performance on this dataset, but we include it so we can compare our results with the results reported for the OpenCV implementation of the Viola–Jones cascade, which are publicly available at the FDDB website. We will show that our single-stage detector slightly outperforms this cascade.

FDDB provides annotations for all the faces in the dataset. Each face is described by an ellipse. Moreover, there is a standard procedure to report results on the benchmark, which permits us to compare the performances of different face detection algorithms. The match degree between a detection window *d* and an annotated face *a* is computed as

$$S(d,a) = \frac{A(d \cap a)}{A(d \cup a)}, \qquad (3.12)$$

where A() is the area operator.

The FDDB evaluation protocol recommends providing two ROC curves that plot the false positives versus the true positive rate by using a discrete or continuous detection score. For the discrete case a detection is considered positive if S(d, a) > 0.5, while in the continuous setting every detection window has a weight S(d, a).

Figure 3.13 displays the discrete and continuous ROC curves corresponding to the rawS and mirrorS results of our 200 features detector. The curves reporting the results for the OpenCV Viola–Jones implementation and for the method proposed by Jain and Learned-Miller [35] are also shown for comparison. This latter method has been included in the comparisons because it also proposes to adapt the detection thresholds of the original Viola–Jones cascade to the image contents, but using an approach different from ours (see below for additional comments about this method). Finally, the results for the 31-stages cascade described in [102] are also displayed. In this case, since a detection score is not provided by the detector, just a single ROC point is displayed.

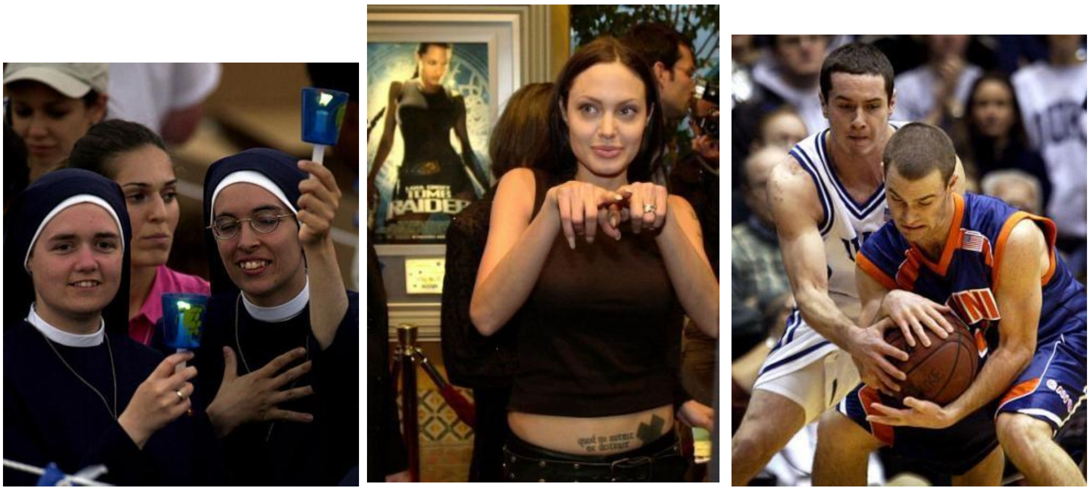
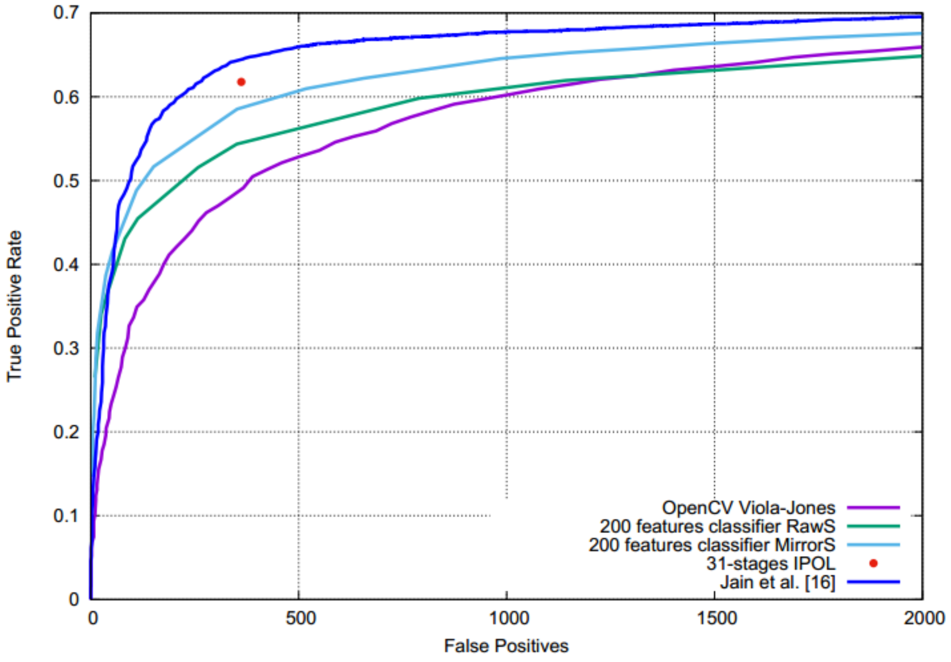


Figure 3.12. Some images in the FDDB dataset.

The rawS results show that a single classifier with 200 features and adaptive detection threshold is able to perform better than the OpenCV Viola–Jones cascade, i.e., it obtains higher detection rates for similar amounts of false positives. With respect to the implementation of the Viola–Jones detector in [102], we get slightly worse results, but this is to be expected since [102] includes a post-processing step that eliminates nonreliable detections while rawS does not. We observe that the use of the mirrorS postprocessing permits, approximately, a 3% increase in the detection rates, achieving performances closer to [102].

With respect to the method proposed in [35], our method gets worse results. The reason is that for the FDDB dataset, several faces are usually present in each image, some of them with high detection values. These are used in [35] to estimate new detection thresholds for similar faces with lower detection values (and which were initially rejected as faces). The consequence is an increase in the detection rates. However, note that this technique will not improve the detection results when only a single face is present in the image (e.g., in the IMM and BioID datasets) or when there is little similarity in appearance among different faces in the same image. Since the threshold adaptation technique proposed by Jain and Learned-Miller is independent of the face detector, we could, in principle, use the same technique to improve our detection results. This is a possibility that shall be explored in our future research.

Figure 3.14 displays some detection results of the 200-features classifier on the FDDB dataset. Additional examples will be shown in the next Section.



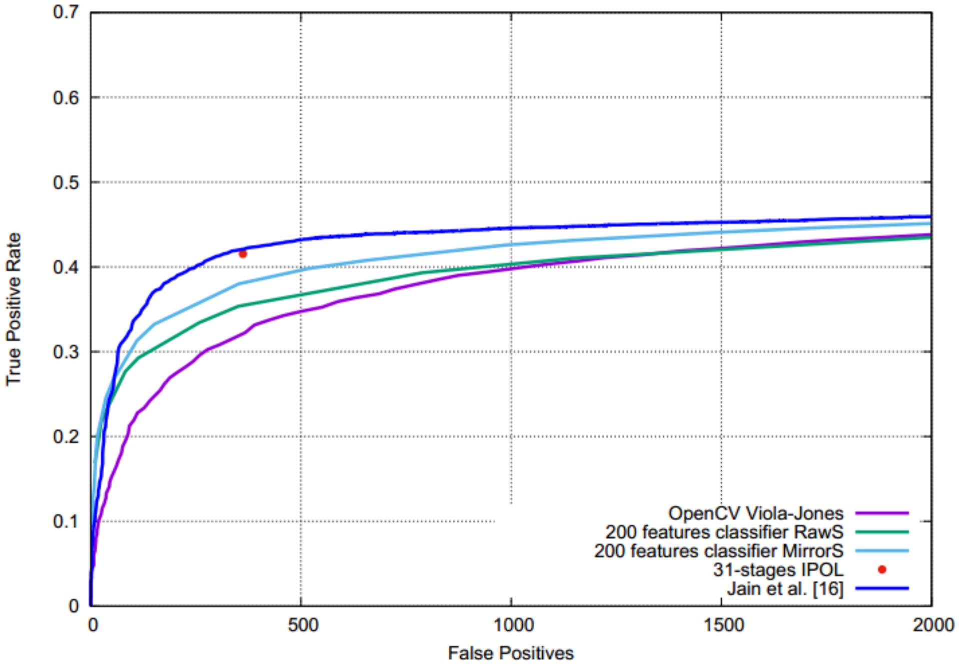


Figure 3.13. FDDB results of the 200-features classifier with adaptive detection threshold. Top, ROC curve using the discrete degree of match. Bottom, ROC curve using the continuous degree of match. In both cases a maximum of 2000 false positives are shown.

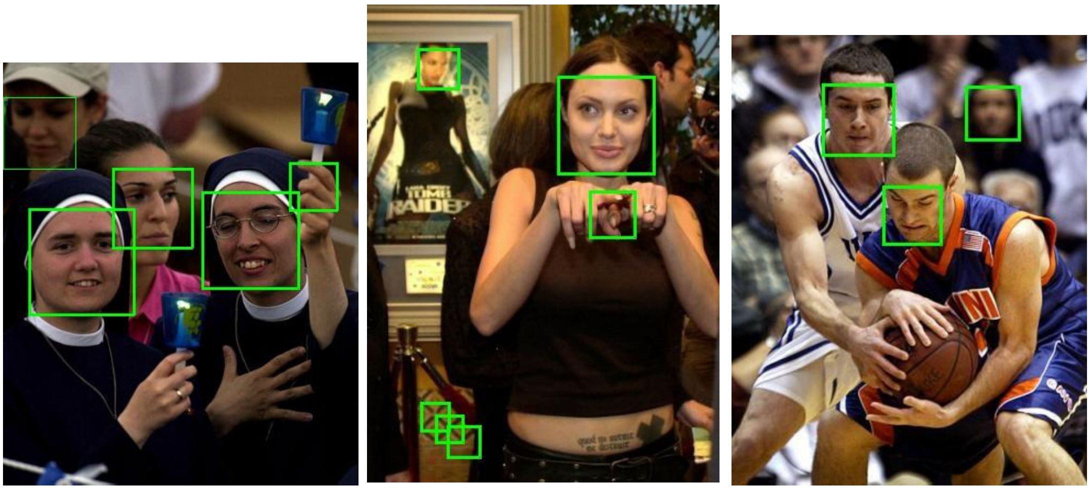


Figure 3.14. Some results of the 200-features classifier on the FDDB dataset. The results correspond to the rawS type of post-processing. The results of the mirrorS post-processing are the same as the ones displayed in Figures 3.16 and 3.17.

3.5 A short cascade of classifiers with adaptive detection thresholds.

The results displayed so far prove that by adapting the detection threshold to the image content, using the a contrario approach, it is possible to dramatically improve the performance of a single strong classifier, achieving detection performances similar to the ones of a full cascade.

Nevertheless, the performance of a face detector can be simply measured not only in terms of its detection rate but also in terms of its computational efficiency. It is clear that a single classifier with 200 features must check all these features on all possible image subwindows in order to produce its result. On the other hand, the use of a cascade of classifiers permits us to reject most of the false positives in the early stages, which are composed of a small number of features. Therefore, even if the total number of features in the full cascade is big (up to 6000 features in the 38-stages original Viola–Jones cascade), the average number of features tested per subwindow is relatively small.

In this Section we will see how to combine the cascade-of-classifiers idea and the adaptive-single-classifier method to produce a very short cascade (just four classifiers) whose performance, in terms of detection rates, shall be comparable to that of a long cascade. Moreover, since its total number of features will be much smaller, it will be computationally more efficient and much faster to train.

3.5.1 The proposed short cascade.

We propose a cascade of four classifiers with 5, 10, 80, and 200 features, respectively. As with any cascade the goal is to reject most of the subwindows in the initial stages (in our case the first three stages) and then apply the 200-features classifier to a fraction of the subwindows. We seek to preserve the detection rates of the single 200-features classifier presented in the previous Sections at a much lower computational cost. We have trained the cascade as proposed in the original Viola–Jones paper [99], using the same set of frontal 24x24 faces and, as negative examples for each stage, the false positives from the previous stage. For detection, ideally we would like to apply the threshold adaptation principle (Section 3.3) to all four classifiers. However, as commented in Section 3.2, only when the number of features is large enough does the Gaussian model for the distribution of detection values apply. Therefore, for the first two stages of the cascade (5 and 10 features, respectively) we just set the detection thresholds to allow a fixed percentage of subwindows through the classifier. For the 5-features classifier we reject all the subwindows whose detection value is below the 80th percentile (only the top 20% of subwindows are let through), while for the 10-features classifier all the subwindows below the 95th percentile are rejected (only the top 5% of subwindows are let through). For the 80-features classifier we set the threshold as in Section 3.3 and we use a fixed value of $\mathrm{NFP}_{max}^{80\text{ feat}} = 100$. These values of the parameters are quite permissive, the goal being to preserve as many as possible of the true positives, which should be correctly classified by the last stage of the cascade. We have tested different sets of values (e.g., 60%, 90%, and 200) but we have found little difference in the final results, so we have fixed the set (80%, 95%, and 100) for all our experiments. The only tunable parameter of the cascade is therefore the maximum number of false positives for the 200-features classifier ($\mathrm{NFP}_{max}^{200\text{ feat}}$) in the last stage. In sub-Section 3.5.2 we further elaborate about the design criteria used to construct the cascade.

3.5.1.1 Sampling the set of subwindows.

In order to characterize the Gaussian functions used to model the distributions of detection values described in Section 3.2 we need to compute their mean and variance. We have assumed so far that these distributions are computed using the entire set of possible image subwindows[9] for each of the classifiers. However, this would imply to check 295 (= 5 + 10 + 80 + 200) features on each subwindow. In order to reduce the number of computations we check all the subwindows just for the 5-features classifier and then subsample the set of subwindows to get the rest of the distributions. If N denotes the total number of subwindows, we use pN of them to obtain the detection histograms for the 10-, 80-, and 200-features classifiers, with p a fixed parameter which we have set to p = 0.01 = 1%.

This choice of the p parameter is justified by basic arguments of inferential statistics [2]. This theory states that the sample mean and variance of a set of values taken from a Gaussian population are random variables that concentrate around the population parameter and whose variance is inversely proportional to the sample size and directly proportional to the population variance. In our case (see Section 3.2, Figures 3.2 and 3.4), N is typically of the order of millions, the population variance is always smaller than 5, and the population mean is always above 1; therefore by using 0.01N samples we are confident that the estimated values will not be far from the real ones. We have also tested with p = 0.1, finding very little difference in the final results.

For the set of fixed parameters described above it is possible to estimate the computational efficiency of the proposed cascade. Let N be the total number of image subwindows; then the

[9] As already noted, in our experiments we consider all nonflat subwindows of sizes ranging from 20 × 20 to 220 × 220 pixels.

average number of checked features per subwindow is computed as

$$Avg = \frac{T_f}{N}\,, \qquad (3.13)$$

where Tf denotes the total number of checked features:

Tf =	$5N +$	All the subwindows go through the 5 features classifier.
	$(10 + 80 + 200)\cdot 0.01N +$	A subset of 0.01N subwindows is used to estimate the distribution of values for the 10, 80, and 200 features classifiers.
	$10\cdot 0.2N +$	20% of the subwindows go through the 10-features classifier.
	$80\cdot 0.05N +$	5% of the subwindows go through the 80-features classifier (this is an upper bound, since some subwindows may be rejected by the previous classifier).
	$200\cdot p'N$	An unknown (but very small) percentage p' of the subwindows goes through the 200-features classifier.

By replacing this expression in (3.13), and taking into account that the unknown value p' is very small, we get an upper bound for the average number of checked features: Avg ≈ 13.9. Table 3.7 in Section 3.6 compares, for different datasets, the actual value of Avg for our short cascade and the values obtained for the 31-stages cascade described in [102].

3.5.2 On the design criteria of the short cascade.

One might argue against the arbitrary decision of using four classifiers in the short cascade presented in this Section. Moreover, the thresholds used at each stage seem also arbitrary. In this paragraph we try to justify these choices. The following criteria have guided the design of the cascade:

1. We want to build a cascade whose last stage is the 200-features classifier presented in the previous Sections. The reason is that this classifier exhibits good detection performance and it uses an adaptive detection threshold with statistical meaning.
2. In order to preserve the detection rates of the 200-features classifier, no true positives should be rejected by the previous stages of the cascade. This implies the use of quite permissive detection thresholds in these stages.
3. In order to accelerate computations the initial classifier(s) of the cascade must be composed of a small number of features.

Concerning the detection thresholds used by the classifiers, note that only classifiers with a large enough number of features display a distribution of detection values regular enough to admit a statistical analysis (see Figure 3.2). For this reason, when using classifiers with fewer that 80 features we are led to use fixed, not adaptive, thresholds. We found that, in general, no true positives were rejected (requirement 2) by a 5-features classifier when admitting through the classifier at least 20% of the subwindows. In the case of a 10-features classifier the minimum value of the threshold was 5%. By using other percentages (we tested 40% and 10%, respectively) the results were very similar but the computational cost (requirement 3) higher. Similarly, for an 80-features classifier, all the true positives are in general preserved when allowing at least 100 false positives at the output.

Several different short cascades could have been designed using these criteria. For example, a 2-steps cascade (e.g., 5 + 200 or 10 + 200 features) could have been used. But in both cases the computational efficiency of the cascade (computed in terms of the average number Avg of checked features per subwindow; see the previous Section) would be low. Indeed, by using the 20% and 5% thresholds proposed above and by estimating Avg as in the previous Section we would get Avg_{5+200} = 47 and Avg_{10+200} = 22. The use of more features in the first stage would imply that more features should be tested on all the image subwindows, thus increasing the computational cost (e.g., for a 20 + 200 cascade $\text{Avg}_{20+200} > 20$). The use of a 3-steps cascade, where the two initial stages are composed of a small number of features, permits us to reduce the overall computational cost (e.g., $\text{Avg}_{5+10+200}$ = 19.1). Another possibility is to use a 3-steps cascade with a very discriminative next-to-last classifier, which reduces considerably the number of subwindows reaching the 200-features classifier (e.g., $\text{Avg}_{5+80+200} \approx 23.8$, $\text{Avg}_{10+80+200} \approx 16.8$). We opted for a 4-steps cascade which combines the advantages of the two types of 3-steps cascades commented above: two initial stages with a small number of features, followed by a very discriminative classifier. Our final proposal is thus a 5 + 10 + 80 + 200 cascade, for which $\text{Avg}_{5+10+80+200} \approx 13.9$. Other choices for the 4-steps cascade were pondered. For example, with a 5 + 10 + 20 + 200 cascade we would get $\text{Avg}_{5+10+20+200}$ = 12.3 (assuming a fixed detection threshold of 1% for the 20-classifiers stage), but at the cost of adding a new fixed threshold to the method. We preferred to use the 80-features classifier for which an adaptive threshold could be estimated using the methodology described in Section 3.3. Of course other choices could have been made (e.g., 4+15+100+200, or the use of more stages in the cascade), but the final results, in terms of detection performance (which ultimately depends on the 200-features classifier of the last stage) and computation time (which mainly depends on the initial stages of the cascade), would not be very different from the ones obtained with the proposed implementation.

3.6 Experiments with the short cascade.

We have tested our short cascade with the same datasets used in Sections 3.4.2 and 3.4.3. Table 3.4 displays the results obtained on the IMM and BioID datasets. When we compare them to the ones in Table 3.3 we observe that the use of a cascade reduces the amount of false positives for the same value of the parameter NFPmax, while the detection rates are only slightly reduced, thus meeting our design goals. Moreover, we see that the results of mirrorS with NFP_{max} = 0.5 are comparable to those of the 31-stages cascade and better than the ones from the OpenCV implementation, for both datasets.

Figure 3.15 displays the discrete and continuous ROC curves corresponding to the rawS and mirrorS results of the cascade on the FDDB dataset. For comparison, we also include the results for the single 200-features classifier (mirrorS result, already reported in Figure 3.13), the ROC curves for the OpenCV Viola–Jones implementation and the adaptive method by Jain and Learned-Miller [35], and the results for the 31-stages cascade described in [102]. Moreover, Table 3.5 shows some of the values in the discrete ROC curve and the value of the NFPmax parameter for which they were obtained. The value corresponding to the 31-stages cascade is also displayed in the table.

	IMM				**BioID**			
	RawS		**MirrorS**		**RawS**		**MirrorS**	
NFPmax	**R**	**FP**	**R**	**FP**	**R**	**FP**	**R**	**FP**
0.001	0.9875	0	0.9042	0	0.7837	3	0.5897	0
0.005	1	0	0.9458	0	0.8659	6	0.7068	1
0.01	1	0	0.9542	0	0.8876	16	0.7423	1
0.05	1	0	0.9792	0	0.9520	70	0.8448	1
0.1	1	2	0.9833	0	0.9592	102	0.8830	3
0.5	1	3	1	1	0.9783	295	0.9448	11
1	1	3	1	1	0.9836	424	0.9606	23
2	1	5	1	2	0.9869	602	0.9724	61
4	1	8	1	3	0.9882	824	0.9796	149
	R		**FP**		**R**		**FP**	
OpenCV [42]	1		7		0.9678		69	
31-stages [102]	1		2		0.9598		10	

Table 3.4. Results on IMM and BioID datasets of the short cascade of classifiers with adaptive detection threshold, and results for other implementations of the Viola–Jones detector.

These results show that the performance of the proposed cascade is similar to the one of the 200-features classifier (at a much smaller computational cost) and that when using the mirrorS postprocessing we can even improve its results. When comparing the short cascade to different implementations of the Viola–Jones face detector we reach the same conclusions, for the FDDB dataset, as those achieved in Section 3.4.3: we obtain higher detection rates for similar amounts of false positives compared to the OpenCV Viola–Jones results reported in the FDDB website and in [111] but worse than Jain and Learned-Miller [35]; we get slightly worse detection results but with a lower number of false positives than the implementation in [102]. It is worth noting (see Tables 3.4 and 3.5) that by fixing the single parameter of our method to NFPmax to 0.5 and using the mirrorS postprocessing we get similar results in all datasets (IMM, BioID, and FDDB) to the 31-stages cascade described in [102].

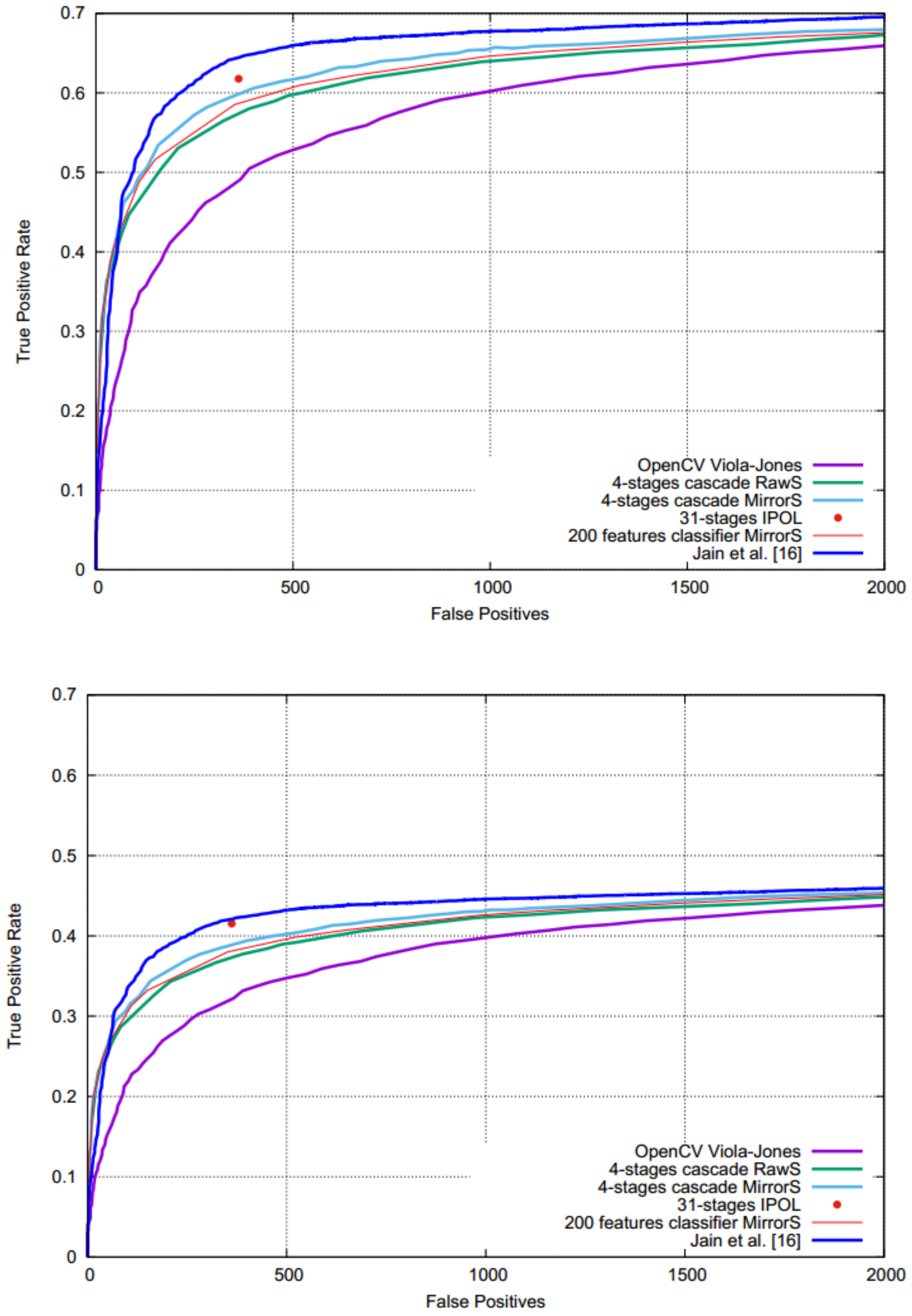


Figure 3.15. FDDB results of the proposed short cascade. Top, ROC curve using the discrete degree of match. Bottom, ROC curve using the continuous degree of match. In both cases a maximum of 2000 false positives are shown.

Figures 3.16 and 3.17 display some detection results of our short cascade for NFPmax = 0.5. The images display some examples of missed detections (mainly nonfrontal or occluded faces) and false positives. The images in the second row of Figure 3.17 show some positive detections that were classified as false positives by the standard evaluation procedure used to assess the performance of face detectors in the FDDB dataset. In some cases, the reason is that the detection areas were too small with respect to the annotated ellipses and the detection score was below the required 0.5 threshold. In other cases (leftmost image in the bottom row of Figure 3.17) it is due to an annotation error. This means that the actual detection rate of the proposed cascade is indeed higher than the one reported in Figure 3.15.

It is interesting to remark that the difference in detection performance between our short cascade and the classical Viola–Jones detector is especially noticeable when considering small amounts of false positives. This can be seen in Table 3.6, which compares the results (discrete version) of our cascade to the ones for the OpenCV version of the Viola–Jones detector, as reported in [111], that correspond to approximatively no false positives (i.e., false positives around 10), false positives around 100, and false positives around 1000.

	FDDB			
	RawS		**MirrorS**	
NFP$_{max}$	**R**	**FP**	**R**	**FP**
0.001	0.45	83	0.31	15
0.005	0.50	165	0.38	36
0.01	0.53	208	0.41	49
0.05	0.58	389	0.48	95
0.1	0.60	621	0.51	130
0.5	0.64	975	0.57	249
1	0.65	1285	0.59	332
2	0.66	1670	0.61	468
4	0.68	2222	0.63	658
	R		**FP**	
31-stages	0.62		362	

Table 3.5. Some values of the discrete ROC curve displayed in Figure 3.15 for the 4-stages cascade, and the corresponding value of the NFPmax parameter. The ROC value for the 31-stages cascade is also shown.

To end this Section we show in Table 3.7 the average number of features checked by our cascade on different datasets[10]. We display also the corresponding values for the 31-stages face detector [102]. Note that our method reduces the computations by a factor of ≈ 5.

[10] The displayed values are the average, over all the images in each dataset, of the average number of features checked on each image.

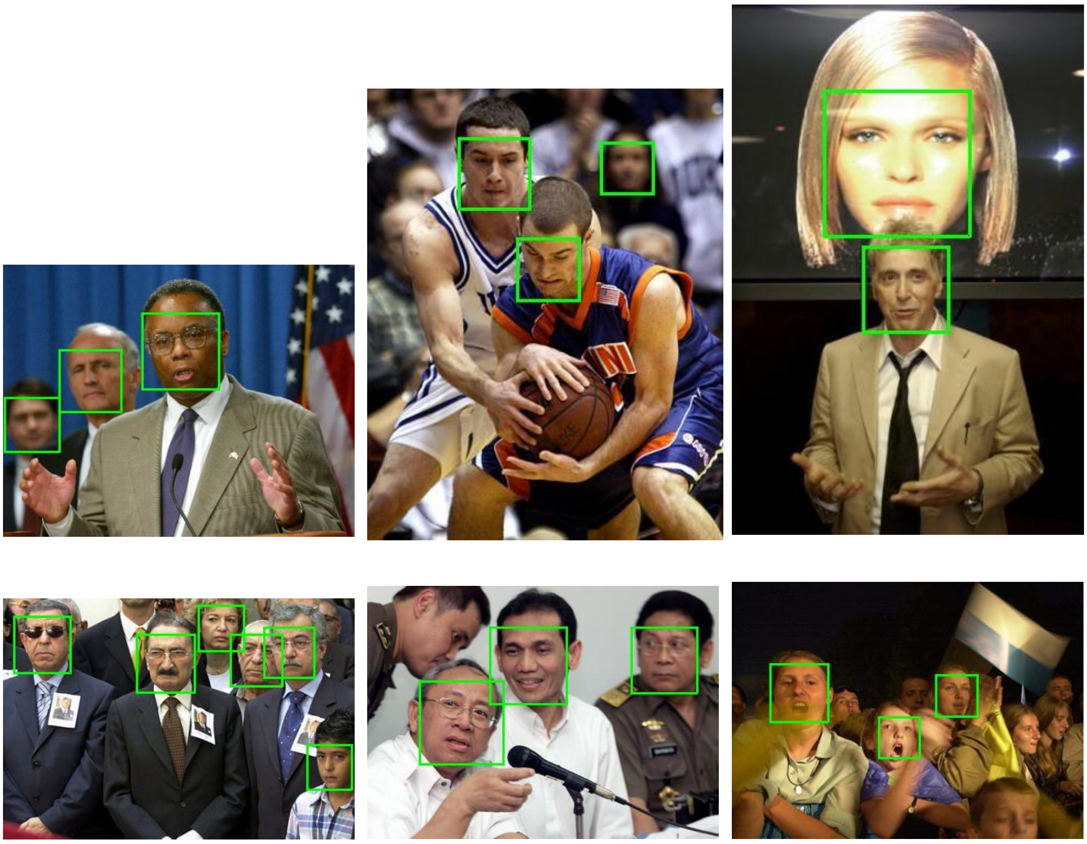


Figure 3.16. Some detection results of our short cascade for NFPmax = 0.5 (mirrorS result).

	False Positives		
Method	**≈10**	**≈100**	**≈100**
4-stages cascade (mirrorS version)	28%	48%	65.5%
Viola-Jones OpenCV version	10%	33%	59.7%

Table 3.6. Comparison of detection results (true positive rate, discrete version) of our cascade and the OpenCV version of the Viola–Jones cascade for different amounts of false positives.

Method	**IMM**	**BioID**	**FDDB**
4-stages	11.44	12.20	11.69
31-stages	60.61	60.64	54.76

Table 3.7. Comparison of the average number of features checked by the 4-stages cascade and a 31-stages cascade on different datasets.

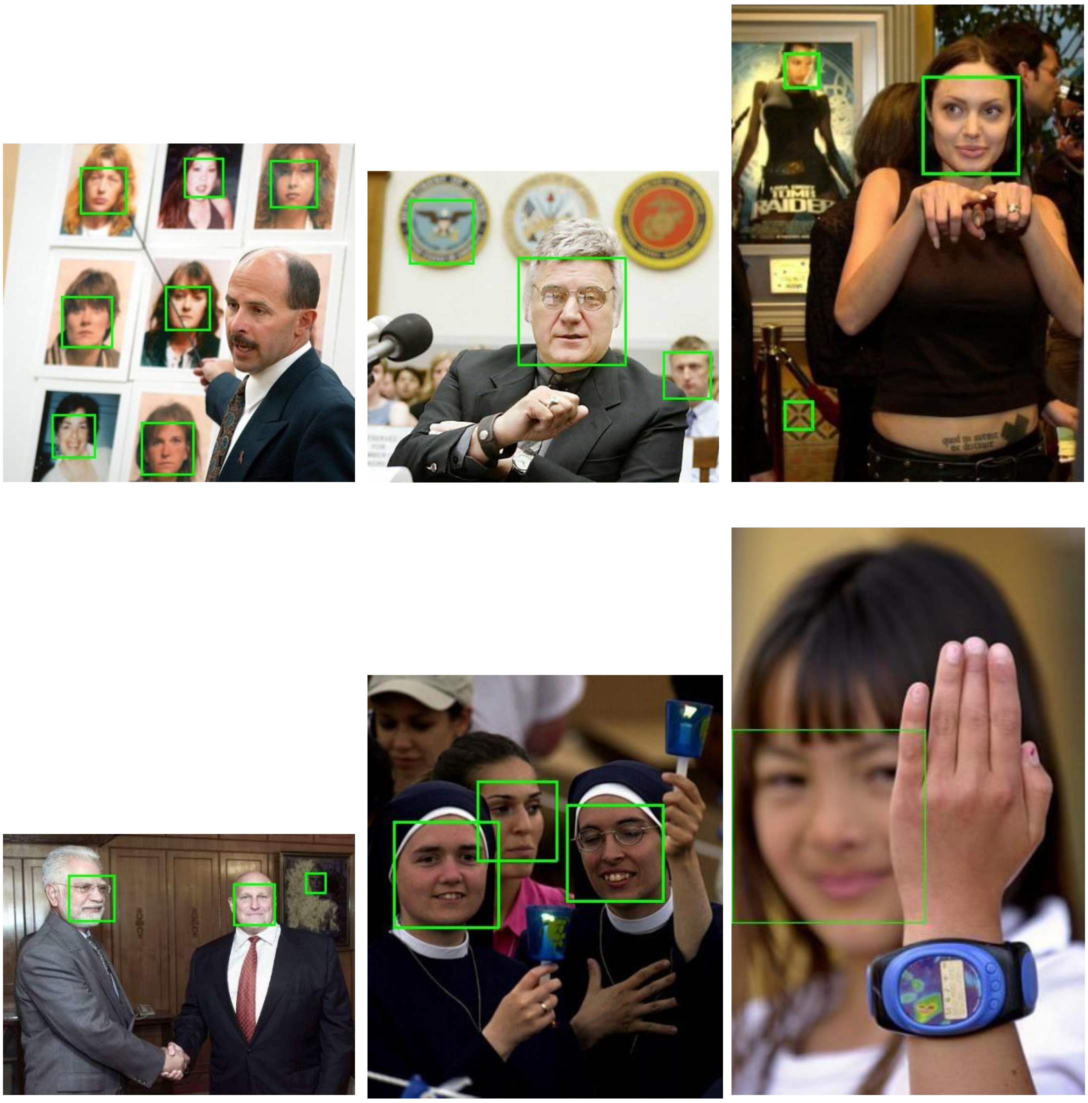


Figure 3.17. More detection results of our short cascade for NFP_{max} = 0.5 (mirrorS result) showing some missing detections and false positives. The images in the last row display some positive detections that were classified as false positives by the standard performance evaluator.

3.7 Conclusions

We have shown in this Chapter that it is possible to successfully use the *a contrario* methodology to improve the performance of the classical Viola–Jones face detector. We have justified that a Gaussian distribution can be used as a background model against which to test the existence of faces in an image, and then we have proposed a method to adapt the detection threshold of a single strong classifier to control the number of false positives. The method has been tested with three different representative datasets (IMM, BioID, FDDB). We have then applied the same principles to build a very short (and hence computationally very efficient and fast to train) cascade of just four stages which is able to compete, in terms of detection performance and computational complexity, with much larger cascades. Our method reduces the computation time by a factor near to five. The obtained results are promising and suggest that the same principles might be applied to more recent face detectors, for which we could achieve state of the art performance. This shall be the subject of our future research. In particular, we will explore the use of integral channel features trained using faces in various poses/views and the application of the threshold adaptation technique of Jain and Leamed- Miller [35] to improve the detection rates.

Chapter 4

Facial Expression Recognition

In Chapter 3, we designed a method which improves the accuracy of a classical cascade, at a much lower computational cost. In this Chapter, we used this method to detect the face in an image and analize its facial expression. In this Chapter we studied the impact of the preprocessing step, we performed an extensive experimental study in cross-dataset facial expression recognition, and we performed a study between the network's classification and that of humans.

Section 4.1 introduces the problem and the most relevant previous works. In Section 4.2, we explain all the datasets used for facial expression recognition, including our proposed datasets (FEGA and FE-Test). In Section 4.3, we present the image preprocessing and data augmentation steps in detail. Other important contribution is the CNN proposed for facial expression recognition, which is described in Section 4.4. The performed experiments and the analysis of the obtained results are presented in Section 4.5. Finally, the last Section of this Chapter shows the conclusion and main contributions.

4.1 Introduction

Facial expression recognition has gained increasing interest in the last years, due to the constantly increasing demand of applications for automatic human behavior analysis and novel technologies for human-machine communication and multimedia retrieval [98]. Although this field has been actively studied recently, few works have combined several datasets to perform a cross-dataset evaluation [67, 112]. This is because it is difficult to standardize all images from different datasets. The same expression among different people can vary according to ethnicity, age or gender. Charles Darwin argued that human emotions were both innate and universal in all cultures, in his book "The Expression of Emotions in Man and Animals" [33], but the researcher and emotion expert Paul Ekman found that, in many cases, the facial expressions tend to be shaped by the culture of origin [20]. Another feature that can affect is the age, which plays an important role in the representation of emotions. For example, elderly people tend to appear sad or angry when they are in their neutral expression due to natural dilation of the facial muscles with age. Even the gender can affect, since women generally use to be more expressive than men. In addition to the above mentioned factors, an expression of the same person can appear differently depending on brightness, background and posture. On the other hand, the image quality, color intensity, resolution, etc. are specifications that depend on the capture process and environment. These can affect the classification accuracy, especially in cross-dataset evaluation, since each dataset uses a different capture protocol. This problem can be observed in many papers in the literature [60, 67], where the classification results may decrease up to a 49% when applying cross-dataset evaluation.

Due to all this complexity, the field of automatic facial expression recognition presents significant challenges. In the majority of the published literature, the problem is simplified by focusing on achieving good results using the same method or combined-methods on a unique dataset or on several datasets separately, but with the training and testing sets belonging to the same dataset [36, 60, 37, 82, 103]. In spite of achieving a significant progress in facial expression recognition, the majority of the above mentioned papers are focused on getting a method to optimize the results in one or several datasets separately, but they do not solve the problem of cross-dataset evaluation.

Recent studies about combination of several datasets to perform a cross-dataset evaluation [67, 112] open a new approach by considering the goal of achieving good accuracy results in datasets different from the ones used for training, so we can apply them in real life applications. In this Section, we propose a fine-tuned convolutional neuronal network for facial expression recognition and a specific image preprocessing method which is applicable to any facial expressions dataset. The preprocessing step permits the combination of images from different datasets into a single dataset. Our method has been evaluated with four datasets widely employed in the literature (BUFDE, CK+, JAFFE, WSEFEP) and a new one (FEGA), using both single and cross datasets protocols. Eventually, these datasets have also been combined for training purposes in order to obtain a more robust system under cross-dataset evaluation. Cross-dataset evaluation is important since in the real world, the technologies that use facial expression recognition should be able to recognize emotions in any image and not just work well with a specific dataset. The new dataset (FEGA) was created in order to train the proposed CNN. Moreover, a new test set (called FE-Test) was also created to validate our system. This test set contains images with different illuminations, backgrounds and image resolution, which will permit to assess the robustness of our system. Besides information on facial expression in both datasets, the new dataset FEGA contains gender and age information. The combination of these three traits, which are closely related [71], would allow the development of better facial expression recognition methods, which shall be the subject of our future work.

Finally, we want to know how trustworthy our results are by comparing the performance of humans and machines in recognizing facial expressions. It may happen that an emotion can be ambiguous both for the human and machine. Therefore, in sub-Section 4.5.5, we carry out experiments using both deep learning techniques and human assessment of 253 participants to recognize the facial expressions on the FE-Test dataset.

4.2 Datasets

In this Section, we present a new dataset labeled, for the first time to our knowledge, with Facial Expression, Gender and Age simultaneously (FEGA). We use it in our experiments together with other four standard datasets widely used in facial expression studies: the Extended Cohn-Kanade (CK+) Dataset [62], the BU-4DFE Dataset [110], the JAFFE Dataset [64] and the WSEFEP Dataset [75]. We also present a new dataset (FE-Test) obtained from Internet, which is used to test facial expression methods with images captured "in the wild".

4.2.1 Standard Datasets in facial expression studies

Four popular standard datasets are used in this work (see Figure 4.1). The Extended Cohn-Kanade (CK+) Dataset [62], which contains 593 sequences from 123 subjects ranging from 18 to 30 years old. These sequences were labeled based on the subject’s impression of each of the 7 basic emotion categories: anger, contempt[11], disgust, fear, happy, sad, and surprise.

The BU-4DFE Dataset [110], which contains 606 3D facial expression sequences captured from 101 subjects, 58 females and 43 males. For each subject, there are six model sequences showing six prototypic facial expressions (anger, disgust, happiness, fear, sad and surprise), respectively.

The Japanese Dataset, JAFFE [64], that contains 213 images of 7 facial expressions (6 basic facial expressions + 1 neutral) posed by 10 female actresses. Each image has been rated with 6 emotion adjectives by 60 Japanese subjects.

And the WSEFEP Dataset [75], which contains 210 high-quality pictures of 30 individuals (14 men and 16 women) with each basic emotion: happiness, surprise, fear, sadness, anger, disgust and neutral. The pictures were carefully selected to fit criteria of basic emotions and then evaluated by independent judges.

4.2.2 New datasets (FEGA and FE-TEST).

We have created a new dataset (*FEGA)* with 51 subjects, 21 females and 30 males, between 21 and 66 years old. For each subject, there are six basic emotions [20] (anger, disgust, fear, happy, sadness and surprise) and neutral face. For each expression and subject, we captured eight RGB images with a resolution of 640x480 pixels. The subjects were asked to interpret the seven basic facial expressions, repeating each expression eight times, and one snapshot was taken each time. These images are similar, but not identical, because they were captured at different times. Also, for each subject, we labeled his/her images with his/her age and gender.

[11] In this paper we do not use this emotion because it is not one of the six basic emotions.

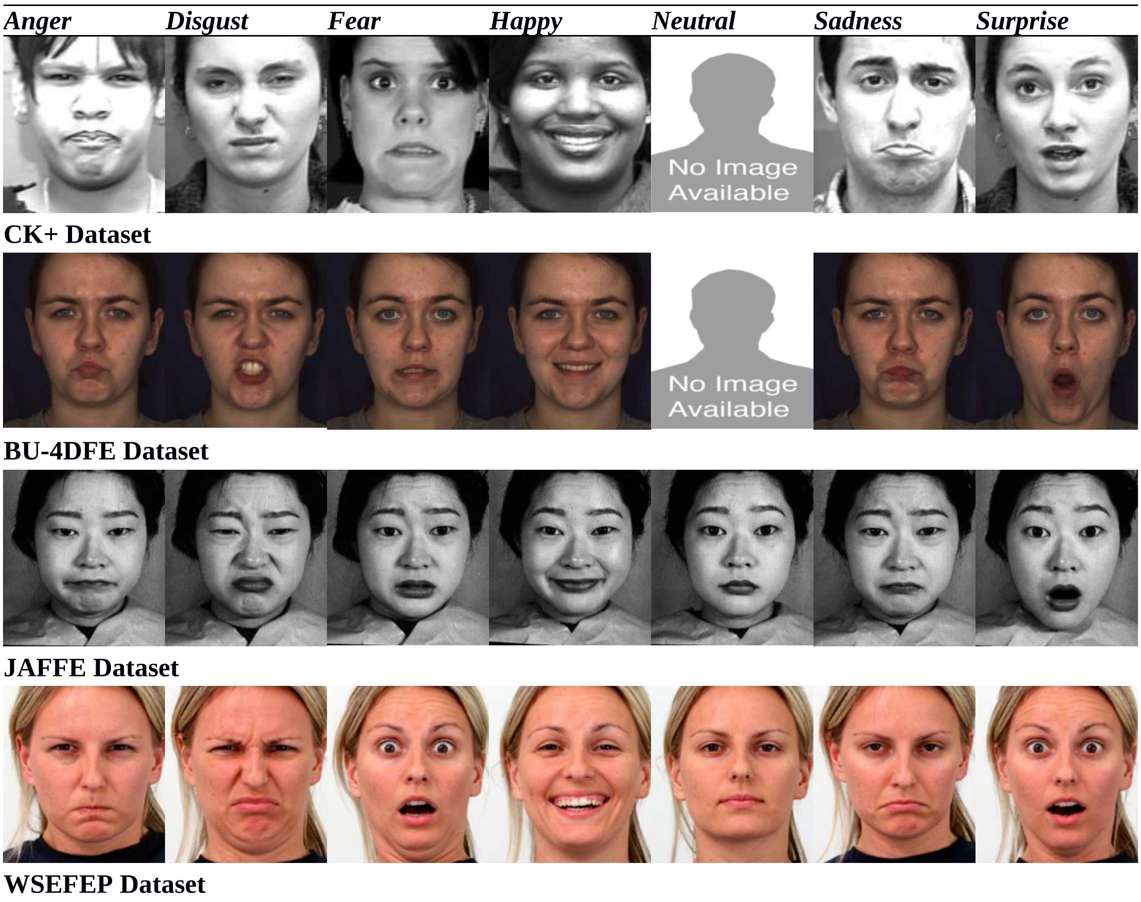


Figure 4.1. Some images of the four popular standard datasets in facial expression. Each column corresponds to one of the seven expressions mentioned above. Each file corresponds to one dataset.

Once we had the dataset, we analyzed the facial expression, image by image, to remove the outlier images which do not conform to the required quality for clear perception of expression. Therefore we present a dataset with 1668 images labeled, for the first time, with facial expression, gender and age simultaneously (see Figure 4.2). We also built a second dataset (FE-Test) which contains 210 frontal images of facial expressions labeled by Google and revised by research team (see Figure 4.3). We chose randomly 30 images from Internet for each expression (anger, disgust, fear, happy, sadness, surprise and neutral) with different illuminations, backgrounds and image resolution, in addition to faces with different ages and ethnicities. This dataset has been employed to test our algorithms with “realistic” images obtained from the Internet.

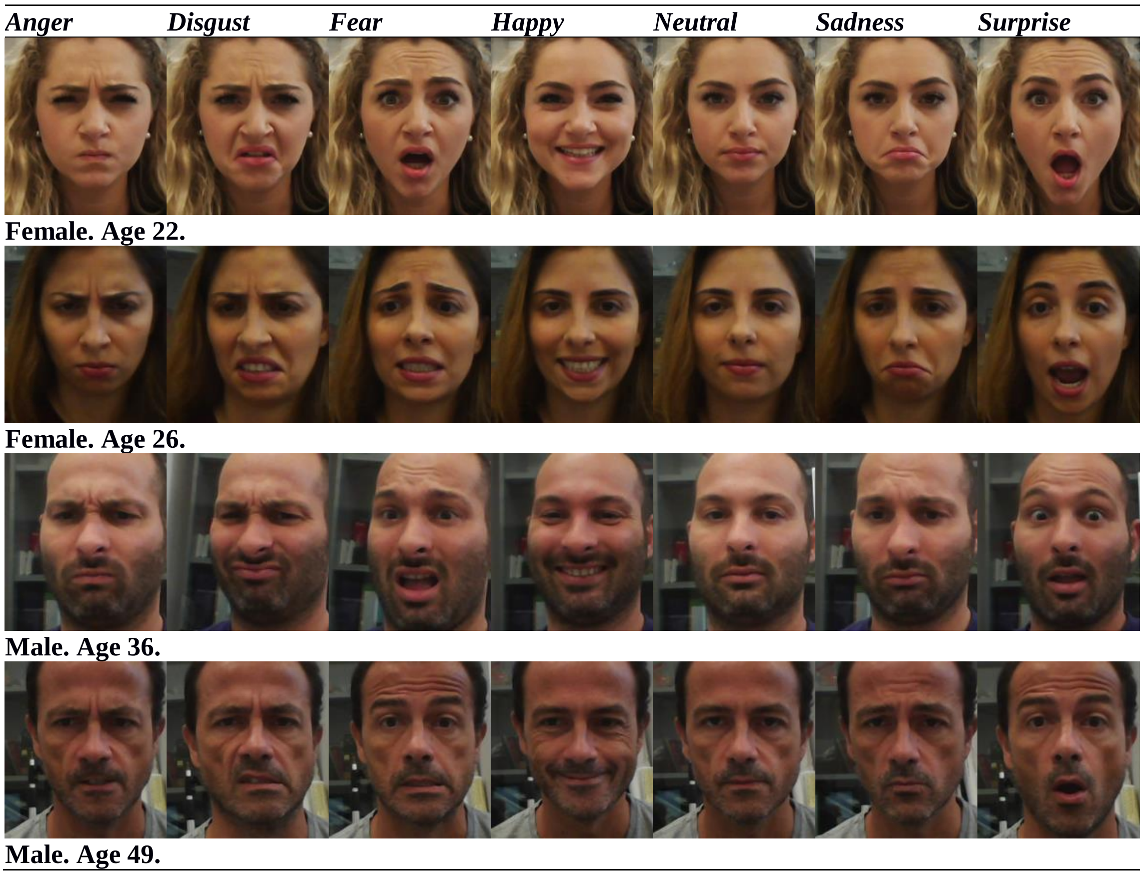


Figure 4.2. A small example of the FEGA Dataset. Each column corresponds to one of the seven expressions mentioned above. Each file corresponds to a gender and age.

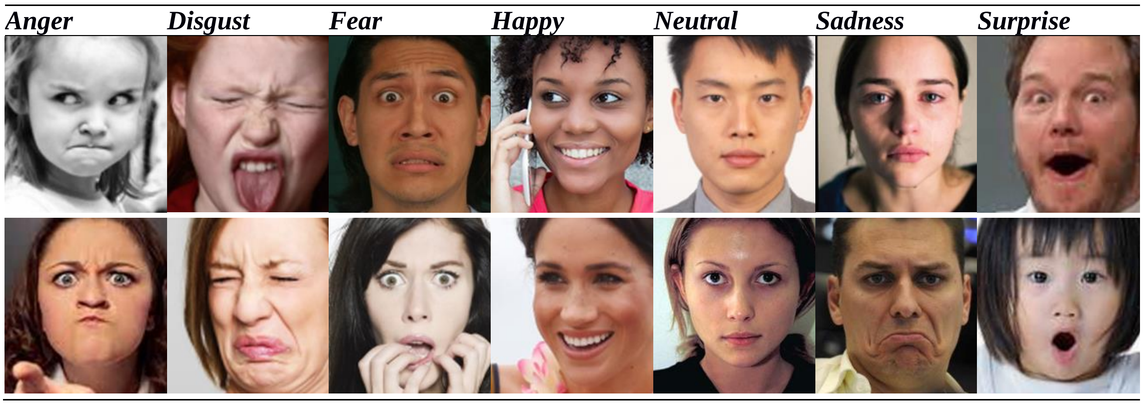


Figure 4.3. Some images of FE-Test dataset. Each column corresponds to one of the seven expressions.

4.3 Image Pre-Processing and Data Augmentation

When CNNs are adopted for any task, one of the most neglected steps is the pre-processing one. In fact, the general claim is that a deep model can manage whichever data variations by the huge number of parameters (weights etc.). The basic assumption is the large availability of data; in our case, facial images labeled with the related expression according to the Paul Ekman's model. However, we show how important the pre-processing step is.

In this Section, we show that a progressive refinement of the pre-processing step can significantly help in the final network's performance. First of all, we detect the face using the method proposed in Chapter 3. Then, we align the images to eliminate the possible rotations and get uniformity between images. We get the eyes position using 68 facial landmarks proposed by [81], which develop the first standardized benchmark for facial landmark localization. We use Dlib library to estimate the face's landmark positions, which uses the ensemble of regression trees proposed in work [43]. From these landmarks, we calculate the geometric centroid of each eye and the distance between them. We draw a straight line (see Figure 4.4) in order to get the angle to rotate the image. The rotation of the axis that crosses the two eyes is then compensated and finally, the face is cropped (see Figure 4.4). Finally, all images are converted to grayscale in range from 0 to 255 and resized to 150x150pixels.

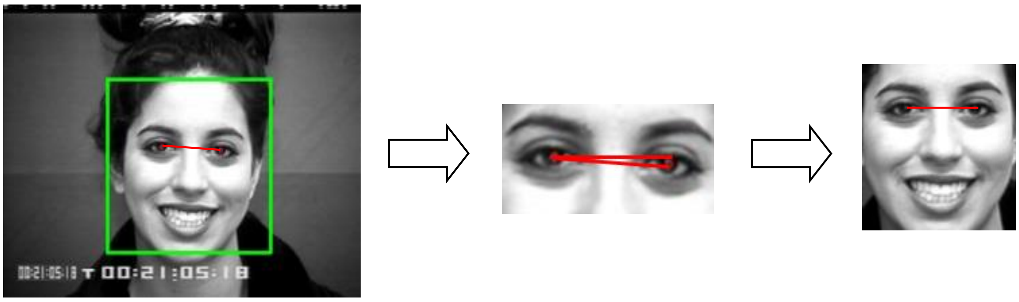


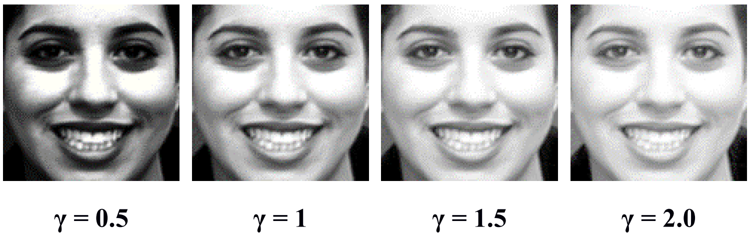
Figure 4.4. In left image we show the face detection and eyes detection. In the middle image we show the angle to rotate the image. In the right image we show the face alignment and image cropping.

A second important step is to meet the assumption that the number of training samples must be large enough and that they must contain significant facial variations. This step is in a certain sense opposite to the pre-processing one, where the face is expected to be "normalized": in other words, the variations must be reduced as much as possible. However, it is worth noting that our pre-processing algorithm is not aimed at reducing lighting and appearance variations. Only the pose and the scale variations are taken into account. Therefore, we seek, in the augmentation steps, to maintain the basic variations in the input data, and, at the same time, to add further ones in terms of lighting and appearance.

With regard to the lighting conditions, we use the gamma correction technique. Equation (4.1) is used to adjust the value of gamma,

$$y = \left(\frac{x}{255}\right)^{\frac{1}{\gamma}} \cdot 255 \qquad (4.1)$$

where x is the original image, y is the new image and γ is the value which we modify to get variations in the illumination. We use $\gamma = 0.5$, $\gamma = 1.5$ and $\gamma = 2.0$ to obtain a perceptible variation of the original image. For values of γ outside of this interval [0.5, 2.0], the face cannot be distinguished. In this way, we quadruplicate the data. Logically, $\gamma = 1$ is not used, because it does not modify the image (see Figure 4.5).



γ = 0.5 **γ = 1** **γ = 1.5** **γ = 2.0**

Figure 4.5. Images with different illuminations using the gamma correction technique.

Finally, we introduced some geometric variations, which are aimed at covering for small errors in the position of the eyes during the eyes location detection. They consist in applying a translation of 4 pixels in both axis, cropping the image (where the face is always present with two eyes, nose and mouth) and introducing a small appearance variation by duplicating the images through horizontal flip (see Figure 4.6).

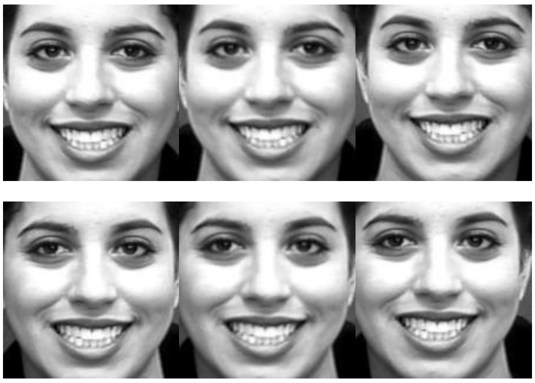


Figure 4.6. Images with different geometrics changes. On the top of the figure, the translation of 4 pixels and on the bottom, the horizontal flip are shown.

4.4 The proposed CNN

Several architectures for facial expression recognition have been developed in this last decade. The accuracy results of some of them are shown in Table 4.1. Note that the majority of architectures [44, 60, 67, 82, 89, 7] use k-cross-validation (explained in Section 4.5.2) to obtain the accuracy results reported in Table 4.1, with the exception of paper [36] which performed tests using 98% of the data for training and only 2% for testing. In [82] the authors used the pre-trained model Face-VGG. In [67] the research team designed a complex architecture using convolutional layers in parallel and combined them to obtain the final result. Papers [44, 60, 89] presented better results using more simple architectures than papers [36, 67, 82]. Although a result similar to paper [89] was presented by paper [7], which used a more complex architecture. Note that paper [60] obtained 96.76% accuracy, but the authors only tested with 1 subject for each partition of the k-cross-validation set and ran the experiment 10 times to select the best result. Their method also includes a pre-processing step, tuned using the k-cross-validation method described in Section 4.5.2, and for which they report 89.7% accuracy.

Model	**Year**	**Accuracy with CK+**
Deepak et al. [36]	2019	93.24%
Sajjanhar et al. [82]	2018	91.37%
Teixera-Lopes et al. [60]	2017	96.76%
Mollahosseini et al. [67]	2016	93.20%
Burkert et al [7]	2015	99.60%
Khorrami et al. [44]	2015	95.70%
Song et al. [89]	2014	99.20%

Table 4.1. Results of recent models in the literature. These models have been trained and tested with the CK+ dataset to classify the 6 basic expressions.

These models used the CK+ dataset in their experiments and classified the six basic expressions. The growing interest in creating new CNNs[12] to improve results in facial expression recognition has been a motivation to deepen in this field. In [30], the authors affirmed that a network with three hidden layers forms a very good generative model of the joint distribution of the images to be classified and their labels. Starting from this base and seeing the different architectures proposed by papers such as [46, 44, 89, 7], we implement a model with more than three hidden layers to recognize facial expressions. We tune this architecture and its parameters by empirical evidence, until we get a model that improves results with respect to other papers [46, 44, 89, 7] (see Table 4.16).

Number of Convolutional layers	**Mean Accuracy with CK+**	**σ**
3	90.14%	3.68
4	90.43%	3.12
5	**91.51%**	**2.57**
6	91.29%	2.68

Table 4.2. Test with different number of convolutional layers to classify 6 classes (one per facial expression). The best result is in bold text with high mean and low standard deviation.

[12] A Convolutional Neural Network (CNN) is designed to automatically and adaptively learn spatial hierarchies of features using typically three types of layers: convolution, pooling and fully connected layers. The first two layers perform feature extraction, and the fully connected layer gives the output [109].

Some of fine tuning of the model are shown in Table 4.2, where we test with three, four, five, and six convolutional layers, using the CK+ dataset for training and k-cross-validation (with k=5) (see Section 4.5.2). The average and standard deviation of the detection accuracies obtained after each step of the cross-validation are shown in Table 4.2. The best results are obtained with 5 convolutional layers, for which we achieve the highest mean accuracy and the lowest standard deviation. The addition of more convolutional layers doesn't improve the results.

The final CNN model is depicted in Figure 4.7. Our network receives as input a 150x150 grayscale image and classifies it into one of the six classes. The CNN architecture consists of 5 convolutional layers, 3 pooling layers and two fully connected layers. The first layer of the CNN is a convolutional layer that applies a kernel size of 11×11 and generates 32 images of 140×140 pixels. This layer is followed by a pooling layer that uses max-pooling, with a kernel size of 2×2 and stride 2, to reduce the image to half of its size. Subsequently, another two convolutional layers are applied with a kernel of 7×7 and a filter of 32, each. This is followed by another pooling layer, with a kernel size of 2×2 and stride 1, two more convolutional layers that apply a kernel of 5x5 and a filter of 64 each one, and two fully connected layers of 512 neurons each. The first fully connected layer also has a dropout [28] to avoid overfitting in the training. Finally, the network has six output nodes (one for each expression) that are connected to the previous layer. Although these output nodes can be changed in case of having more expressions and fine-tune again the network. The output node with the maximum value is classified as the expression of the image. Table 4.3 compares our architecture with recent proposals in the literature. Note that in [36, 7, 67] the architectures are more complex than the rest. In [36] the authors use 6 convolutional layers and 2 residual blocks which consist of 4 convolutional layers. Paper [7] uses 1 convolutional layer and 2 blocks. Each block consists of parallel path. The first path uses 2 convolutional layers and the second path uses 1 pooling layer and 1 convolutional layer. In [67] they use 2 convolutional layers and 3 modules which consist of 4 parallel convolutional layers.

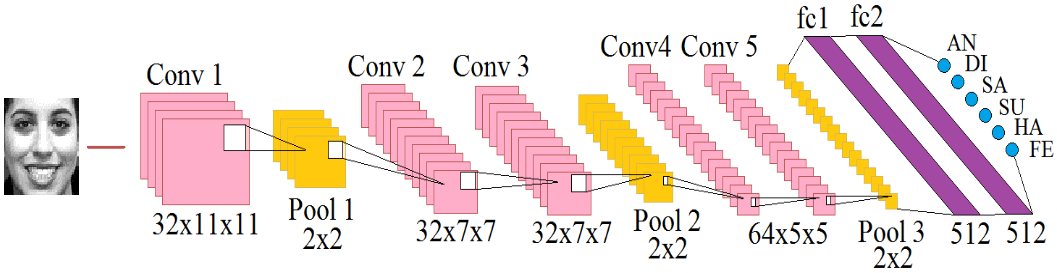


Figure 4.7. Architecture of the CNN proposed with 5 convolutional layers, 3 pooling layers and two fully connected layers.

Weight initialization is an important step in Neural Networks as a careful initialization of the network can speed up the learning process and provide better accuracy results after a fixed number of iterations. Therefore, we carried out a study of the weight initialization techniques most used in CNNs. In Table 4.4 we show accuracy results with different initializations of weights. These initializations consist of combinations of Xavier [45] (used in the experiments whose results are shown in Table 4.2), MSRA [53] and Gaussian [26] methods. The Gaussian method uses a standard deviation of 0.01. We trained our CNN using k-cross-validation (described in Section 4.5.2) with each initialization method. We can see in Table 4.4 that the combination of Xavier and Gaussian methods, and the combination of Gaussian and MSRA methods result in higher average accuracy values (marked in bold).

Model	**[36]**	**[60]**	**[7]**	**[67]**	**[82]**	**[46]**	**[44]**	**[89]**	**Our Model**
Images	128x96	32x32	224x224	48x48	224x224	224x224	96x96	96x96	150x150
LRN[13]	No	No	Yes	No	No	Yes	No	No	No
Conv.	6+2*	2	1+2*	2+3*	13	5	3	4	5
Pooling	3	2	5	4	5	3	3	4	3
Dropout	2	0	1	0	2	2	1	2	1
FC	2	1	1	2	3	3	1	1	2

Table 4.3. Results of recent models in the literature. These models have been trained and tested with the CK+ dataset to classify the 6 basic expressions. *The authors use an architecture more complex.

Initialization	**Mean**	**σ**
xavier	91.51%	2.57
gaussian	91.91%	0.94
msra	90.47%	3.07
gaussian+msra	**93.15%**	**1.32**
gaussian+xavier	91.44%	1.48
xavier+gaussian	**92.53%**	**1.63**
xavier+msra	90.62%	2.56
msra+gaussian	90.82%	1.36
msra+xavier	90.95%	1.09

Table 4.4. Test with different initializations.

Moreover, in the case of Gaussian+MSRA, we get the lowest standard deviation, meaning that all the accuracy values are close to the average. For these reasons, we have decided to use, in all our experiments, the Gaussian + MSRA initialization (i.e. a Gaussian filler is used for the convolutional layers and a MSRA filler for the fully connected layers). The loss is calculated using a logistic function of the softmax output as in several related works [82, 103, 68], the activation function of the neurons is a ReLu, which generally learns much faster in deep architectures [27] and the method used to calculate the weights between neurons is the Adam method [46], since this method shows better convergence than other methods.

[13] Local Response Normalization (LRN) is a layer that square-normalizes the pixel values in a feature map in a local neighborhood [46].

4.5 Experiments and Analysis of Results

In this Section we present five experiments to evaluate the accuracy of our system (our proposed CNN and image pre-processing) and validate our new dataset in facial expression (FEGA). In the first experiment, we show that the pre-processing step is relevant to improve the performance despite the intrinsic complexity of a CNN. In the second experiment, we have performed a subject-independent evaluation to compare our system with other works in the literature. In the third experiment, we show how merging information captured with different cameras significantly helps in the network's training; we evaluate our system using a cross-datasets protocol, in addition to comparing the results between the second and the third experiment. In the fourth experiment we compare the performance of our system with respect to other architectures. And in the fifth experiment, we validate our system by comparing its performance with the opinion of 253 human participants, to analyze whether humans and machines perform similarly in facial expression recognition.

4.5.1 Experiment 1. Role of the pre-processing step in the network's training.

Table 4.5 reports the contribution of the different preprocessing steps to the emotion classification accuracy by adding one at a time. We show the results with the six basic emotions AN (angry), DI (Disgust), FE (Fear), HA (Happy), SA (Sad) and SU (Surprise). These pre-processing steps have been used with all datasets. In Table 4.5 we have only shown results with the CK + dataset, using 80% of the images as training set and the other 20% as testing set. We have used the CNN model described in Section 4.4. The training was performed using k-cross-validation (described in Section 4.5.2) with k = 5 and only 60 epochs each time. The combinations of pre-processing steps evaluated are: (a) original images without image pre-processing, (b) face alignment and crop, (c) face alignment, crop, and illumination variations using gamma correction technique, (d) face alignment, crop, illumination variations, and geometric changes. As we can see in Table 4.5, just the image alignment adds a great improvement to the classification accuracy. But the best option is the last option (that incorporates all the pre-processing steps), which considerably improves the results with respect to the others.

Pre-processing step	**AN (%)**	**DI (%)**	**FE (%)**	**HA (%)**	**SA (%)**	**SU (%)**	**Average**
(a)	62.57	89.27	78.37	95.78	63.04	91.85	80.15%
(b)	77.83	87.96	62.42	99.02	71.20	94.27	82.11%
(c)	76.31	94.19	84.22	98.33	63.30	96.34	85.45%
(d)	95.96	97.41	82.69	100	84.88	97.93	93.15%

Table 4.5. Test with different image pre-processing with CK+ Dataset. Pre-processing steps: (a) no pre-processing, (b) face alignment and crop, (c) face alignment, crop and illumination variations, and (d) face alignment and crop, illumination variations and geometric changes.

Additionally, we found a similar work [60] which employs some of the image pre-processing steps used here. In Table 4.6 we compare our results with this recent work [60] which also used the CK+ dataset for this experiment. In [60], the used image pre-processing was (h) alignment of the face, crop (only the face without hair), down-sampling of the face image to 32x32 pixels, normalization of the image intensity, and generation of 30 more samples. Although both image pre-processing steps are quite similar, the main difference is that we apply the horizontal flip and vary the illumination in order to get more diversity of data instead of applying down-sampling and normalize the image intensity by [60]. The results show that our proposed pre-processing step is better than those reported in [60].

Pre-processing step	**AN (%)**	**DI (%)**	**FE (%)**	**HA (%)**	**SA (%)**	**SU (%)**	**Average**
(d) Our proposed	95.96	97.41	82.69	100	84.88	97.93	93.15%
(h) [60]	79.3	94.4	73.1	99.4	72.8	94.9	89.7%

Table 4.6. Comparison of results with a similar work in the literature. Pre-processing steps: (d) face alignment, crop, illumination variation and geometric changes, and (h) face alignment, crop, down-sampling, normalization and generation of 30 samples more.

4.5.2 Experiment 2. Subject-independent evaluation.

In this experiment, we have evaluated our model on each dataset separately by means of k-fold cross-validation. It consists of splitting the dataset in k groups, and using (k-1) groups as training set and the other one as testing set. We perform k-fold cross-validation using k = 5, since these values have been shown empirically to yield test error rate estimates that suffer neither from excessively high bias nor from very high variance [38].

Training Set	**Test Set**	**Model**	**Accuracy**
JAFFE	JAFFE	CNN [36]	95.23%
		CNN [60]	53.44%
		CNN [37]	**94.91%**
		CNN [82]	86.67%
		Our model	56.22%
WSEFEP	WSEFEP	LBP+KNN [68]	80.00%
		LBP+SVM [68]	78.10%
		Our model	**87.22%**
CK+	CK+	CNN [36]	93.24%
		CNN [60]	**96.76%**
		CNN [67]	93.20%
		CNN [82]	91.37%
		Our model	93.15%
BU-4DFE	BU-4DFE	CNN [60]	72.89%
		Our model	**73,58%**
FEGA (our Dataset)	FEGA (our Dataset)	Our model	72,61%

Table 4.7. Comparision of subject-independent results with related works to classify 6 expressions. The best results are shown in bold text.

In Table 4.7 we present a comparative study between our results and the recent studies in the literature to classify the six basic expressions. Although we do not get good results with the JAFFE Dataset, we can prove that our results are competitive with respect to other works. As we have explained in Section 4.3, we use the same image preprocessing for all datasets. Unfortunately, the JAFEE Dataset contains only 10 actresses and needs more data augmentation to have good results. However we get better results with the BU-4DFE and WSEFEP Datasets, and for the CK+ Dataset we get~~s~~ results close to the ones published in [36, 67]. Evidently, the FEGA Dataset cannot be used for comparison with other works because it is proposed for the first time in the present research work.

4.5.3 Experiment 3. Cross-datasets evaluation.

Unlike in Table 4.7, where good classification results are obtained using the same dataset to train and test the system, this no longer holds when the training and test datasets are different (cross-datasets protocol).

Training Set	**Cross-datasets experiment**		**Subject-independent experiment**	
	Test Set	**Accuracy**	**Test Set**	**Accuracy**
BU-4FDE	FEGA	24.96%	BU-4FDE	73.58%
	JAFFE	43.17%		
	WSEFEP	**62.22%**		
	CK+	47.90%		
FEGA	BU-4DFE	44.67%	FEGA	72,61%
	JAFFE	37.70%		
	WSEFEP	73.89%		
	CK+	**78.19%**		
JAFFE	BU-4DFE	32.49%	JAFFE	56.22%
	FEGA	23.42%		
	WSEFEP	26.67%		
	CK+	**42.97%**		
WSEFEP	BU-4DFE	45.61%	WSEFEP	87.22%
	FEGA	36.55%		
	JAFFE	34.43%		
	CK+	**59.22%**		
CK+	BU-4DFE	48.60%	CK+	93.15%
	FEGA	48.45%		
	JAFFE	23.50%		
	WSEFEP	**67.78%**		

Table 4.8. Cross-datasets evaluation to classify 6 expressions. One dataset as training set against four datasets as testing set. The best results are shown in bold text.

In Table 4.8 we show a comparative between both experiments, using our CNN and our image pre-preprocesing. As we can see, when we apply the cross-datasets protocol, the results are generally worse. This is because each dataset contains pictures of people of different ethnicities and ages, with different illuminations and backgrounds. For example the BU-4DFE Dataset contains Asian, Black, Latin and White people. The CK+ Dataset contains mostly Euro-American people and in a minority, it contains Afro-American and other groups of ethnicities. However, both FEGA and WSEFEP Datasets contain mostly white people. And finally the JAFFE Dataset is only of Japanese females. It is thus normal that when we train with datasets as FEGA and WSEFEP we obtain better results when testing against the CK+ Dataset, because it contains Euro-American people in its majority. However CK+ and BU-4DFE obtain best results when tested against the WSEFEP Dataset. This suggests that CK+ and BU-4DFE contain an adequate number of white people to be tested with other Dataset with white people. As for example, with the WSEFEP Dataset, whose expressions are very clear. On the other hand the JAFFE Dataset is very small for training and the accuracy results when testing against any other dataset are very low.

Additionally, we study if our new dataset FEGA can be a good dataset to train a face expression recognition system. In Table 4.8, we see that it can achieve acceptable results with two test datasets (WSEFEP and CK+), because our dataset only contains Caucasian people in its majority. In the case of JAFFE and BU-4DFE, we obtain worse results because these two datasets contain Japanese people and, in the case of BU-4FDE, it also contains Afro-American people. Therefore, FEGA can be considered as a good dataset to train the emotions, since it produces very good results when it is tested with white people, as in the WSEFEP and CK+ Datasets.

Train	BU-4DFE	FEGA	JAFFE	WSEFEP	CK+	Mean
5 DBs	100%	98.56%	98.29%	95.55%	95.93%	**97.67%**

Table 4.9. Results of classification of the different datasets. We use the five combined datasets (BU-4DFE, FEGA, JAFFE, WSEFEP and CK+) as training set and we test it with each dataset applying k-fold cross-validation.

All this suggests that a solution to get satisfactory results is a good combination of different datasets to train the system, which contain all type of ethnicity, age and gender with different illuminations and backgrounds (see Table 4.9). The high accuracy results of Table 4.9 show that the CNN distinguishes well between different datasets when it is trained with a dataset containing data with sufficient diversity.

Hence, we can affirm that each dataset adds an important value in the training. This may also be not only because of the diversity in the population, but because of the different capture conditions of each dataset. Therefore, we have experimented with different combinations of datasets for the training set and have used the other datasets as testing sets.

4.5.3.1 Different combinations of datasets.

In order to maximize the success of a neural network model $\mathcal{R}$ using N datasets. We define the set of datasets used for learning as $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_N\}$, where N corresponds to number of available datasets. To get the best combination of datasets, we have to test all possible combinations for each subset of $\mathcal{D}$ (see Tables 4.10, 4.11 and 4.12), except for the empty set $\emptyset$. Each table is divided into four groups of combinations (without combining datasets, 2 combined datasets, 3 combined datasets and 4 combined datasets). The number of combinations to be tested is $card\ (\mathcal{P}\ (\mathcal{D}))$ - $1 = 2^N$ - 1. In case of having four datasets, it would be necessary to train and evaluate the network 2^4 -1 = 15 times. We would like to reduce this number of combinations, that could be unaffordable using 30 datasets.

We define the function $f(\mathcal{R}, T)$, where T is the Test dataset, and $\mathcal{R}$ is a neural network model.

The function *f* returns the subset of $\mathcal{D}$ that achieve best accuracy. We define the function $f_k(\mathcal{R}, T)$ for *k* in [1, *N*]. This function returns the subset of $\mathcal{D}$ using *k* elements that achieve best accuracy. For example, f_3(Our Model, JAFFE) would return *{WSEFEP, BU-4DFE, FEGA}*. Note that $f(\mathcal{R}, T)$ can be computed from all $f_k(\mathcal{R}, T)$ by comparison. As we can see in Tables 4.10, 4.11 and 4.12, for each test set, we train 15 different combinations. This means a high computational cost, if the number of datasets augments.

In each group of each table (4.10, 4.11 and 4.12), the best result is highlighted in bold. Note that the best result of each group contains the dataset of the previous combination. This gives as a hint on how to reduce the number of combinations of datasets that need to be checked in order to obtain the optimum result. We devise an iterative procedure in which *k* datasets are used at each step. If we denote as B_k the optimum set of datasets used at step k ($B_k = f_k(\mathcal{R}, T)$), B_k is defined as $B_{k-1} \cup \mathcal{D}_{k^*}$, where $\mathcal{D}_{k^*}$ is the dataset in $\mathcal{D}$-B_{k-1} which maximizes the accuracy of the combined dataset $B_{k-1} \cup \mathcal{D}_j$, for $\mathcal{D}_j \in \mathcal{D}$-$B_{k-1}$. By definition $B_0 = \emptyset$.

Under this premise, we can obtain the best training with a computational cost N^2. An example for N=4 is shown in (4.2),

$$\begin{aligned} B_0 &= \emptyset \\ B_1 &= \{\mathcal{D}_3\} \\ B_2 &= \{\mathcal{D}_3, \mathcal{D}_1\} \\ B_3 &= \{\mathcal{D}_3, \mathcal{D}_1, \mathcal{D}_4\} \\ B_4 &= \{\mathcal{D}_3, \mathcal{D}_1, \mathcal{D}_4, \mathcal{D}_2\} = \mathcal{D} \end{aligned} \qquad (4.2)$$

where $\mathcal{D}j$ is the added dataset that maximizes the function $f(\mathcal{R}, T)$. Until arriving at the solution for $k = N$, $B_N = \mathcal{D}$. With this procedure, we reduce the number of trainings to needed to obtain B_N to:

$$\sum_{i=1}^{N} i = \frac{N*(N+1)}{2} \qquad (4.3)$$

For example, if N = 100, instead of training the neural network $\mathcal{R}$ with 2^{100} - 1 combinations of datasets, we would reduce the number of trainings to 5050 combinations.

$$\sum_{i=1}^{100} i = \frac{100*(100+1)}{2} = 5050 \qquad (4.4)$$

One aspect to keep in mind is that adding a new dataset does not always improve the results. So, we must determine the value of *k* for which the function, $f\ (\mathcal{R}_{Bk}, T)$, is maximum. In Tables 4.10, 4.11 and 4.12, we show the best result in bold. But as we want to generalize the system, we search the best combination in the majority of cases. By and large, four combined datasets achieved better results in the majority of cases. In Table 4.13, we show results with the following combinations of datasets for training: (a) FEGA, CK+, BU-4FDE and WSEFEP, (b) FEGA, CK+, BU-4FDE and JAFFE, (c) FEGA, BU-4FDE, JAFFE and WSEFEP, (d) FEGA, CK+, JAFFE and WSEFEP, and (e) JAFFE, CK+, BU-4FDE and WSEFEP. These results have been compared with other related works [60, 67, 112] that use CNN of six classes (one for each facial expression). It can be seen that a good combination of training datasets improves the results. Our results are better in the majority of cases, only the work [112] is better when testing with CK+. In our knowledge, only works [67] and [112] combine several datasets. In [67], they combine MultiPIE, MMI, DISFA, FERA, SFEW, and FER2013 Datasets as training set and use CK+ as testing set (f). And in [112], they combine JAFFE, MMI, RaFD, KDEF, BU3DFE and ARFace Datasets to test with CK+ (g) and combine CK+, MMI, RaFD, KDEF, BU3DFE and ARFace Datasets to test with JAFFE (h). Unfortunately, we have not found works which have been tested with the WSEFEP dataset in a cross-datasets evaluation scenario. And naturally the FEGA dataset has been tested for the first time.

Therefore, in summary, the combination of several datasets to train the system improves the results according to Tables 4.9 and 4.13. Based on these reflections, we detail the results obtained when we train with the four datasets of the case (c) of Table 4.13, and when we train with the five datasets (see Table 4.14). We have performed k-fold cross-validation using k = 5 to classify both six and seven expressions using our CNN. That is, we separate these datasets (4 combined DBs and 5 combined DBs) in 5 blocks both in the training set and in the test set. For example, we train with blocks 1, 2, 3 and 4 (with data augmentation), and we test with block 1 (without data augmentation). Each block consists of fifth part of all combined dataset (BU4DFE, CK+, JAFFE, FEGA and WSEFEP).

Training Set	**Test Set**	**Accuracy**	**Training Set**	**Test Set**	**Accuracy**
Without combining Datasets					
FEGA	BU-4FDE	44.67%	BU-4DFE	FEGA	24.96%
JAFFE		32.49%	JAFFE		23.42%
WSEFEP		45.61%	WSEFEP		36.55%
CK+		**48.60%**	**CK+**		**48.45%**
2 Combined Datasets as training set					
WSEFEP & CK+	BU-4FDE	**54.36%**	**WSEFEP & CK+**	FEGA	**54.79%**
FEGA & CK+		52.04%	BU-4DFE & CK+		51.47%
FEGA & WSEFEP		52.19%	BU-4DFE & WSEFEP		39.88%
JAFFE & CK+		52.82%	JAFFE & CK+		48.61%
FEGA & JAFFE		54.21%	BU-4DFE & JAFFE		26.66%
WSEFEP & JAFFE		43.55%	WSEFEP & JAFFE		38.56%
3 Combined Datasets as training set					
WSEFEP, CK+ & JAFFE	BU-4FDE	**57.72%**	WSEFEP, JAFFE & CK+	FEGA	48.92%
FEGA, WSEFEP & CK+		53.61%	**WSEFEP, CK+ & BU-4DFE**		**51.16%**
FEGA, JAFFE & WSEFEP		50.32%	BU-4DFE, JAFFE & WSEFEP		42.43%
JAFFE, FEGA & CK+		54.06%	JAFFE, BU-4DFE & CK+		46.06%
4 Combined Datasets as training set					
WSEFEP, CK+, JAFFE & FEGA	BU-4DFE	**57.68%**	**WSEFEP, CK+, BU-4FDE & JAFFE**	FEGA	**55.02%**

Table 4.10. Results with different combinations of datasets for the testing set of BU-4DFE (on the left) and FEGA (on the right).

Training Set	Test Set	Accuracy	Training Set	Test Set	Accuracy
Without combining Datasets					
FEGA	JAFFE	37.70%	BU-4DFE	WSEFEP	62.22%
BU-4DFE		**43.17%**	JAFFE		26.67%
WSEFEP		34.43%	FEGA		73.89%
CK+		23.50%	**CK+**		**67.78%**
2 Combined Datasets as training set					
WSEFEP & CK+	JAFFE	46.45%	**CK+ & FEGA**	WSEFEP	**82.22%**
FEGA & CK+		41.53%	BU-4DFE & CK+		78.33%
FEGA & WSEFEP		26.23%	BU-4DFE & FEGA		78.89%
BU-4DFE & CK+		45.90%	JAFFE & CK+		73.33%
FEGA & BU-4DFE		53.55%	BU-4DFE & JAFFE		55.00%
WSEFEP & BU-4DFE		**60.66%**	FEGA & JAFFE		73.33%
3 Combined Datasets as training set					
WSEFEP, BU-4DFE & CK+	JAFFE	49.18%	FEGA, JAFFE & CK+	WSEFEP	81.67%
FEGA, WSEFEP & CK+		38.25%	**CK+, FEGA & BU-4DFE**		**84.44%**
BU-4DFE, FEGA & CK+		49.73%	JAFFE, BU-4DFE & CK+		74.44%
WSEFEP, BU-4DFE & FEGA		**59.02%**	BU-4DFE, JAFFE & FEGA		81.67%
4 Combined Datasets as training set					
WSEFEP, BU-4DFE, FEGA & CK+	JAFFE	**46.45%**	**JAFFE, CK+, BU-4FDE & FEGA**	WSEFEP	**84.44%**

Table 4.11. Results with different combinations of datasets for the testing set of JAFFE (on left) and WSEFEP (on right).

Training Set	Test Set	Accuracy
Without combining Datasets		
FEGA	CK+	**78.19%**
BU-4DFE		47.9%
WSEFEP		59.22%
JAFFE		42.97%
2 Combined Datasets as training set		
WSEFEP & JAFFE	CK+	55.29%
FEGA & JAFFE		74.54%
FEGA & WSEFEP		**81.11%**
BU-4DFE & JAFFE		52.65%
FEGA & BU-4DFE		79.84%
WSEFEP & BU-4DFE		73.18%
3 Combined Datasets as training set		
WSEFEP, BU-4DFE & JAFFE	CK+	69.71%
FEGA, WSEFEP & JAFFE		77.01%
BU-4DFE, FEGA & JAFFE		81.30%
FEGA, WSEFEP & BU-4DFE		**82.66%**
4 Combined Datasets as training set		
FEGA, JAFFE, BU-4DFE & WSEFEP	CK+	**84.76%**

Table 4.12. Results with different combinations of datasets for the testing set of CK+.

Training Set	**Test Set**	**Model**	**Accuracy**
Our 4 combined DBs (a)	JAFFE	Our model	**46.45%**
Only BU-4FDE		Our model	43.17%
6 combined DBs (h)		CNN [112]	44.32%
Only CK+		CNN [60]	38.80%
Our 4 combined DBs (b)	WSEFEP	Our model	**84.44%**
Only FEGA		Our model	73.89%
Our 4 combined DBs (c)	CK+	Our model	84.76%
Only FEGA		Our model	78.19%
6 combined DBs (f)		CNN [67]	64.20%
6 combined DBs (g)		CNN [112]	**88.58%**
Our 4 combined DBs (d)	BU-4DFE	Our model	**57.68%**
Only CK+		Our model	48.60%
Only CK+		CNN [60]	45.91%
Our 4 combined DBs (e)	FEGA	Our model	**55.02%**
Only CK+		Our model	48.60%

Table 4.13. Comparison of cross-datasets results with related works to classify 6 expressions. Combinations of datasets: (a) FEGA, CK+, BU-4FDE and WSEFEP, (b) FEGA, CK+, BU-4FDE and JAFFE, (c) FEGA, BU-4FDE, JAFFE and WSEFEP, (d) FEGA, CK+, JAFFE and WSEFEP, and (e) JAFFE, CK+, BU-4FDE and WSEFEP, (f) MultiPIE, MMI, DISFA, FERA, SFEW, and FER2013, (g) JAFFE, MMI, RaFD, KDEF, BU3DFE and ARFace, and (h) CK+, MMI, RaFD, KDEF, BU3DFE and ARFace.

Training Set	**BU-4DFE (Test)**	**FEGA (Test)**	**JAFFE (Test)**	**WSEFEP (Test)**	**CK+ (Test)**	**Mean**
4 DBs (6 expressions)	71.67%	74.42%	60.22%	88.89%	--	**73.80%**
5 DBs (6 expressions)	74.56%	76.32%	68.45%	92.22%	94.07%	**81,12%**
4 DBs (7 expressions)	71.79%	72.31%	65.36%	90%	--	**74.86%**
5 DBs (7 expressions)	71.14%	74.10%	70.09%	91.91%	93.23%	**80.10%**

Table 4.14. Comparison between results with four (4 DBs) and five (5 DBs) combined datasets to classify 6 and 7 expressions.

Training Set	**BU-4DFE (Test)**	**FEGA (Test)**	**JAFFE (Test)**	**WSEFEP (Test)**	**CK+ (Test)**	**Mean**
1 DB (Subject-indep. experiment) (6 expressions)	73.58%	72.61%	56.22%	87.22%	93.15%	**76.48%**
5 DBs (combined datasets) (6 expressions)	74.56%	76.32%	68.45%	92.22%	94.07%	**81,12%**

Table 4.15. Comparison between results with five combined datasets (5 DBs) and the results of the subject-independent experiment (1 DB) for the classification of 6 expressions.

Therefore, in order to know the test accuracy in each dataset, we separate the images of this test block in each dataset. Results improve in all cases when training with five datasets, except when testing with BU-4FDE, which obtains similar results in the case of seven expressions. On the whole we improve the results to 80.1% in the test accuracy to classify seven expressions and 81.12% to classify six expressions. In addition, we also improve the accuracy achieved when only the same dataset is used for training and testing (see Table 4.15).

4.5.4 Experiment 4. Comparison of our system with others architectures

In order to verify the proper functionally of the CNN of our system, we have compared it with several CNNs [46, 44, 89, 7], using the five combined datasets for training and our image pre-processing. We have implemented the architectures [44, 89, 7] following the descriptions in the corresponding papers. These models were specifically created for the facial expression recognition task. Moreover, we have also tested the performance of the well-known AlexNet [46] network architecture, which is available in Caffe[14]. The results from each CNN are shown in Table 4.16. As we can see, our CNN shows the best results in the majority of cases. Only for the BU-4DFE test dataset it is overcome by the SongNet network [89]. Therefore we can affirm that our CNN is competitive with respect to other existing CNNs and that works well for facial expression recognition.

Model	**BU-4DFE**	**FEGA**	**JAFFE**	**WSEFEP**	**CK+**	**Mean**
AlexNet [46]	71.21%	70.32%	67.56%	90.95%	91.76%	78.36%
Burkert et al. [7]	57.09%	55.61%	45.66%	70.48%	83.44%	62.46%
Khorrami et al. [44]	72.31%	73.62%	68.43%	91.43%	90.82%	79.32%
Song et al. [89]	**72.67%**	69.57%	64.33%	88.09%	88.36%	76.60%
Our Model	71.14%	**74.10%**	**70.09%**	**91.91%**	**93.23%**	**80.10%**

Table 4.16. Results of the five (5 DBs) combined datasets to classify 7 expressions using different architectures and the same image pre-processing.

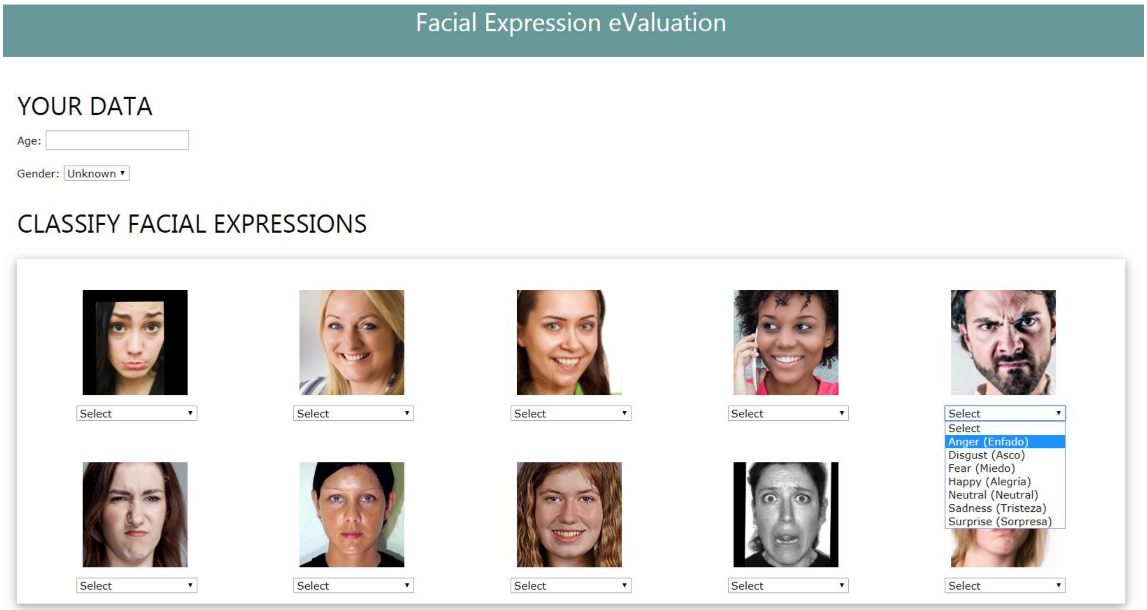
4.5.5 Experiment 5. A Comparative Study between Human and Machine using FE-Test.

In this fifth experiment we want to know how closely these results are in comparison with human perception. For this, we made two principal experiments using the FE-Test dataset: Facial Expression Recognition by Humans and by our system. Both experiments had to classify these images in the six basic emotions (anger, disgust, fear, happy, sadness and surprise) and one more, neutral expression. It should be highlighted that the test dataset (FE-Test) used in this sub-Section has not been employed previously in the Chapter. Therefore, we also aim to test how our system generalizes to new unseen data. The FE-test dataset contains facial expression images "in the wild" from Internet validated by research team. We decided to create this dataset because we consider that they are more realistic than the images used in classical datasets, taken with constant background and controlled illumination conditions. For this reason, recognition is

[14] Caffe is a deep learning framework made with expression, speed, and modularity in mind.

also expected to be more difficult. Facial Expression Recognition by Humans

This experiment was done with a set of 253 participants between 18 and 66 years old, 27.27% females and 72.72% males. For this, we created a web page, where 10 images were shown to each participant, who had to classify each image in one of the seven emotions (AN (angry), DI (Disgust), FE (Fear), HA (Happy), SA (Sad), SU (Surprise) and NE (Neutral)). In Figure 4.8, an image of this experiment is shown. It is a picture of the web page where the participants classified several faces according to their criteria in facial expression recognition.



Facial Expression eValuation

YOUR DATA

Age:

Gender: Unknown

CLASSIFY FACIAL EXPRESSIONS

Figure 4.8. The created web page for the experiment of Facial Expression Recognition by humans.

The confusion matrix is shown in Table 4.17, where we obtain a total mean of 83.53% in the accuracy of this testing set evaluated by humans. Also, we can observe that there are some expressions that can be confused by others as Sad and Fear, which are often confused with Neutral and Surprise, respectively. However, Happy is the most clear to distinguish, and most of the participants recognize it easily.

	AN	**DI**	**FE**	**HA**	**NE**	**SA**	**SU**	**Total**
AN	**329**	**21**	5	2	3	2	3	90.14 %
DI	**23**	**303**	14	3	1	14	7	83.01 %
FE	7	22	**243**	0	1	5	**88**	66.39 %
HA	1	2	2	**331**	12	2	2	**94.03 %**
NE	6	4	5	13	**331**	5	0	90.93 %
SA	7	12	11	2	**45**	**276**	7	76.67 %
SU	5	4	13	**29**	7	1	**299**	**83.52 %**

Table 4.17. Confusion Matrix from human assessment (7 expressions). Results of the FE-Test dataset (described in Section 4.2.2) using the cross-datasets protocol.

4.5.5.1 Facial Expression Recognition by our system

In this experiment, we employ our CNN and image pre-processing steps to classify expressions of the FE-Test dataset. We first study the capability of the system to discriminate between 6 expressions. The system is trained in two ways: with each of the five datasets separately available from previous Sections, and with the five combined datasets together. In Table 4.18, we can see that better results are obtained with the combination of five datasets than with only one dataset. As we see, we improve the results up to a 37.22% (the worst result was with the JAFFE as training dataset, which got 32.78%, while using the 5 DBs for training resulted in a 70% accuracy). And we improve even in the best case with a 6.11%.

Training set	**FE-Test (Test)**
BU-4FDE	48.33 %
FEGA	63.89 %
JAFFE	32.78 %
WSEFEP	63.33 %
CK+	53.89 %
5 DBs	**70.00 %**

Table 4.18. Results of the test FE-Test (6 expressions).

Therefore, in the case of 7 expressions we use the combination of five datasets as training set using our CNN and pre-processing step. We used the FE-Test as testing set, thus this is also a cross-datasets evaluation.

	AN	**DI**	**FE**	**HA**	**NE**	**SA**	**SU**	**Total**
AN	**73**	**55**	2	4	3	2	11	48.67 %
DI	**22**	**117**	7	0	0	4	0	78.00 %
FE	6	5	**82**	0	3	5	**49**	54.67 %
HA	0	1	2	**147**	0	0	0	**98.00 %**
NE	**31**	1	5	5	**101**	6	1	67.33 %
SA	**29**	**24**	6	2	9	**74**	6	49.33 %
SU	1	2	10	4	0	4	**129**	**86.00 %**

Table 4.19. Confusion Matrix from our system (7 expressions). Results of the FE-Test dataset using the cross-datasets protocol.

The confusion matrix is shown in Table 4.19, where we have obtained a total average of 68.86% in the accuracy. The best accuracy is obtained with Happy and Surprise, with machines (CNN) performing better than humans. Also, in both experiments we obtained the worst results in Sad and Fear, although humans are better than machines in this case. Humans also recognized better the neutral, angry and disgust emotions.

Finally, we can also see a correlation between the experiments, especially in the recognition of Angry, Disgust and Fear, which are usually confused with Disgust, Angry and Surprise, respectively. Interestingly, these mistakes are done both by humans and machines, that is, both perform similar misclassifications.

4.6 Conclusions

In this Section we showed that: (1) the pre-processing step is relevant to improve the performance despite the intrinsic complexity of a CNN; (2) merging information captured with different cameras significantly helps in the network's training; (3) facial expression classification from non-expert humans is correlated with the one of the CNN (especially in the recognition of Angry, Disgust and Fear) that is, we can see that the same types of facial expressions are misclassified by both the humans and the neural network.

Several experiments have been performed to build our proposed CNN and find the adequate steps for image preprocessing. We have evaluated the system using six datasets, including two new datasets (FEGA and FE-Test) that are presented in this Section. One of the captured datasets (FEGA) is the first one in the literature including simultaneously labeling of facial expression, gender and age of the individuals. Another contribution is the combination of different datasets to train our system. Up to our knowledge, this is the most extensive experimental study to date in cross-dataset facial expression recognition using CNNs, since most previous studies in the literature only employ one dataset for testing. Our study shows that each dataset adds an important value in the training, because each one of them has been captured in different conditions, and contains people from different ethnicities and ages. Therefore, not only the quantity is important to train the data with CNN, but also the variety of information. Thus, the combination of these datasets into one single training dataset, using our image preprocessing steps to unify them improves significantly the results with respect to using only one dataset for training. Furthermore, we have got about 70% in accuracy using the cross-datasets protocol when the test set comes from a never-seen-before dataset. Finally, we have performed a comparative study of facial expression classification using our system vs. human opinion. The experiments show that our system outperforms other proposed solutions in the literature (Table 4.16), in addition to get good accuracy results in real world situations. Also, we have observed that humans and machine are prone to similar misclassification~~s~~ errors. As future work, we intend to refine our system with more datasets, in addition to studying the pre-processing step for color images. We also plan to extend this study using age and ethnicity to develop a new multimodal system, more robust, for facial expression recognition.

Chapter 5

Evaluation on Social Robots

In Chapter 4, we designed a system based on a convolutional neuronal network and a specific image preprocessing for facial expression recognition. We used a combination of five datasets to get about 70% in accuracy using the cross-datasets protocol when the network is tested with a dataset unseen in the training. In this Chapter, we describe an application with social robots to evaluate our system in a real environment.

Section 5.1 introduces the context of this work and the most relevant related literature. In the next Section, we explain the performed experiment. In Section 5.3, we explain the design and procedure in detail. Section 5.4 is devoted to analyze the obtained results. The last Section shows the conclusions, review the main contributions and propose future lines of work.

5.1 Introduction

Facial expression recognition plays an important role in recognizing and understanding human emotion by robots [11]. Studies as [41] have demonstrated that a robot can affect its social environment beyond the person who is interacting with it. For example, studies of robots in autism therapy [83] show that robots can influence how children interact with others. For that reason, facial expression recognition is important to shape a good human-robot interaction and get a better user experience. Since social robots can simulate empathy and decide the best way to interact according to the facial expression of the user. Robots equipped with expression recognition capabilities can also be a useful tool to get feedback in videogames, for example, since they can assess the degree of satisfaction of the users. They can act as mediators, motivate the user and adapt the game according to the user's emotions. On the other hand, many papers have demonstrated that the use of robots in the field of rehabilitation has a considerable effect in the improvement of the patients [91, 73, 24]. There are several types of social robots in the current market [92]. But we can highlight the robot NAO [69], which is a humanoid robot with friendly aspect and pleasant voice. This contributes to have a better user experience. Many papers have used the social robot NAO [69] in their experiments as in [32, 14, 94, 9], where the social component of natural interaction is common to all the proposed applications, in addition to be a tool for motivation in rehabilitation sessions.

Given the growing interest in Human-Robot Interaction [87], we have created an advanced interaction system using the social robot NAO. The system consists in a serious game to evaluate the facial expression made by the user in front of social robot. The robot acts as if it were an evaluator of actors and actresses. Then the robot interacts with the person according to his or her facial expression. With each recognized expression, the robot responds with a positive phrase to encourage the user with the game. In the design of the facial expression recognition system we have used the trained network described in Chapter 4. The experiment has as main goal the evaluation of our trained network in real environments with non-expert users. Although, the system can also be used in several ways:

- As a user-experience evaluation tool, where the robot is adapted according to the user's emotions (positive feedback).
- As a tool for training the emotions, where the social robot acts as a supervisor of the user's level of success regarding the emotion performed. This system allows replicating and learning in a playful way seven facial expressions (happy, sadness, disgust, anger, surprise, fear and neutral). This kind of experiment also seeks to encourage attention and motivation of users, especially people with special needs, as for example children with autism. In [24], the authors affirm that the social robot can be a good imitation and interaction tool for children with attention deficit disorder (ADD) syndrome.
- As a new capture method to get a new dataset on facial expressions "on the flight" through natural interaction with the game.

Finally, this experiment allowed the evaluation of our system with 29 non-expert participants, in addition to evaluate the interaction of the robot (dialogues and the fluidity of movements) with the participant, the attention (level of user’s concentration) and the difficulty to express an emotion through a final interview with each participant. Since the participants were non-experts in this field, some of them did not know how to express some emotion. Therefore, the results obtained by the CNN were also compared with the ground truth provided by 10 experts in facial expression recognition, in order to validate the system. We have considered as experts the 10 persons that ranked best in an initial test with 30 participants and which obtained a hit rate of 100% in experiment 5 of Chapter 4 (sub-Section 4.5.5.1).

5.2 Experiment

The goal of this study is to measure both the interaction and the attention of users with a Social Robot. In addition to evaluate our trained neural network in real time with a completely new set of users.

A total of 29 people participated in the experiment. Each participant was evaluated individually and signed the informed consent at the beginning of the experiment, since our robot captured his or her images. The participant sat in front of the robot (see Figure 5.1) and followed the instructions of NAO, without any help of the interlocutor. The robot began with an explanatory presentation of the game and involved the user by addressing him or her by name, to give a sense of personalized application. In this presentation, the robot acts as if it were an evaluator of actors and actresses, challenging the participant to perform each one of the 6 basic expressions (happy, sadness, disgust, anger, surprise and fear) [20], in addition to the neutral expression. Each expression was evaluated with the CNN proposed in Chapter 4. Then, the robot maintained a certain dialogue with the user depending on the recognized expression. These dialogs are usually funny phrases in relation to the expression made, and therefore, users usually smile and have a better user experience (see Figure 5.2). In Figure 5.1, we show the experiment with one of the participants. In this figure, we capture the moment when the user interpreted the expression of surprise. This emotion was analyzed by the social robot to interact with the user. In Figure 5.2, we show a natural reaction of the participant when he heard the robot's answer. Finally, the participants performed a questionnaire at the end of the experiment, where they evaluated this new experience in terms of interaction with the robot, attention in the game and difficulty of expression, among others (see Section 5.4.1.3).

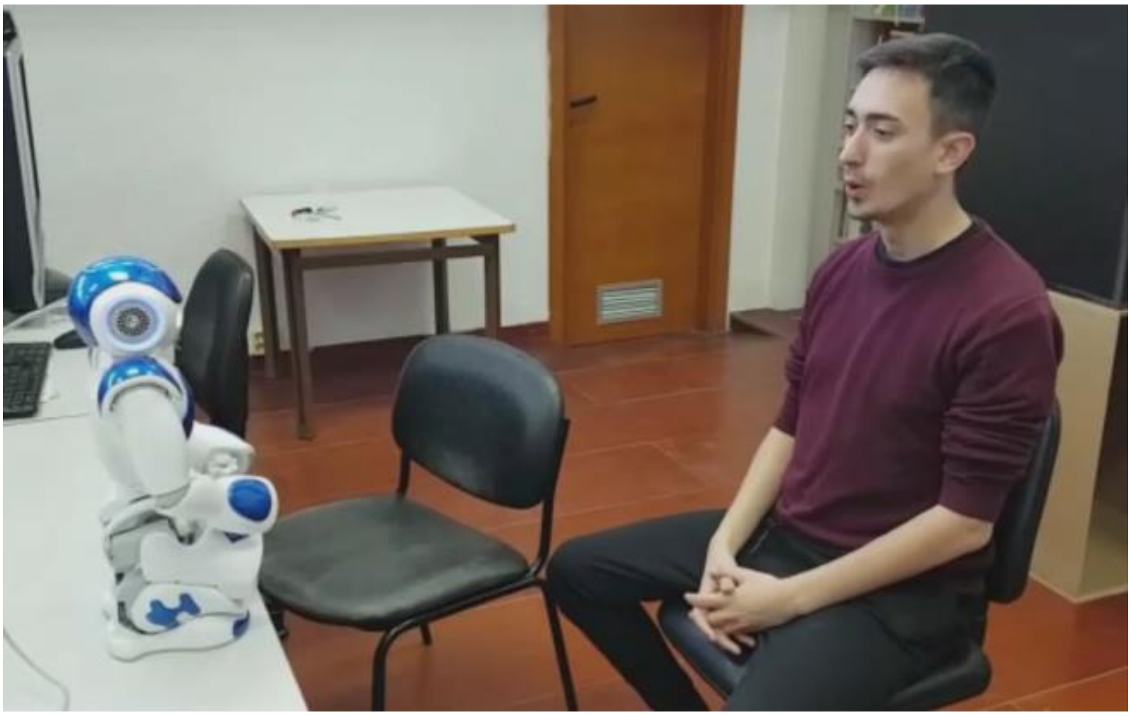


Figure 5.1. Interaction between the participant and the NAO Robot. In this capture the robot recognize the expression performed by user.

5.2.1 Participants

The experiment was performed with 29 participants between 18 and 38 years old (mainly students and teachers of the University of Balearic Islands), with an average age of 23.34 years. 41% were women and 59% were men. 97% of the participants did not have any previous experience with the NAO Robot or any other social robot. 79% of the participants considered themselves bad actors, compared to 21% who considered themselves good actors for this experiment.

5.2.2 Sessions

This experiment consisted in two sessions. Each session was launched in a personalized way with the name of the participant. In the first session, the social robot introduced itself and gave the instructions to the user. The user had to interpret a sequence of emotions. This sequence consisted on interpreting the emotions from easier to more difficult, with the neutral expression in the middle position. The expressions of happiness, surprise and anger were considered as the easiest expressions, since these emotions were the best rated in Table 4.17 (Section 4.5.5.1). The expressions of sadness, fear and disgust were considered as the most complicated to represent. In the second session the same exercise was performed but with a different presentation of the robot, much shorter, because the user already knew the game.

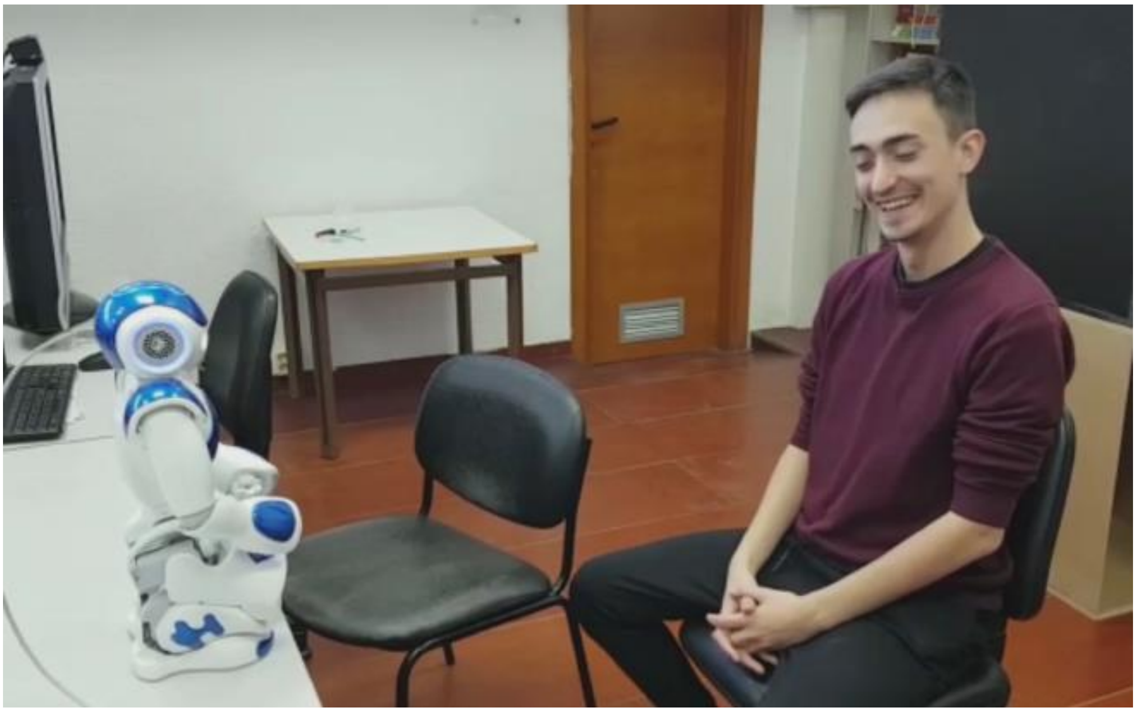


Figure 5.2. The reaction of the participant is shown in this figure. The robot's answer respect to the emotion performed by the participant provoked good reactions.

5.3 Design and Procedure

First step was to guarantee an efficient interaction, without delays in the response and allowing a fluid natural communication. For this reason, part of the processing is done on a computer via WiFi connection, since the CPU of the NAO Robot is not very powerful.

The NAOqi SDK is a software development kit, which manages and controls both the verbal communication and the movement of the engines of NAO. In this application we use this software to create a fluid movement with the arms of the robot to simulate a gestural interaction and gain the user's attention. These movements were performed synchronously when the robot was talking, to simulate a real dialogue. The frontal camera of the robot takes pictures with a resolution of 1280x960 pixels, to acquire images of the user, which are fed to the system described in Chapters 3 and 4 for facial expression recognition.

5.3.1 Image pre-processing and CNN

The images captured by the NAO robot are first analyzed by the method proposed in Chapter 3 to detect whether there is a face or not. If the face is detected, the image is cropped, converted to grayscale and aligned to eliminate possible face rotations (see Figure 4.1 in Chapter 4). This pre-processing step is important for a good recognition by our proposed CNN, since our trained neural network uses this first pre-processing step in the training set.

Finally, the image is processed by our CNN, described in Chapter 4, to obtain the recognized expression. Since none of the participants of this experiment were included in any of the datasets used for training, the results of this experiment can be considered as a test set that evaluates this CNN in a real environment.

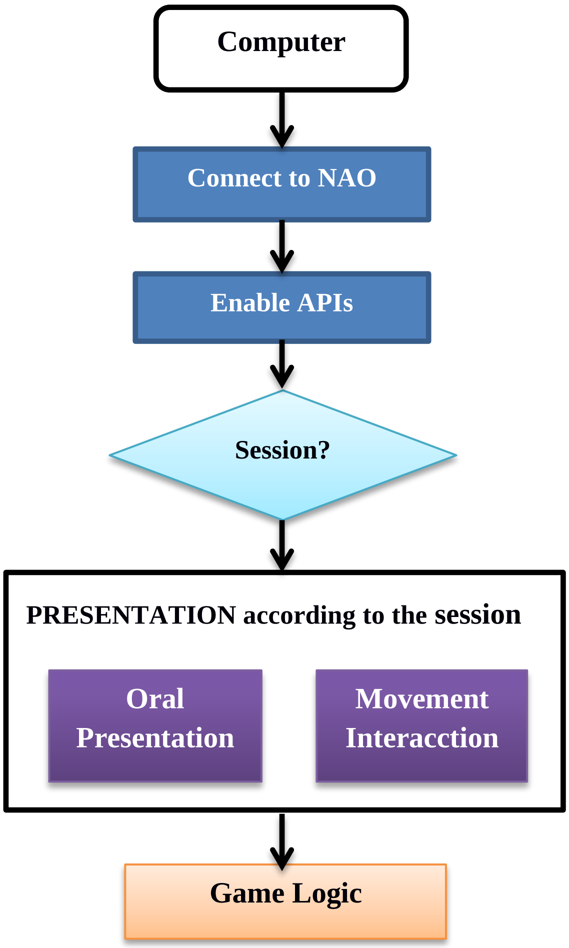


Figure 5.3. Game initializations. The interlocutor introduces the name of the user and selects the session in which the user will play.

5.3.2 Application Design

In this sub-Section the structure of the game is explained (see Figure 5.3). First, a connection is established between the computer and the NAO Robot. Second, the APIs responsible of speech and movement are enabled for allowing to initiate the interaction with the user.

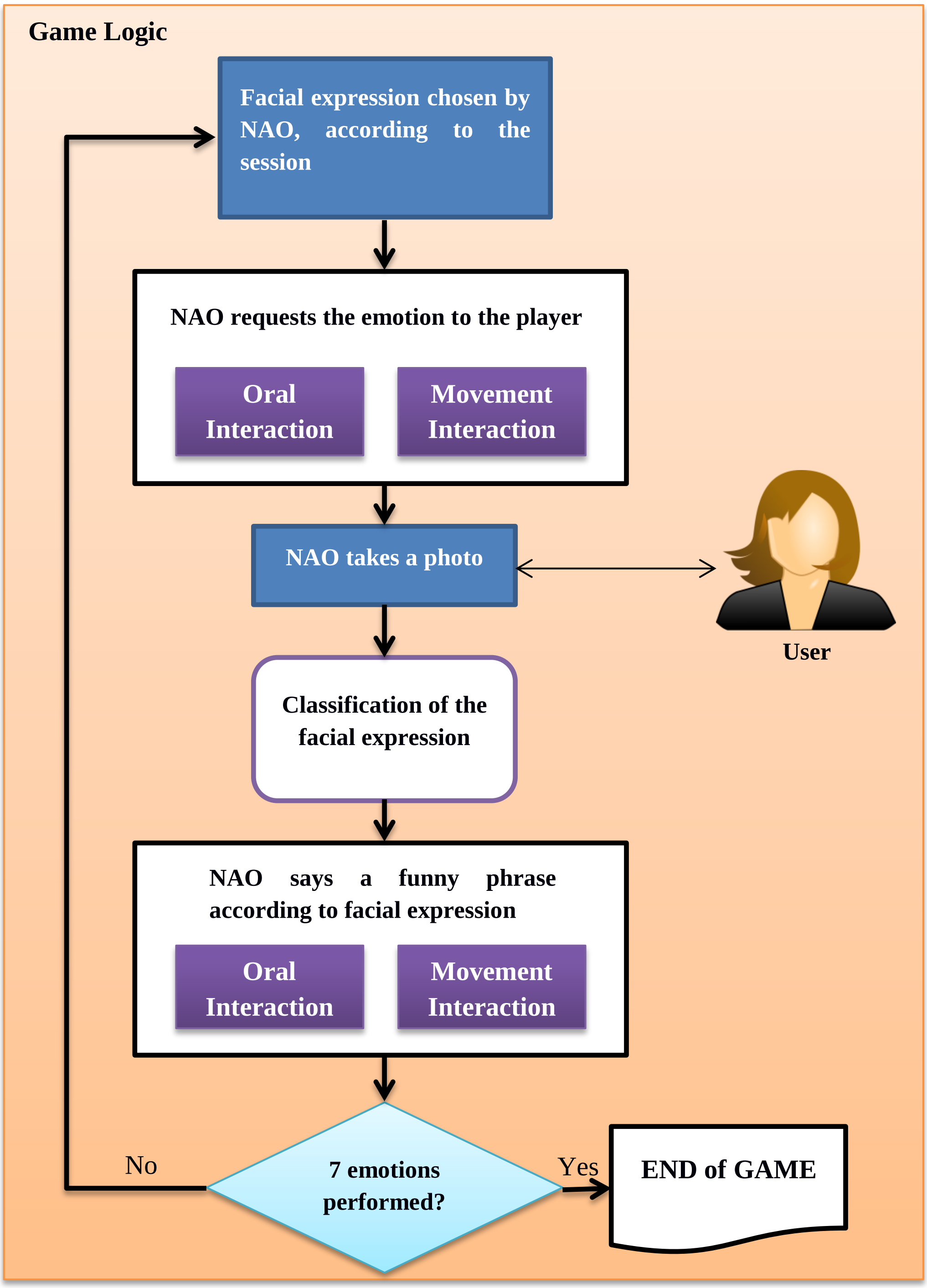

Game Logic
Facial expression chosen by NAO, according to the session
NAO requests the emotion to the player
Oral Interaction
Movement Interaction
NAO takes a photo
User
Classification of the facial expression
NAO says a funny phrase according to facial expression
Oral Interaction
Movement Interaction
No
7 emotions performed?
Yes
END of GAME

Figure 5.4. Game Logic for both sessions.

Third, the robot verifies the session in which is the game and varies its oral presentation according to the session, while makes smooth movements with its arms in order to create a simulation of reality. In this presentation, the robot explains how the experiment will be performed by the user and the game logic begins (see Figure 5.4). This logic consists of selecting an emotion of the 7 facial expressions (Anger, Disgust, Fear, Happy, Neutral, Sad and Surprise) according to the session initiated. Then, the robot begins to interact with the user, challenging the participant to make the proposed expression. The user performs the facial expression proposed by the robot and the robot takes a photo of the user. If the detection of the face is favourable, the image will be pre-processed and classified with the neural network obtained in Chapter 4. With this process, the robot is able to recognize the expression made by the user. For each recognized expression, the robot interacts with the user, trying to motivate and involve the user in the game through funny phrases. In case of not recognizing a face, the robot apologizes to the user and requests a replay of the emotion. All this process is repeated until the 7 emotions are performed, in order to finish the game correctly.

5.4 Results

In this Section, we analyze the results obtained both by the CNN and by 10 human experts in facial expression recognition, in addition to analyze the results of the questionnaire which were performed by the users at the end of the experiment. Therefore, this Section is divided into two parts. In the first part a comparison between the results obtained by the CNN and by the experts is done, in addition to analyze the difficulty to perform a particular expression by non-experts participants. Finally, in the second part, the results of the questionnaires are analyzed.

5.4.1 Comparison between CNN and Human experts

In this sub-Section, the results obtained by our CNN are analyzed together with the results obtained by 10 experts. 182 images of the first session and 175 images of the second session were analyzed. If one of the sessions could not be performed due to user unavailability, no value is shown in the table (see Tables 5.1, 5.2, 5.3 and 5.4). Because the neural network has been trained with five datasets (CK+, BU4DFE, JAFFE, WSEFEP and FEGA) and two of which do not contain the neutral face, we will show separately the results for both 6 and 7 expressions. When analyzing the results obtained in the case of 7 expressions, we shall take into account that the neutral face expression is under-represented in the training set.

5.4.1.1 Results using six expressions

In Table 5.1, we show the results of each participant in the first session, obtained both by the CNN and by the experts (E1, E2, E3, E4, E5, E6, E7, E8, E9 and E10). In Table 5.2, we show the results of each participant in the second session. In both tables (5.1 and 5.2), the 6 basic facial expressions are analyzed. As we see in both tables, we obtain competitive results using our CNN trained for 6 expressions (without neutral face).

One reason why the experts get better results is that the human capacity in facial expression recognition is more trained by the acquired experience of all their life. When they classified a facial expression and were not sure, they tried to remember what expression had not classified.

Participants (Session 1)	**CNN**	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**	**E9**	**E10**
User 1	83%	100%	83%	100%	83%	100%	67%	67%	100%	50%	67%
User 2	83%	83%	67%	100%	83%	83%	83%	83%	83%	83%	83%
User 3	67%	67%	100%	33%	100%	100%	100%	100%	67%	67%	67%
User 4	40%	60%	40%	60%	40%	60%	60%	60%	60%	60%	60%
User 5	67%	67%	67%	83%	83%	67%	67%	67%	83%	67%	67%
User 6	75%	50%	50%	100%	50%	50%	50%	50%	50%	50%	50%
User 7	83%	83%	83%	83%	83%	83%	83%	67%	83%	83%	83%
User 8	50%	67%	50%	67%	50%	50%	67%	33%	33%	67%	67%
User 9	83%	83%	83%	83%	83%	67%	83%	100%	83%	67%	83%
User 10	--	--	--	--	--	--	--	--	--	--	--
User 11	75%	100%	100%	100%	100%	100%	75%	100%	100%	100%	100%
User 12	33%	67%	67%	67%	67%	67%	67%	33%	50%	67%	33%
User 13	60%	100%	80%	80%	100%	60%	100%	60%	100%	80%	100%
User 14	60%	60%	60%	60%	60%	60%	80%	60%	60%	20%	60%
User 15	83%	67%	83%	83%	67%	67%	83%	50%	50%	67%	67%
User 16	50%	67%	67%	83%	83%	67%	83%	33%	83%	67%	67%
User 17	75%	25%	25%	50%	50%	50%	50%	25%	25%	25%	100%
User 18	60%	20%	40%	40%	40%	40%	60%	80%	60%	40%	60%
User 19	33%	67%	83%	33%	67%	83%	50%	50%	67%	67%	67%
User 20	--	--	--	--	--	--	--	--	--	--	--
User 21	50%	83%	83%	83%	83%	83%	83%	83%	83%	100%	100%
User 22	80%	50%	0%	75%	50%	0%	75%	50%	75%	50%	50%
User 23	67%	67%	50%	33%	50%	50%	50%	33%	33%	50%	83%
User 24	67%	80%	80%	100%	100%	80%	80%	100%	80%	80%	100%
User 25	67%	83%	67%	83%	100%	83%	67%	83%	83%	100%	100%
User 26	60%	75%	75%	75%	100%	100%	100%	75%	100%	100%	100%
User 27	60%	40%	40%	60%	80%	40%	40%	40%	20%	40%	80%
User 28	67%	83%	83%	67%	100%	83%	100%	83%	83%	100%	100%
User 29	--	--	--	--	--	--	--	--	--	--	--
Average	**64.6%**	69%	65.6%	72.4%	75.1%	68.2%	73.2%	**64.1%**	69.1%	67.1%	**76.7%**

Table 5.1. Comparison between CNN and Human experts for the 6 basic facial expressions. An empty value in the table corresponds to users who could not perform session 1 and only performed session 2. We show the average of the results obtained by the CNN and by the best and worst experts in bold text.

Therefore, we tried to avoid this discarding by telling the experts that if they thought that two expressions were similar, they should label them with the same emotion. In spite of this, the average classification accuracy obtained by our trained neural network is higher than for some expert in both tables (5.1 and 5.2). The best result in the first session is for expert E10, which obtained 12.6% more accuracy than our CNN. However, the best result in the second session is for expert E3, which obtained 10.1% more accuracy than the CNN. Nonetheless, the results obtained with our proposed CNN are competitive, with respect to other networks proposed in the literature using cross-datasets, since this experiment have allowed to collect a set of new images.

Participants (Session 2)	**CNN**	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**	**E9**	**E10**
User 1	83%	83%	67%	100%	83%	83%	83%	83%	83%	83%	67%
User 2	67%	67%	83%	100%	67%	83%	83%	33%	67%	67%	67%
User 3	100%	100%	100%	100%	83%	83%	100%	100%	67%	100%	100%
User 4	50%	50%	67%	33%	50%	50%	50%	33%	50%	50%	33%
User 5	67%	67%	67%	67%	67%	67%	67%	50%	67%	83%	67%
User 6	67%	67%	67%	83%	83%	100%	100%	100%	83%	83%	100%
User 7	67%	83%	83%	100%	83%	100%	100%	83%	83%	83%	100%
User 8	67%	33%	33%	33%	33%	33%	33%	33%	33%	33%	33%
User 9	67%	67%	50%	83%	67%	50%	67%	50%	50%	67%	50%
User 10	50%	50%	50%	67%	50%	33%	50%	67%	67%	17%	83%
User 11	--	--	--	--	--	--	--	--	--	--	--
User 12	67%	80%	60%	60%	80%	80%	60%	80%	80%	60%	60%
User 13	50%	100%	100%	83%	67%	83%	100%	100%	67%	67%	100%
User 14	--	--	--	--	--	--	--	--	--	--	--
User 15	67%	33%	50%	50%	50%	67%	50%	50%	50%	33%	50%
User 16	33%	83%	83%	83%	100%	67%	83%	33%	67%	83%	100%
User 17	--	--	--	--	--	--	--	--	--	--	--
User 18	--	--	--	--	--	--	--	--	--	--	--
User 19	75%	100%	100%	100%	100%	100%	75%	100%	100%	75%	100%
User 20	60%	17%	33%	33%	17%	17%	17%	17%	33%	17%	17%
User 21	67%	67%	67%	67%	67%	67%	67%	67%	67%	83%	67%
User 22	50%	60%	100%	80%	120%	100%	60%	100%	100%	100%	100%
User 23	67%	67%	67%	67%	83%	83%	67%	67%	67%	67%	83%
User 24	50%	67%	50%	67%	67%	67%	50%	17%	33%	50%	67%
User 25	83%	100%	100%	100%	83%	100%	100%	100%	83%	83%	100%
User 26	83%	83%	83%	83%	83%	83%	83%	83%	83%	83%	83%
User 27	33%	33%	33%	67%	33%	33%	33%	33%	33%	33%	33%
User 28	67%	83%	83%	83%	67%	67%	83%	33%	83%	67%	67%
User 29	80%	80%	80%	100%	80%	80%	80%	80%	80%	100%	60%
Average	**65.5%**	68.8%	70.3%	**75.6%**	70.5%	71.1%	69.7%	**63.7%**	67.1%	66.7%	71.5%

Table 5.2. Comparison between CNN and Human experts for the 6 basic facial expressions. An empty value in the table corresponds to users who could not perform session 2 and only performed session 1. We show the average of the results obtained by CNN and by the best and worst experts in bold text.

5.4.1.2 Results using seven expressions

In Tables 5.3 and 5.4, the 7 facial expressions are analyzed. In Table 5.3, we show the results of each participant in the first session and in Table 5.4, we show their results in the second session, both by CNN and by experts. As we can see in both tables, we obtain worse results using our CNN trained for 7 expressions than with the expert evaluation. For this reason we compare the results in detail (see Tables 5.5 and 5.6). Both in Table 5.5 and 5.6, we show the results obtained for each facial expression by each expert and by our CNN trained with 7 expressions. In Table 5.5 we show the results of session 1 and in Table 5.6, the results of session 2. The two last rows of these tables show the results between the average of the experts and our CNN. In these last rows of Table 5.5, we can observe that the CNN overcomes the experts in some facial expression such as Happy (HA) and Angry (AN). But, Surprise (SU), Sad (SA) and Disgust (DI) are better recognized by humans. Instead, Fear (FE) is difficult to recognize both by humans and by CNN. The main difference in this first session is the Neutral (NE) face. The experts recognize the Neutral face with 68% more accuracy than the CNN.

Participants (Session1)	**CNN**	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**	**E9**	**E10**
User 1	71%	100%	86%	100%	86%	100%	71%	71%	100%	57%	71%
User 2	86%	86%	71%	100%	86%	86%	86%	86%	86%	86%	86%
User 3	75%	75%	100%	50%	100%	100%	100%	100%	75%	75%	75%
User 4	33%	67%	50%	67%	50%	67%	50%	67%	67%	67%	50%
User 5	67%	67%	67%	83%	83%	67%	67%	67%	83%	67%	67%
User 6	75%	50%	50%	100%	50%	50%	50%	50%	50%	50%	50%
User 7	83%	83%	83%	83%	83%	83%	83%	67%	83%	83%	83%
User 8	43%	71%	43%	71%	57%	57%	57%	29%	43%	71%	71%
User 9	71%	86%	86%	86%	86%	71%	71%	100%	86%	71%	86%
User 10	--	--	--	--	--	--	--	--	--	--	--
User 11	60%	80%	100%	100%	80%	100%	60%	80%	80%	100%	100%
User 12	43%	71%	57%	71%	71%	71%	57%	43%	57%	71%	43%
User 13	50%	100%	83%	83%	100%	50%	100%	50%	83%	83%	83%
User 14	50%	67%	50%	67%	67%	67%	83%	50%	67%	33%	67%
User 15	71%	71%	86%	86%	71%	71%	86%	57%	57%	71%	71%
User 16	43%	71%	71%	86%	86%	71%	86%	43%	86%	71%	71%
User 17	60%	40%	40%	60%	60%	60%	60%	40%	40%	40%	100%
User 18	50%	17%	33%	50%	50%	50%	67%	67%	67%	50%	67%
User 19	33%	67%	83%	33%	67%	83%	50%	50%	67%	67%	67%
User 20	--	--	--	--	--	--	--	--	--	--	--
User 21	43%	71%	71%	86%	86%	86%	86%	71%	86%	100%	100%
User 22	67%	60%	20%	60%	60%	0%	80%	60%	80%	60%	60%
User 23	57%	71%	57%	43%	57%	57%	57%	43%	43%	57%	86%
User 24	57%	83%	83%	100%	100%	83%	67%	100%	83%	83%	100%
User 25	57%	86%	71%	86%	100%	86%	71%	86%	86%	100%	100%
User 26	67%	80%	80%	80%	100%	100%	100%	80%	100%	100%	100%
User 27	60%	40%	40%	60%	80%	40%	40%	40%	20%	40%	80%
User 28	57%	86%	86%	71%	100%	86%	100%	86%	86%	100%	100%
User 29	--	--	--	--	--	--	--	--	--	--	--
Average	**58.9%**	71.0%	67.3%	75.5%	77.5%	70.9%	72.5%	**64.7%**	71.5%	71.4%	**78.2%**

Table 5.3. Comparison between CNN and Human experts for the 7 facial expressions. An empty value in the table corresponds to users who could not perform session 1 and only performed session 2. We show the average of the results obtained by CNN and by the best and worst experts in bold text.

This occurs because many participants have thick beard, and our CNN is trained mainly with men with small beard or without any beard. Another cause is if the participant is very tall, since the NAO Robot is a small humanoid robot. If the person is very tall; the perspective of the image is distorted, which hinders the recognition of emotions. Therefore, our CNN confused most of the neutral faces with angry faces. Nevertheless the neutral face of women was recognized better by our CNN, although sometimes confused the Neutral face with an expression of anger or sadness. This problem can also be because in our training set of Chapter 4, there are less neutral faces because the CK+ and BU4DFE datasets do not contain the neutral face. And these two datasets are the largest of the five datasets which we use (described in Chapter 4). In Table 5.6, we can see a similar situation to Table 5.5. In this case the CNN overcomes the experts in recognition of facial expressions such as Surprise and Angry, although, Happy, Sad and Disgust are better recognized by humans. Like in Table 5.5, the main difference is in the Neutral face, which is better recognized by experts. Although our CNN confuses the neutral face with the angry face, and this makes our average accuracy decrease about a 12% with respect to the experts. We can affirm that our CNN is mostly competitive, insomuch as this experiment is performed by non-expert participants in real time and it can be considered as a cross-validation experiment. Therefore, we can conclude that our CNN is close to the human perception, especially for the 6 basic expressions.

Participants (Session 2)	**CNN**	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**	**E8**	**E9**	**E10**
User 1	71%	100%	71%	100%	86%	86%	86%	86%	86%	86%	71%
User 2	57%	57%	86%	100%	71%	86%	71%	43%	71%	71%	71%
User 3	100%	100%	100%	100%	86%	86%	100%	100%	71%	100%	100%
User 4	43%	43%	57%	43%	57%	57%	43%	43%	43%	57%	43%
User 5	57%	100%	57%	71%	71%	71%	71%	57%	71%	86%	71%
User 6	57%	86%	71%	86%	86%	100%	100%	86%	71%	86%	100%
User 7	67%	100%	71%	100%	86%	100%	86%	86%	86%	86%	100%
User 8	67%	33%	33%	33%	33%	33%	33%	33%	33%	33%	33%
User 9	57%	17%	67%	100%	83%	67%	83%	67%	67%	83%	67%
User 10	43%	57%	43%	71%	57%	43%	57%	71%	71%	29%	86%
User 11	--	--	--	--	--	--	--	--	--	--	--
User 12	67%	100%	50%	67%	83%	83%	67%	67%	67%	67%	67%
User 13	43%	86%	86%	86%	71%	86%	100%	100%	71%	71%	100%
User 14	--	--	--	--	--	--	--	--	--	--	--
User 15	71%	57%	57%	57%	57%	71%	57%	57%	57%	43%	57%
User 16	29%	57%	86%	86%	100%	71%	86%	43%	71%	86%	100%
User 17	--	--	--	--	--	--	--	--	--	--	--
User 18	--	--	--	--	--	--	--	--	--	--	--
User 19	60%	80%	80%	80%	80%	80%	60%	80%	80%	60%	80%
User 20	50%	50%	33%	50%	33%	33%	33%	33%	50%	33%	33%
User 21	57%	71%	57%	71%	71%	71%	71%	71%	71%	86%	71%
User 22	50%	67%	100%	67%	100%	100%	67%	100%	100%	100%	100%
User 23	57%	57%	57%	71%	86%	86%	71%	71%	71%	71%	86%
User 24	57%	71%	57%	71%	71%	71%	57%	29%	43%	57%	71%
User 25	71%	100%	86%	100%	86%	100%	100%	100%	86%	86%	100%
User 26	86%	86%	86%	86%	86%	86%	86%	86%	86%	86%	86%
User 27	33%	50%	43%	71%	43%	50%	50%	50%	50%	43%	43%
User 28	57%	43%	71%	86%	71%	71%	71%	43%	86%	71%	71%
User 29	67%	67%	67%	100%	67%	67%	67%	67%	67%	83%	50%
Average	**59,0%**	69.4%	66.9%	**78.2%**	73.6%	74.2%	71.0%	**66.7%**	69.1%	70.4%	74.3%

Table 5.4. Comparison between CNN and Human experts for the 7 facial expressions. An empty value in the table corresponds to users who could not perform session 2 and only performed session 1. We show the average of the results obtained by CNN and by the best and worst experts in bold text.

Another question that arose during the experiment was the difficulty, for each participant, of representing the different emotions, since most of them doubted in some expression. This caused the bad capture of some images. For this reason we needed experts to evaluate the images so we compare and verify the results. In addition to evaluate the images by experts, we measured this difficulty of the participants to express themselves through a questionnaire, where they rated between 1 and 4 (1 the least and 4 the most difficult) the difficulty to represent each of the emotions (see Table 5.7).

Session 1	**AN**	**DI**	**FE**	**HA**	**NE**	**SA**	**SU**	**Mean**
E1	76%	95%	24%	88%	86%	58%	76%	71.0%
E2	72%	79%	14%	92%	76%	54%	80%	67.3%
E3	68%	84%	33%	92%	95%	71%	84%	75.5%
E4	72%	84%	33%	92%	95%	75%	88%	77.5%
E5	64%	79%	24%	83%	90%	79%	76%	70.9%
E6	72%	74%	33%	88%	71%	83%	80%	72.5%
E7	48%	68%	19%	96%	71%	54%	88%	64.7%
E8	72%	74%	38%	88%	90%	67%	72%	71.5%
E9	60%	89%	29%	88%	100%	63%	76%	71.4%
E10	76%	79%	43%	92%	90%	79%	84%	78.2%
Mean Experts	68%	81%	29%	90%	87%	68%	80%	72%
CNN	76%	48%	29%	100%	19%	50%	72%	58.9%

Table 5.5. Accuracy rate of each facial expression, in the first session, by the 10 experts and by our CNN, in addition to their mean. In two last files we show the main differences.

Session 2	**AN**	**DI**	**FE**	**HA**	**NE**	**SA**	**SU**	**Mean**
E1	63%	63%	25%	92%	90%	76%	83%	69.4%
E2	63%	79%	13%	100%	81%	88%	48%	66.9%
E3	79%	79%	29%	92%	95%	88%	91%	78.2%
E4	50%	83%	25%	100%	81%	84%	96%	73.6%
E5	54%	75%	38%	96%	86%	80%	96%	74.2%
E6	46%	71%	46%	100%	76%	84%	78%	71.0%
E7	50%	58%	29%	96%	71%	76%	87%	66.7%
E8	46%	75%	21%	96%	100%	68%	83%	69.1%
E9	46%	88%	29%	88%	86%	68%	96%	70.4%
E10	54%	75%	33%	96%	90%	84%	96%	74.3%
Mean_Experts	55%	75%	29%	96%	86%	80%	85%	71.3%
CNN	63%	63%	25%	92%	14%	56%	92%	59%

Table 5.6. Accuracy rate of each facial expression, in the second session, by the 10 experts and by our CNN, in addition to their mean. In two last files we show the main differences.

Dificulty to express emotions	**AN**	**DI**	**FE**	**HA**	**NE**	**SA**	**SU**
Mean_Participants	1,34	2,07	3,07	1,10	1,45	2,00	1,69
Mean_recognition accuracy Experts	62%	78%	29%	93%	86%	74%	83%
Mean_recognition accuracy CNN	70%	56%	27%	96%	17%	53%	82%

Table 5.7. Comparison between the mean of experts, CNN and the opinions of the participants about the difficulty to express emotions. These means were calculated from two sessions performed for each facial expression.

In Table 5.7, we display the mean recognition accuracies of the two sessions obtained both by experts and the CNN, and compared them with the average difficulty ratings in interpreting each emotion. We observe that the facial expressions more difficult to express by the participants are disgust, sad and fear, which obtain a score equal or greater than 2. These results correlate with the recognition accuracy results obtained by the CNN, which are the lowest. Although for the CNN the worst result is with the neutral expression because if the person have thick beard or it is very tall, it is difficult to recognize. On the other hand the easiest facial expressions to interpret, according to the participants, are angry, happy, neutral and surprise, which obtain a score lower than 2. These results coincide with better recognition accuracies in both cases (CNN and experts), except in the angry face expression in the case of the human experts, because, in case of doubt they always chose the neutral expression. This explains why the results of the neutral face in the evaluation by experts were high. In Figures 5.5 and 5.6, we show two extreme cases in the representation of facial expressions by the participants.

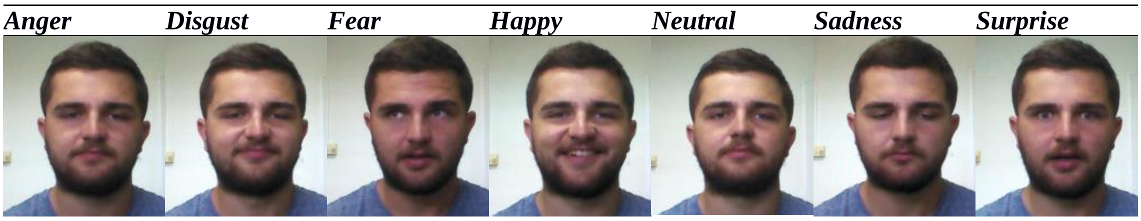


Figure 5.5. Interpretation of the 7 expressions (49% recognized by experts, 43% recognized by CNN).

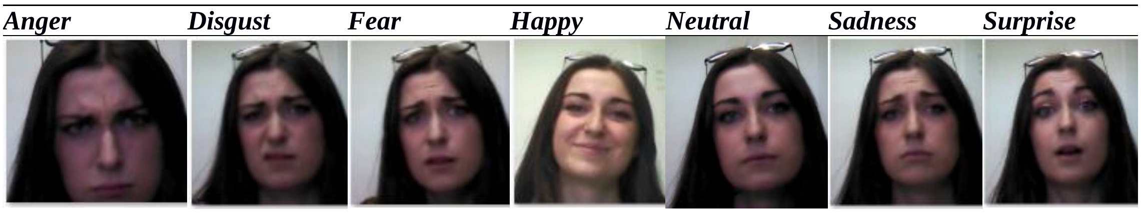


Figure 5.6. Interpretation of the 7 expressions (94% recognized by experts, 100% recognized by CNN).

In Figure 5.6 the experts recognized a mean of 94% of their expressions (7 experts recognized 100% of the expressions, 2 experts recognized 86% of them and 1 expert recognized 71% of them). And the CNN recognized all of them (100%), coinciding with the majority of experts. In Figure 5.5 the experts only recognize a mean of 49% of the expressions (6 experts recognized 43% of them and 4 experts recognized only 57% of them). Finally, the CNN recognized 43% of the emotions, coinciding with the majority of experts. In both figures the order of the expressions is the same. As we can see, performing this type of experiment with non-actor participants leads to interpretation difficulties both for neural networks and humans.

5.4.1.3 Results of the questionnaire

Finally, participants were surveyed at the end of the experiment. Most of the users (93%) did not need any help. They were guided only by the robot's instructions. Only 7% of the participants asked the interlocutor some questions. Table 5.8 shows the averages obtained both in the level of amusement and interaction experienced by the participants as well as their level of attention in the game. This measure was evaluated between 1 and 4 (1 for the lowest and 4 for the highest). These results show that the participants of this experiment obtained a quite satisfactory experience (see Table 5.8).

Participants	Amusement	Attention	Interaction
User 1	4	4	4
User 2	4	4	4
User 3	4	3	4
User 4	3	4	4
User 5	3	2	3
User 6	4	3	4
User 7	4	4	4
User 8	3	3	4
User 9	3	3	4
User 10	4	4	4
User 11	4	4	4
User 12	3	3	4
User 13	3	4	4
User 14	3	3	4
User 15	2	4	4
User 16	4	4	4
User 17	4	4	4
User 18	3	3	4
User 19	4	4	4
User 20	2	3	2
User 21	3	4	4
User 22	3	3	4
User 23	2	3	4
User 24	4	4	4
User 25	3	3	4
User 26	4	4	4
User 27	3	3	2
User 28	4	4	4
User 29	4	4	4
Average	3.38	3.52	3.83

Table 5.8. Results obtained in the questionnaire performed by the participants.

Among the comments left by the participants, we highlighted that they liked the experience of being able to interact with a social robot, that the robot was able to recognize their facial expressions and be able to evaluate their capacity as an actor or actress. The funny dialogues that the robot had according to the recognized expression and the harmonious movements that the NAO robot performed when interacting with the user, resulted in a satisfactory user experience.

5.5 Conclusions

An advanced interaction system based on a social robot which allows users to replicate and learn in a playful way the basic facial expressions. The Convolutional Neural Network (CNN) from Chapter 4 has been implemented in the application of the robot. This system has been validated in 29 non-expert users. We have shown that our CNN is mostly competitive, taking into account that this experiment is performed by non-expert participants in real time and can be considered a cross-validation experiment. The results show that our CNN is close to human perception, especially for the 6 basic expressions. However, our system fails in the neutral expression recognition, because many participants have thick beards, and our CNN is mainly trained with men with little beard or no beard. Another cause was if the participant was very tall, the perspective of the image was distorted, since the NAO robot is a small humanoid robot. Therefore, our CNN confused most of the neutral faces with angry faces. In future work, we try to resolve this question applying more information in the training set or improving the pre-process step in order to recognize better this kind of images. On the other hand, a study has been performed in order to determine the level of fun, interaction and attention that the participants experienced in the game. These results show that the participants of this experiment obtained a quite satisfactory experience. As future work, it is planned to perform this same experiment with several sessions, especially for children with attention deficit disorder (ADD).

Chapter 6

Conclusion and Future Work

In this work a system for facial detection and expression recognition has been developed and validated by non-expert users in a real environment using a social robot. This system is totally modular and it can be adapted to other applications or research fields. This work consists of three parts: face detection, facial expression recognition and an application in a social robot in order to validate the system and measure some parameters of interaction.

1. *Face detection.* We have shown that it is possible to successfully use the *a contrario* model to improve the performance of the classical Viola–Jones face detector. We have justified that a Gaussian distribution can be used as a background model against which to test the existence of faces in an image, and then we have proposed a method to adapt the detection threshold of a single strong classifier to control the number of false positives. We validate this method with three frontal face datasets (IMM, BioID, FDDB). However, since the use of a single classifier is not efficient in terms of computation time, we couple the adaptive threshold principle with the cascade concept to build a very short cascade (just 4 stages) that improves the results of much larger cascades, in terms of detection rates and computation times. Our method reduces the computation time by a factor near to five. The obtained results are promising and suggest that the same principles might be applied to more recent face detectors.

2. *Facial expression recognition.* We have shown that the use of a pre-processing step is relevant to improve the performance of a convolutional neural network (CNN), despite its intrinsic complexity, and that the results improve significantly when information captured with different cameras is combined. We have also shown that the facial expression classification from non-expert humans is correlated with the one of the CNN (especially in the recognition of Angry, Disgust and Fear), that is, we can see that the same types of facial expressions are misclassified by both the humans and the neural network. Several experiments have been performed to build our proposed CNN and find the adequate preprocessing steps. We have evaluated the system using six datasets, including two new datasets (FEGA and FE-Test) that have been created specifically for this thesis. One of the captured datasets (FEGA) is the first one in the literature including simultaneously

labeling of facial expression, gender and age of the individuals. Another contribution is the combination of different datasets to train our system. Up to our knowledge, this is the most extensive experimental study to date in cross-dataset facial expression recognition using CNNs, since most previous studies in the literature only employ one dataset for testing. Our study shows that each dataset adds an important value in the training, because each one of them has been captured in different conditions, and contains people from different ethnicities and ages. Therefore, not only the quantity is important to train the data with CNN, but also the variety of information. Thus, the combination of these datasets into one single training dataset, using our image preprocessing steps to unify them, improves significantly the results with respect to using only one dataset for training. Furthermore we have got about 70% in accuracy using the cross-datasets protocol when the test set comes from a never-seen-before dataset. Finally, we have performed a comparative study of facial expression classification using our system vs. human opinion. We have observed that humans and machine are prone to similar misclassifications errors.

3. *An application on a Social Robot to validate the system.* A system for capturing and classifying expressions based on a social robot and a multimodal interaction has been presented. This system has been validated with 29 non-expert users. The system uses the CNN network described in the previous paragraph for the recognition of facial expression in the application with the social robot. It has been shown that our CNN is competitive with the state of the art, taking into account that this experiment is carried out by non-expert participants in real time and can be considered as a cross-validation experiment. Therefore, we can conclude that our CNN is close to human perception, especially for the 6 basic expressions. On the other hand, a study has been carried out in order to determine the level of amusement, interaction and attention that the participants experienced in the interaction with the robot. These results show that the participants of this experiment obtained a quite satisfactory experience.

Each one of these parts has contributed to the achievement of the thesis' goals and has generated new lines of research.

As future work, and in the field of face detection, we propose to explore the use of integral channel features trained using faces in various poses/views and the application of the threshold adaptation technique of Jain and Leamed-Miller [35] to improve the detection rates.

In the field of facial expression recognition, we intend to refine our system with more datasets, in addition to studying the pre-processing step for color images. We also plan to make experiments with the FER2013 Dataset which is used in many papers. This dataset consists of 35.887 grayscale face images with 7 emotions labelled. Initially, we used this dataset, but it gave similar results both in the training and test. Generally, in the test gets a lower accuracy than in the training. Therefore, we revised the dataset in detail. We discovered that in the training set and in the testing set more than 5000 images were duplicated. As future work we pretend to present a clean version of this dataset which can be a good contribution in this field. We also plan to extend this study using age and ethnicity to develop a new multimodal system, more robust, for facial expression recognition.

Finally, in the validation step using a social robot, we have encountered problems with the identification of neutral faces. This suggests that we need to use more information of neutral faces in the training set or to improve the pre-processing step in order to recognize better this kind of images. On the other hand, the results show that the participants had a satisfactory experience and we plan to perform this same experiment with several sessions, especially for children with attention deficit disorder (ADD).

Chapter 7

Bibliography

[1] Ahonen, T., Hadid, A., & Pietikäinen, M. (2004). Face recognition with local binary patterns. In *European conference on computer vision* (pp. 469-481). Springer, Berlin, Heidelberg. https://doi.org/10.1007/ 978-3-540-24670-1 36.

[2] Asadoorian, M. O., & Kantarelis, D. (2005). *Essentials of inferential statistics*. University Press of America.

[3] Ash, R. B., Robert, B., Doleans-Dade, C. A., & Catherine, A. (2000). *Probability and measure theory*. Academic Press.

[4] *Armon-Jones, C.* (1986). The *social functions* of *emotion.* In *R. Harré* (*Ed.*), *The social construction of emotions* (pp. *57*-82). *Oxford*, England: *Basil Blackwell*.

[5] BioID Face Database-FaceDB. https://www.bioid.com/About/BioID-Face-Database, last visit: 25/05/2019.

[6] Breazeal, C., Dautenhahn, K., & Kanda, T. (2016). Social robotics. In *Springer handbook of robotics* (pp. 1935-1972). Springer, Cham.

[7] Burkert, P., Trier, F., Afzal, M. Z., Dengel, A., & Liwicki, M. (2015). Dexpression: Deep convolutional neural network for expression recognition. *arXiv preprint arXiv:1509.05371*.

[8] Calvo, M. G., & Nummenmaa, L. (2016). Perceptual and affective mechanisms in facial expression recognition: An integrative review. *Cognition and Emotion, 30*(6), 1081-1106. doi: 10.1080/02699931.2015.1049124.

[9] Calvo-Varela, L., Regueiro, C. V., Canzobre, D. S., & Iglesias, R. (2016). Development of a Nao humanoid robot able to play Tic-Tac-Toe game on a tactile tablet. In *Robot 2015: Second Iberian Robotics Conference* (pp. 203-215). Springer, Cham. DOI: 10.1007/978-3-319-27146-0_16.

[10] Carroll, J. M., & Kjeldskov, J. (2013). The encyclopedia of human-computer interaction. *2nd. Ed. Interaction Design Foundation*.

[11] Chen, L., Zhou, M., Su, W., Wu, M., She, J., & Hirota, K. (2018). Softmax regression

based deep sparse autoencoder network for facial emotion recognition in human-robot interaction.*Information Sciences*, *428*, 49-61.

[12] Chen, D., Ren, S., Wei, Y., Cao, X., & Sun, J. (2014). Joint cascade face detection and alignment. In *European Conference on Computer Vision* (pp. 109-122). Springer, Cham. https://doi.org/10.1007/978-3-319-599-4 8.

[13] Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In Proceedings of the *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (pp. 886–893).

[14] Daniş, F. S., Meriçli, T., Meriçli, C., & Akın, H. L. (2010). Robot detection with a cascade of boosted classifiers based on haar-like features. In *Robot Soccer World Cup* (pp. 409-417). Springer, Berlin, Heidelberg.

[15] Delon, J., Desolneux, A., Lisani, J. L., & Petro, A. B. (2007). A nonparametric approach for histogram segmentation. *IEEE Transactions on Image Processing*, *16*(1), 253-261.

[16] Desolneux, A., Moisan, L., & Morel, J. M. (2001). Edge detection by Helmholtz principle. *Journal of mathematical imaging and vision*, *14*(3), 271-284.

[17] Desolneux, A., Moisan, L., & More, J. M. (2003). A grouping principle and four applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *25*(4), 508-513.

[18] Desolneux, A., Moisan, L., & Morel, J. M. (2003). Maximal meaningful events and applications to image analysis. *The Annals of Statistics*, *31*(6), 1822-1851.

[19] Desolneux, A., Moisan, L., & Morel, J. M. (2008). *From gestalt theory to image analysis: a probabilistic approach* (Vol. 34). Springer, New York.

[20] Ekman, P. (1977). Facial Expression. In Siegman, A. & Feldstein, S. (Eds.), *Nonverbal Communication and Behavior* (pp. 97-126). New Jersey: Lawrence Erlbaum Association.

[21] Fogg, B. J. (1999). Persuasive technologies. *Communications of the ACM*, *42*(5), 27-29.

[22] Fong, T., Nourbakhsh, I., & Dautenhahn, K. (2003). A survey of socially interactive robots. *Robotics and autonomous systems*, *42*(3-4), 143-166.

[23] Fröba, B., & Ernst, A. (2003). Fast frontal-view face detection using a multi-path decision tree. In *International Conference on Audio-and Video-Based Biometric Person Authentication* (pp. 921-928). Springer, Berlin, Heidelberg.

[24] Fujimoto, I., Matsumoto, T., De Silva, P. R. S., Kobayashi, M., & Higashi, M. (2010). Study on an assistive robot for improving imitation skill of children with autism. In *International Conference on Social Robotics* (pp. 232-242). Springer, Berlin, Heidelberg.

[25] Furrer, F., Burri, M., Achtelik, M., & Siegwart, R. (2016). RotorS—A modular Gazebo MAV simulator framework. In *Robot Operating System (ROS)* (pp. 595-625). Springer, Cham.

[26] Giryes, R., Sapiro, G., & Bronstein, A. M. (2016). Deep neural networks with random

gaussian weights: A universal classification strategy?. *IEEE Transactions on Signal Processing*, *64*(13), 3444-3457. doi: 10.1109/TSP.2016.2546221.

[27] Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics* (pp. 315-323).

[28] Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics* (pp. 249-256).

[29] Happy, S. L., & Routray, A. (2014). Automatic facial expression recognition using features of salient facial patches. *IEEE transactions on Affective Computing*, *6*(1), 1-12.

[30] Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, *18*(7), 1527-1554.

[31] Huang, W. (2015). When HCI Meets HRI: the intersection and distinction.

[32] Ismail, L., Shamsuddin, S., Yussof, H., Hashim, H., Bahari, S., Jaafar, A., & Zahari, I. (2011). Face detection technique of Humanoid Robot NAO for application in robotic assistive therapy. In *2011 IEEE International Conference on Control System, Computing and Engineering* (pp. 517-521). IEEE.

[33] Jack, R. E., Caldara, R., & Schyns, P. G. (2012). Internal representations reveal cultural diversity in expectations of facial expressions of emotion. *Journal of Experimental Psychology: General*, *141*(1), 19.

[34] Jain, V., & Learned-Miller, E. (2010). FDDB: A benchmark for face detection in unconstrained settings. *Tech. Report UM-CS-2010-009*. University of Massachusetts, Amherst. http://vis-www. cs.umass.edu/fddb/.

[35] Jain, V., & Learned-Miller, E. (2011). Online domain adaptation of a pre-trained cascade of classifiers. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 577-584). https://doi.org/10.1109/CVPR.2011.5995317.

[36] Jain, D. K., Shamsolmoali, P., & Sehdev, P. (2019). Extended deep neural network for facial emotion recognition. *Pattern Recognition Letters*, *120*, 69-74. ISSN 0167-8655.

[37] Jain, N., Kumar, S., Kumar, A., Shamsolmoali, P., & Zareapoor, M. (2018). Hybrid deep neural networks for face emotion recognition. *Pattern Recognition Letters*, *115*, 101-106. ISSN 0167-8655.

[38] James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning with Applications in R. Springer, New York. eBook ISBN 978-1-4614-7138-7.

[39] Jones, M., & Viola, P. (2003). Fast multi-view face detection. In proceedings of Computer Vision and Pattern Recognition.

[40] Jun, B., Choi, I., & Kim, D. (2012). Local transform features and hybridization for accurate face and human detection. *IEEE transactions on pattern analysis and machine intelligence*, *35*(6), 1423-1436. https://doi.org/10. 1109/TPAMI.2012.219.

[41] Jung, M., & Hinds, P. (2018). Robots in the wild: A time for more robust theories of human-robot interaction. *ACM Transactions on Human-Robot Interaction (THRI)*, *7*(1), 2.

[42] Kaehler, A., & Bradski, G. (2013). *Learning OpenCV: computer vision in C++ with the OpenCV library*. O'Reilly Media, Inc. http:/sourceforge.net/projects/opencvlibrary/.

[43] Kazemi, V., & Sullivan, J. (2014). One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1867-1874).

[44] Khorrami, P., Paine, T., & Huang, T. (2015). Do deep neural networks learn facial action units when doing expression recognition?. In *Proceedings of the IEEE International Conference on Computer Vision Workshops* (pp. 19-27). IEEE.

[45] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[46] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).

[47] Ko, B. (2018). A brief review of facial emotion recognition based on visual information. *sensors*, *18*(2), 401.

[48] Köstinger, M., Wohlhart, P., Roth, P. M., & Bischof, H. (2012). Robust face detection by simple means. In *DAGM 2012 CVAW workshop*.

[49] Leite, I., Castellano, G., Pereira, A., Martinho, C., & Paiva, A. (2012). Modelling empathic behaviour in a robotic game companion for children: an ethnographic study in real-world settings. In *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction* (pp. 367-374). ACM.

[50] Li, S. Z., Zhu, L., Zhang, Z., Blake, A., Zhang, H., & Shum, H. (2002). Statistical learning of multi-view face detection. In *European Conference on Computer Vision* (pp. 67-81). Springer, Berlin, Heidelberg.

[51] Li, S. Z., Zhang, Z., Shum, H. Y., & Zhang, H. (2002). FloatBoost learning for classification. In *Proceedings of the 15th International Conference on Neural Information Processing Systems* (pp. 1017-1024). MIT Press. Cambridge, MA, USA.

[52] Li, H., Lin, Z., Shen, X., Brandt, J., & Hua, G. (2015). A convolutional neural network cascade for face detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5325-5334).

[53] Li, Y., Fan, C., Li, Y., Wu, Q., & Ming, Y. (2018). Improving deep neural network with multiple parametric exponential linear units. *Neurocomputing*, *301*, 11-24. https://doi.org/10.1016/j.neucom.2018.01.084.

[54] Lienhart, R., Kuranov, A., & Pisarevsky, V. (2003). Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In *Joint Pattern Recognition Symposium* (pp. 297-304). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-45243-0 39.

[55] Lienhart, R., & Maydt, J. (2002). An extended set of haar-like features for rapid object detection. In *Proceedings, international conference on image processing* (Vol. 1, pp. 900–903). IEEE. https://doi.org/10.1109/ICIP.2002.1038171.

[56] Lindeberg, J. W. (1922). Eine neue Herleitung des Exponentialgesetzes in der Wahrscheinlichkeitsrechnung. *Mathematische Zeitschrift*, *15*(1), 211-225.

[57] Lisani, J. L., & Morel, J. M. (2003). Detection of major changes in satellite images. In *Proceedings 2003 International Conference on Image Processing (Cat. No. 03CH37429)* (Vol. 1, pp. 941–944). IEEE. https://doi.org/10.1109/ ICIP.2003.1247119.

[58] Lisani, J. L., Rudin, L., & Buades, A. (2011). Fast video search and indexing for video surveillance applications with optimally controlled False Alarm Rates. In *2011 IEEE International Conference on Multimedia and Expo* (pp. 1-6). IEEE https://doi.org/10.1109/ICME.2011.6012151.

[59] Lisani, J. L., Ramis, S., & Perales, F. J. (2017). A Contrario Detection of Faces: A Case Example. *SIAM Journal on Imaging Sciences*, *10*(4), 2091-2118.

[60] Lopes, A. T., de Aguiar, E., De Souza, A. F., & Oliveira-Santos, T. (2017). Facial expression recognition with convolutional neural networks: coping with few data and the training sample order.*Pattern Recognition*, *61*, 610-628. ISSN 0031-3203.

[61] Lowe, D. (1985). *Perceptual organization and visual recognition*. Kluwer Academic Publishers, Dordrecht.

[62] Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. (2010). The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops* (pp. 94-101). IEEE.

[63] Luo, R. C., Lin, P. H., Wu, Y. C., & Huang, C. Y. (2012). Dynamic face recognition system in recognizing facial expressions for service robotics. In *2012 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)* (pp. 879-884). IEEE.

[64] Lyons, M. J., Akamatsu, S., Kamachi, M., Gyoba, J., & Budynek, J. (1998). The Japanese female facial expression (JAFFE) database. In *Proceedings of third international conference on automatic face and gesture recognition* (pp. 14-16).

[65] Mathias, M., Benenson, R., Pedersoli, M., & Van Gool, L. (2014). Face detection without bells and whistles. In *European conference on computer vision* (pp. 720-735). Springer, Cham. https://doi.org/10.1007/978-3-319-10593-2 47.

[66] McColl, D., Hong, A., Hatakeyama, N., Nejat, G., & Benhabib, B. (2016). A survey of autonomous human affect detection methods for social robots engaged in natural HRI. *Journal of Intelligent & Robotic Systems, 82*(1), 101-133.

[67] Mollahosseini, A., Chan, D., & Mahoor, M. H. (2016). Going deeper in facial expression recognition using deep neural networks. In *2016 IEEE Winter conference on applications of computer vision (WACV)* (pp. 1-10). IEEE. doi: 10.1109/WACV.2016.7477450.

[68] Najah, G. M. S. (2017). *Emotion estimation from facial images* (Master's Thesis). Atilim University.

[69] Nao Aldebarán. http://doc.aldebaran.com/2-1/index.html, last visit: 21/04/2019.

[70] Nejat, G., & Ficocelli, M. (2008, May). Can I be of assistance? The intelligence behind an assistive robot. In *2008 IEEE International Conference on Robotics and Automation* (pp. 3564-3569). IEEE.

[71] Nguyen, D. T., Cho, S. R., Shin, K. Y., Bang, J. W., & Park, K. R. (2014). Comparative study of human age estimation with or without preclassification of gender and facial expression. *The Scientific World Journal*. http://dx.doi.org/10.1155/2014/905269.

[72] Nordstrøm, M. M., Larsen, M., Sierakowski, J., & Stegmann, M. B. (2004). The IMM face database-an annotated dataset of 240 face images. Tech. report, Informatics and Mathematical Modelling, Technical University of Denmark.

[73] Norouzi-Gheidari, N., Archambault, P. S., & Fung, J. (2012). Effects of robot-assisted therapy on stroke rehabilitation in upper limbs: systematic review and meta-analysis of the literature. *Journal of Rehabilitation Research & Development*, *49*(4).

[74] Ojala, T., Pietikäinen, M., & Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. on Pattern Analysis & Machine Intelligence*, (7), 971-987. https://doi.org/10.1109/TPAMI.2002.1017623.

[75] Olszanowski, M., Pochwatko, G., Kuklinski, K., Scibor-Rylski, M., Lewinski, P., & Ohme, R. K. (2015). Warsaw set of emotional facial expression pictures: a validation study of facial display photographs. *Frontiers in psychology*, *5*, 1516. doi:10.3389 /fpsyg.2014.01516.

[76] Owusu, E., Zhan, Y., & Mao, Q. R. (2014). A neural-AdaBoost based facial expression recognition system. *Expert Systems with Applications*, *41*(7), 3383-3390.

[77] Paulos, E., & Canny, J. (2001). Personal tele-embodiment. In *Beyond webcams* (pp. 155-167). MIT Press.

[78] Picard, R. W. (2000). *Affective computing*. MIT press.

[79] Poursaberi, A., Noubari, H. A., Gavrilova, M., & Yanushkevich, S. N. (2012). Gauss–Laguerre wavelet textural feature fusion with geometrical information for facial expression identification. *EURASIP Journal on Image and Video Processing*, *2012*(1), 17.

[80] Qin, H., Yan, J., Li, X., & Hu, X. (2016). Joint training of cascaded CNN for face detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3456-3465).

[81] Sagonas, C., Tzimiropoulos, G., Zafeiriou, S., & Pantic, M. (2013). 300 Faces in-the-wild challenge: The first facial landmark localization challenge. In *Proceedings of the IEEE International Conference on Computer Vision Workshops* (pp. 397-403).

[82] Sajjanhar, A., Wu, Z., & Wen, Q. (2018). Deep learning models for facial expression recognition. In *2018 Digital Image Computing: Techniques and Applications (DICTA)* (pp. 1-6). IEEE. doi: 10.1109/DICTA.2018.8615843.

[83] Scassellati, B., Admoni, H., & Matarić, M. (2012). Robots for use in autism research. *Annual review of biomedical engineering*, *14*, 275-294.

[84] Scherer, K. R. (2000). Psychological models of emotion. *The neuropsychology of emotion*, *137*(3), 137-162.

[85] Sebe, N., Lew, M. S., Sun, Y., Cohen, I., Gevers, T., & Huang, T. S. (2007). Authentic facial expression analysis. *Image and Vision Computing*, *25*(12), 1856-1863.

[86] Siddiqi, M. H., Ali, R., Sattar, A., Khan, A. M., & Lee, S. (2014). Depth camera-based facial expression recognition system using multilayer scheme. *IETE Technical Review*, *31*(4), 277-286.

[87] Sim, D. Y. Y., & Loo, C. K. (2015). Extensive assessment and evaluation methodologies on assistive social robots for modelling human–robot interaction–A review. *Information Sciences*, *301*, 305-344.

[88] Sochman, J., & Matas, J. (2005). Waldboost-learning for time constrained sequential detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* (Vol. 2, pp. 150-156). https://doi.org/10.1109/CVPR.2005.373.

[89] Song, I., Kim, H. J., & Jeon, P. B. (2014). Deep learning for real-time robust facial expression recognition on a smartphone. In *2014 IEEE International Conference on Consumer Electronics (ICCE)* (pp. 564-567). IEEE.

[90] Sorbello, R., Chella, A., Calí, C., Giardina, M., Nishio, S., & Ishiguro, H. (2014). Telenoid android robot as an embodied perceptual social regulation medium engaging natural human–humanoid interaction. *Robotics and Autonomous Systems*, *62*(9), 1329-1341.

[91] Tapus, A., Ţăpuş, C., & Matarić, M. J. (2008). User—robot personality matching and assistive robot behavior adaptation for post-stroke rehabilitation therapy. *Intelligent Service Robotics*, *1*(2), 169.

[92] The top 12 social companion robots. *The Medical Futurist Newsletter*, 31 July 2018. https://medicalfuturist.com/the-top-12-social-companion-robots, last visit 03/05/2019.

[93] Tielman, M., Neerincx, M., Meyer, J. J., & Looije, R. (2014). Adaptive emotional expression in robot-child interaction. In *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction* (pp. 407-414). ACM.

[94] Torta, E., Werner, F., Johnson, D. O., Juola, J. F., Cuijpers, R. H., Bazzani, M., Oberzaucher, J., Lemberger, J., Lewy, H., & Bregman, J. (2014). Evaluation of a small socially-assistive humanoid robot in intelligent homes for the care of the elderly. *Journal of Intelligent & Robotic Systems*, *76*(1), 57-71. DOI 10.1007/s10846-013-0019-0.

[95] Trujillo, L., Olague, G., Hammoud, R., & Hernandez, B. (2005). Automatic feature localization in thermal images for facial expression recognition. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)-Workshops* (pp. 14-14). IEEE.

[96] Tscherepanow, M., Hillebrand, M., Hegel, F., Wrede, B., & Kummert, F. (2009). Direct imitation of human facial expressions by a user-interface robot. In *2009 9th IEEE-RAS*

International Conference on Humanoid Robots (pp. 154-160). IEEE.

[97] Uçar, A., Demir, Y., & Güzeliş, C. (2016). A new facial expression recognition based on curvelet transform and online sequential extreme learning machine initialized with spherical clustering. *Neural Computing and Applications*, *27*(1), 131-142.

[98] Valstar, M. F., Almaev, T., Girard, J. M., McKeown, G., Mehu, M., Yin, L. & Cohn, J. F. (2015, May). Fera 2015-second facial expression recognition and analysis challenge. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)* (Vol. 6, pp. 1-8). IEEE. doi: 10.1109/FG.2015.7284874.

[99] Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *CVPR (1)*, *1*(511-518), 3. https://doi.org/10.1109/CVPR.2001.990517.

[100] Von Gioi, R. G. (2014). *A Contrario Line Segment Detection*. Springer, New York.

[101] Von Gioi, R. G., & Randall, G. (2016). Unsupervised smooth contour detection. *Image Processing On Line*, *6*, 233-267.

[102] Wang, Y. Q. (2014). An analysis of the Viola-Jones face detection algorithm. *Image Processing On Line*, *4*, 128-148. http://dx.doi.org/10.5201/ipol.2014.104.

[103] Wen, G., Hou, Z., Li, H., Li, D., Jiang, L., & Xun, E. (2017). Ensemble of deep neural networks with probability-based fusion for facial expression recognition. *Cognitive Computation*, *9*(5), 597-610.

[104] Werry, I., Dautenhahn, K., Ogden, B., & Harwin, W. (2001). Can social interaction skills be taught by a social agent? The role of a robotic mediator in autism therapy. In*International Conference on Cognitive Technology* (pp. 57-74). Springer, Berlin, Heidelberg.

[105] Wimmer, M., MacDonald, B. A., Jayamuni, D., & Yadav, A. (2008). Facial expression recognition for human-robot interaction–a prototype. In *International Workshop on Robot Vision* (pp. 139-152). Springer, Berlin, Heidelberg.

[106] Witkin, A. and Tenenbaum, J. (1983). *On the role of structure in vision, in Human and Machine Vision*. Academic Press (pp. 481–543). New York.

[107] Wu, B., Ai, H., Huang, C., & Lao, S. (2004). Fast rotation invariant multi-view face detection based on real adaboost. In *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings.* (pp. 79-84). IEEE.

[108] Xiao, R., Zhu, H., Sun, H., & Tang, X. (2007). Dynamic cascades for face detection. In *2007 IEEE 11th International Conference on Computer Vision* (pp. 1-8). IEEE.

[109] Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights into imaging*, *9*(4), 611-629.

[110] Yin, L., Wei, X., Sun, Y., Wang, J., & Rosato, M. J. (2006). A 3D facial expression database for facial behavior research. In *7th international conference on automatic face and gesture recognition (FGR06)* (pp. 211-216). IEEE.

[111] Zafeiriou, S., Zhang, C., & Zhang, Z. (2015). A survey on face detection in the wild: past,

present and future. *Computer Vision and Image Understanding*, *138*, 1-24. https://doi.org/http://dx.doi.org/ 10.1016/j.cviu.2015.03.015.

[112] Zavarez, M. V., Berriel, R. F., & Oliveira-Santos, T. (2017). Cross-database facial expression recognition based on fine-tuned deep convolutional network. In *2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)* (pp. 405-412). IEEE.

List of Figures

List of Tables