

DISS. ETH NO. 20375

Automatic Head and Face Analysis for Human-Computer Interaction

A dissertation submitted to
ETH ZURICH

for the degree of
Doctor of Sciences

presented by
Gabriele Fanelli
M.Eng. Sapienza University of Rome
born July 12, 1980
citizen of Italy

accepted on the recommendation of
Prof. Dr. Luc Van Gool, ETH Zurich and KU Leuven, examiner
Prof. Dr. Nicu Sebe, University of Trento, co-examiner

2012

TO MY FAMILY.

Abstract

Humans rely on head and facial movements for numerous tasks, especially in relation to communication. From speech to facial expressions, from nods to the subtlest non-verbal cues, our heads produce a great amount of information which we effortlessly read and transmit every day. Accurate and robust algorithms for the analysis and synthesis of both head and facial motions have therefore been actively advocated in the computer vision, machine learning, and computer graphics research communities. The driving force to these efforts is represented by the countless possible applications of such automatic methods: From security to health care, from human-computer interaction to intelligent tutoring, just to name some.

This work presents new tools for the analysis of head and facial motion targeted at improving our experience in the interaction with machines, nowadays still performed mainly through unnatural devices like keyboards and mice.

A first contribution is a method for robust mouth localization in videos of talking people. The algorithm does not rely on specific features like lip contours to be visible and we experimentally prove it accurate enough for the speech recognition task, achieving results comparable to the ones obtained from semi-automatically cropped mouth images.

We further describe a method for the automatic classification of facial videos into a discrete set of expression labels which also localizes the expression's apex in time. Our voting approach achieves results comparable to the state of the art when applied to standard databases and proves capable of handling a certain degree of occlusion.

A multimodal corpus of affective speech is presented next, together with procedures for its acquisition and automatic labeling. The recordings

include high quality facial scans of native English speakers engaged in emotional speech. Though we used video clips to elicit the affective states, thus trading naturalness for quality, an online evaluation showed that the recorded data retained the expressivity of the inductive films. The corpus is targeted to the research fields of realistic visual speech modeling for animation and recognition, 3D facial features localization, and view-independent expression recognition.

Finally, we present a framework for head pose estimation from depth images and extend it to localize a set of facial features in 3D. Our algorithm handles large rotations, partial occlusions, and noisy depth data. Moreover, it works on each frame independently and in real time, thus lending itself as a complement to tracking algorithms for their initialization and recovery. We thoroughly evaluate the system on challenging and realistic datasets, among which stands out a new annotated head pose database collected using a Microsoft Kinect, which we made available to the community.

Sommario

I movimenti della testa e del volto sono parte fondamentale della comunicazione tra esseri umani. Parlato, espressioni facciali, cenni del capo: la testa ed il viso producono un gran numero di informazioni che ogni giorno trasmettiamo e decifriamo naturalmente. Innumerevoli applicazioni pratiche beneficerebbero di algoritmi affidabili per l'analisi e la sintesi dei movimenti del capo, ragione che ha spinto i recenti sforzi fatti nei campi della visione artificiale, apprendimento automatico e computer grafica.

Questa tesi presenta nuovi metodi per l'analisi automatica dei movimenti della testa e del volto umani. Lo scopo principale è di migliorare l'interazione uomo-macchina, oggi per la maggior parte ancora condotta tramite dispositivi poco naturali come tastiera e mouse.

Un primo contributo è un metodo per localizzare la bocca in video raffiguranti persone che parlano. L'algoritmo non dipende da specifici tratti facciali come il contorno delle labbra e quindi non è suscettibile della loro parziale copertura. Gli esperimenti mostrano come il metodo suggerito sia utilizzabile per il riconoscimento automatico del parlato, ottenendo risultati simili a quelli ricavati da immagini della bocca estratte tramite intervento manuale.

Descriviamo poi un metodo per il riconoscimento automatico delle espressioni facciali da video e la localizzazione delle stesse nella dimensione temporale. Il nostro approccio ottiene risultati simili allo stato dell'arte quando applicato a delle basi di dati standard e si dimostra inoltre capace di funzionare anche quando parti del volto non sono visibili.

Segue una raccolta di scansioni facciali 3D di alta qualità, con corrispettivo audio, di 14 madrelingua Inglesi impegnati nella produzione di parlato emozionale. I dati sono presentati insieme alle procedure necessarie per la loro acquisizione e annotazione automatica. L'uso di video

per indurre gli stati affettivi è necessario per ottenere dati di alta qualità, a scapito della loro naturalezza. Un sondaggio online dimostra che i dati raccolti trasmettono emozioni simili a quelle provate guardando i video originali. Il database può essere utile per la ricerca sulla modellazione delle deformazioni facciali associate al parlato per l'animazione e il riconoscimento, oltre che sulla localizzazione di tratti facciali in 3D.

Infine, viene presentato un algoritmo per la stima della postura della testa da immagini di profondità, esteso alla localizzazione di alcuni importanti tratti facciali in 3D. Il metodo proposto riesce anche in presenza di rotazioni notevoli, immagini parzialmente corrotte o di bassa qualità. Il funzionamento in tempo reale e su ogni immagine indipendentemente ne fanno un complemento ideale ad algoritmi di tracciamento esistenti per la loro inizializzazione e recupero. Un'accurata valutazione dell'approccio proposto è eseguita su basi di dati impegnative e realistiche, tra cui un nuovo database registrato con un Microsoft Kinect, annotato con la postura delle teste e reso disponibile alla comunità scientifica.

Acknowledgements

Pursuing a Ph.D. is a long journey, often along uncertain paths. Many helped me find my way through years of studies, failures, successes, sadnesses, and joys.

First of all, thanks to Prof. Dr. Luc Van Gool, for believing in me, for allowing me to work in a truly inspiring environment, and for his motivating advices. I am also grateful to Prof. Dr. Nicu Sebe for acting as co-referee for my dissertation.

Secondly, I owe a lot to Dr. Jürgen Gall, for his enlightened guidance and constant availability, and to Dr. Thibaut Weise for his generous and continuous support.

During my years at ETH I had the fortune to be surrounded by a lot of great people. First of all, my collaborators, who contributed to this thesis: Matthias Dantone, Angela Yao, Andrea Fossati, Harald Romsdorfer, and Mihai Gurban. The whole Biwi crowd meant interesting discussions and a great deal of good memories; thanks in particular to Stefano Pellegrini, Marcin Eichner, Alex Mansfield, Severin Stalder, Fabian Nater, Stephan Gammeter, Lukas Bossard, Valeria De Luca, Christian Leistner, Mukta Prasad, Alessandro Prest, Peter Baki, Jianke Zhu, Beat Fasel, and Wicher Visser. A special mention goes to the administrative staff, always ready to help: Barbara Widmer, Vreni Vogt, Christina Krueger, and Fiona Matthews. Thanks also to my office mates Esther Koller-Meier, Konrad Schindler, and Nima Razavi.

I have collected a few datasets for my research and my gratitude goes to the volunteers who patiently had their faces recorded during the years. I would also like to thank the students who chose me as a supervisor, for that was an enriching experience. Moreover, I am grateful for the support I received from the projects IM2, VSHMI, Hermes, TANGO, and the people working within them.

In case you are reading the printed version of this thesis, its cover was designed by Diego Aluisi; with him, I'd like to thank all my friends, old and new, for all the great times spent together.

Of course, my family deserves its great share of gratitude. My mother Grazia, my sister Serenella, my brother Pierpaolo, my aunt Germana and my uncle Ippolito all gave me constant support and never lost faith in me. My thoughts go also to my father Vittorio, he would be proud of me right now.

Last but definitely not least, the person who shared my adventure the most, who multiplied my happiness during success and comforted me in the hard moments: Anna, you are the most important person in my life.

Contents

1	Introduction	3
1.1	Contributions	4
1.2	Organization	5
2	Hough Transform-based Mouth Localization	7
2.1	Related Work	9
2.1.1	Hough Forests	10
2.2	System Overview	13
2.3	Face Normalization	13
2.4	Hough Transform-based Mouth Localization	15
2.4.1	Learning	16
2.4.2	Localization	19
2.5	Audio-Visual Speech Recognition	19
2.6	Experiments	21
2.6.1	Estimation of Scale and Orientation	21
2.6.2	Mouth Localization	22
2.6.3	Speech Recognition	27
2.6.4	Processing Speed	29
2.7	Conclusion	31
3	Facial Expression Recognition from Video Sequences	33
3.1	Related work	36
3.2	Voting Framework	38
3.2.1	Training	38
3.2.2	Facial Expression Classification	40
3.3	Building the Expression Tracks	40
3.3.1	Feature Extraction	42
3.4	Experiments	44
3.5	Conclusion	51

4	Acquisition of a Multimodal Corpus of Affective Communication	53
4.1	Related Work	56
4.2	Data Acquisition	59
4.2.1	Corpus Definition	59
4.2.2	Recording Protocol	59
4.2.3	Video Processing	61
4.2.4	Audio Processing	64
4.3	Evaluation	70
4.3.1	Eliciting Videos Evaluation	70
4.3.2	Corpus Evaluation	73
4.3.3	Data Analysis	77
4.4	Conclusion	81
5	Random Forests for Real Time 3D Face Analysis	83
5.1	Related work	85
5.1.1	Head pose estimation	85
5.1.2	Facial features localization	87
5.2	Random forests for 3D face analysis	89
5.2.1	Random forest	90
5.2.2	Head pose estimation	91
5.2.3	Facial features localization	97
5.3	Evaluation	98
5.3.1	Head pose estimation - high resolution	98
5.3.2	Head pose estimation - low resolution	110
5.3.3	Facial features localization	120
5.4	Conclusions	131
6	Conclusions and Outlook	135
6.1	Discussions	135
6.2	Future Work	137
	Bibliography	141

1

Introduction

*“Ut imago est animi voltus sic indices oculi.” M. T. Cicero,
De oratore, 55 B.C.*

“The face is a picture of the mind as the eyes are its interpreters”. Today as centuries ago, the face remains one of the most interesting sights the human eyes can come across.

Every day, we all read and convey thousands of vital information through the face and its deformations. Already in its neutral configuration, it reveals identity, gender, age, attractiveness, health, etc. But being a marvelous engineering product of evolution, the human face can also produce a great number of expressions. Such movements, though often very subtle, give us very important clues about someone’s emotional state, focus of attention, attentiveness, sincerity, etc. Examples of situations when we use our innate face reading capabilities are countless, for human communications rely on facial expressions and head movements for a great part. Speech, our other fundamental mean of information exchange, is also uttered by the mouth, conveniently located where it can easily be read when the audio is of no use because of noise: on the face.

For all the above reasons, the human face has been an important object of study in many research fields, ranging from psychology to computer science. It goes without saying that automatic methods for the analysis of the face are of great importance for many useful application: From security to human-computer interaction, from health care to intelligent tutoring.

Throughout evolution, mankind has seen trillions of faces in all their deformations. This made us experts at reading facial images through simple and robust algorithms, yet to be fully understood, implemented in specially devoted areas of the brain [Tsao *et al.* 2006, Kanwisher and Yovel 2006]. The way we produce and perceive the motions and deformations on our faces is so complex that the field is still an open ground for psychologists and anthropologists, let alone computer scientists. Matching our brain’s outstanding face reading capabilities is one of the goals of the computer vision and machine learning communities, without having millions of years of evolution available for training.

1.1 Contributions

This work wants to bring new tools to the research in the fields of face and head movements analysis, with a focus on human-computer interaction applications. The main contributions of this thesis can be summarized as follows.

- First, we propose a novel method for robustly localizing the mouth region in videos of talking faces, with a direct application to audio-visual speech recognition. The algorithm takes a voting approach, where different image regions each suggest a possible location of the mouth center, thus being less sensitive to partial occlusions.
- A second contribution is a fully automatic system for facial expression recognition from video. We extend a voting algorithm initially designed for human action recognition to the task of classifying sequences of facial images into the discrete set of expressions of the six basic emotions.
- Thirdly, we present a setup for the acquisition and automatic labeling of a large audio-visual corpus of emotional speech. The resulting Biwi 3D Audiovisual Corpus of Affective Communication, $B3D(AC)^2$, is a high quality database, aimed at research fields like the modeling, recognition, and synthesis of multimodal emotional speech.

- In the last part of this thesis, we present a framework for the automatic, real time, frame based head pose estimation from depth data of varying quality. We further extend the algorithm to localize the 3D position of 14 facial feature points.

1.2 Organization

In Chapter 2, we describe a method for automatic real time mouth localization from standard videos of a talking person. The automatic tracking of the eye centers allows us to normalize the facial images with respect to scale and orientation. We then take a voting approach for the actual mouth localization, where different image patches are mapped to probabilistic votes on a Hough image. The mapping itself is performed by a random forest [Gall *et al.* 2011]. The proposed method does not rely on the detection of mouth corners or lip contours, which could be occluded, and proves successful when applied to the task of audio-visual speech recognition. In this first chapter, beside the related works on mouth detection and audio-visual speech recognition, we also give an introduction to random forests, which are used in several other parts of the thesis. The work presented in this chapter appeared in [Fanelli *et al.* 2009].

Chapter 3 presents a fully automatic system for the recognition of facial expressions from video. After eye tracking and image normalization, here we extend the Hough forest algorithm, in its action recognition variant [Gall *et al.* 2011], to the task of facial expression classification. The image normalization process and the more discriminative image features allow us to recognize the subtler movements on the face. This part of the thesis was previously presented in [Fanelli *et al.* 2010a].

In Chapter 4, we introduce the Biwi Audiovisual Corpus of Affective Communication, $B3D(AC)^2$, and the framework used for its collection and automatic annotation. The corpus contains a large number of recordings of native English speakers engaged in emotional speech. Audio and dense dynamic 3D facial scans were automatically annotated, both in terms of phoneme segmentation and detailed tracking of the facial deformations through a generic template. This chapter contains work previously published in [Fanelli *et al.* 2010c] and [Fanelli *et al.* 2010b].

Chapter 5 presents a random forest framework for solving the problems of real time 3D head pose estimation and facial features detection. We use depth images of various quality, acquired both with a high-resolution scanner and an affordable Kinect camera. The proposed approach does not rely on the detection of specific features like the nose, runs on a frame-by-frame basis, and proves robust to rotations, facial expressions, facial hair, and partial occlusions. Parts of this work were already presented in [Fanelli *et al.* 2011a], [Fanelli *et al.* 2011b], [Fanelli *et al.* 2012b], and [Fanelli *et al.* 2012a]. In [Dantone *et al.* 2012], the algorithm is extended to jointly solve for head pose estimation and facial features localization, in real time, from standard 2D images with arbitrary conditions.

In Chapter 6, we summarize the thesis, discuss results, and point out future challenges in the discussed research topics.

Even if face analysis is the common denominator of this work, the single chapters touch different areas of research and tackle rather independent problems. For this reason, each chapter contains an overview of recent works related to the topic at hand.

2

Hough Transform-based Mouth Localization

Speech is among the most natural forms of human communication: We use it every day to convey complex messages efficiently and reliably, provided that a language is agreed upon. Clearly, this strongly motivates the development of robust and reliable speech-driven interfaces for the field of human computer interaction.

Even though recent Automatic Speech Recognition (ASR) systems have become reliable and usable by the large audience [Schalkwyk *et al.* 2010], they still suffer from noise on the audio channel, which is unavoidable in many application-relevant environments (*e.g.*, a busy street).

Multimodal approaches try to circumvent the problem by augmenting the audio stream with additional sensory information in order to improve the recognition accuracy [Potamianos *et al.* 2004]. In particular, the fusion of audio and visual cues is a popular choice [Petajan 1984] motivated by human perception: We use both audio and visual information when understanding speech [McGurk and MacDonald 1976]. There are indeed sounds which are very similar in the audio modality, but easier to discriminate visually, and vice versa. The so-called Audio-Visual Speech Recognition (AVSR) systems use both cues to recognize speech uttered by a person recorded using a camera and a microphone, significantly increasing performance over audio-only setups, especially when the auditory channel is corrupted by noise.

In order to extract visual features carrying information about the speech being pronounced, the area of the picture containing the mouth must be

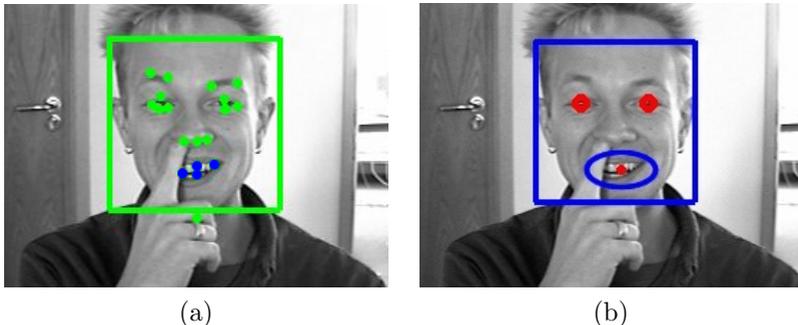


Figure 2.1: a) Facial points like mouth corners (blue dots) are sensitive to occlusions [Vukadinovic and Pantic 2005]. b) Our Hough transform-based approach localizes the center of the mouth (red dot) even in the case of partial occlusions. The ellipse indicates the region of interest used for visual speech recognition.

localized as a region-of-interest [Patterson *et al.* 2002], a set of feature points [Vukadinovic and Pantic 2005, Valstar *et al.* 2010], or lip contours [Luetttin and Thacker 1997, Liu *et al.* 2010, Li *et al.* 2012]. Although the lip contours appear to contain more information about the mouth shape than the appearance inside a bounding box, they do not necessarily encode more information valuable for speech recognition, as demonstrated in [Potamianos *et al.* 1998]. In addition, extracting a bounding box is usually more robust and efficient than lip contour extraction approaches, most of which anyway need such a bounding box for initialization, as for example in [Liu *et al.* 2010].

In this chapter, we propose a method for localizing the mouth region in images of faces in a near-frontal view. Contrary to the many standard approaches which seek to extract its corners to estimate scale, position, and orientation of the mouth, we propose an algorithm based on the generalized Hough transform which lets different image patches cast votes for the mouth location. The rationale behind this choice is that a certain feature point might be difficult to detect due to occlusions, lighting conditions, or facial hair, as exemplified by Figure 2.1 a), where one of the mouth corners is occluded. For this reason, instead of detecting specific fiducials, our method maps the appearance of small image patches

into probabilistic votes which accumulate in a Hough image, the peak of which is considered to be the mouth center. This approach allows for the localization of the mouth even in difficult situations when parts of it are covered, as shown in Figure 2.1 b).

In order to make the process faster and usable for real time applications, we exploit the shape of the iris, whose rotation invariance is unique among the other facial features and allows for very efficient localization using isophote curvature [Valenti and Gevers 2011]. Knowing the approximate in-plane rotation and scale of the face from the eye centers, the lower face region is normalized with respect to scale and orientation. We thus reduce the great appearance variations which a highly deformable object like the mouth is capable of. The actual learning of the mapping is performed by a random forest, or Hough forest [Gall and Lempitsky 2009, Gall *et al.* 2011].

2.1 Related Work

Rather than recognizing speech from the audio signal alone [Schalkwyk *et al.* 2010, Rabiner and Juang 1993, Jiang 2010], AVSR methods fuse features extracted from both the auditory and the visual channel to better recognize the words being pronounced by a speaker.

Pioneered by [Petajan 1984], AVSR is still an active area of research today [Galatas *et al.* 2011, Gurban and Thiran 2009, Cooke *et al.* 2006, Livescu *et al.* 2007, Lucey *et al.* 2002, Potamianos *et al.* 2004]. Several approaches have been proposed for combining audio and visual cues, based for example on artificial neural networks [Heckmann *et al.* 2001], support vector machines [Gordan *et al.* 2002], or AdaBoost [Yin *et al.* 2004]. In this work, we employ the commonly used multi-stream hidden Markov models (MSHMM) [Young *et al.* 1999]; however, since we focus on the visual part of the pipeline, *i.e.*, mouth localization within the image frame, any other standard recognition method could have been used.

For visual features, lip contours [Luetttin and Thacker 1997] and optical flow [Gray *et al.* 1996] share the popularity with image compression techniques such as Linear Discriminant Analysis (LDA), Principal Component Analysis (PCA), Discrete Cosine Transform (DCT), or

Discrete Wavelet Transform (DWT) [Potamianos *et al.* 2004]. Within monomodal visual speech recognition (lip reading), snakes [Bregler and Omohundro 1995] and Active Shape Models (ASM) [Kaucic *et al.* 1996, Matthews *et al.* 1998] have been intensively studied for lip tracking.

Lip contours-based methods do not encode all possible geometric information (like the tongue), therefore space-time volume features have been proposed for lip-reading in [Pachoud *et al.* 2008].

Most of the listed approaches assume that a bounding box can be reliably extracted around the mouth in the image, which is the contribution of this chapter.

2.1.1 Hough Forests

Here we introduce the concepts of Hough transform, random forests, and their combination [Gall *et al.* 2011].

The Hough transform, originally designed to detect straight lines [Duda and Hart 1972] and successively extended to localize generic parametric shapes [Ballard 1981], is an established method in computer vision, especially for the task of object detection [Leibe *et al.* 2008, Maji and Malik 2009, Bourdev and Malik 2009, Opelt *et al.* 2008, Ommer and Malik 2009, Gall and Lempitsky 2009, Okada 2009, Lehmann *et al.* 2011]. The process involves splitting the image into small appearance patches, each of which can vote for the hypotheses about the object's configuration which might have generated it. Such votes are accumulated into a Hough image, living within a parametric domain called Hough space. A point in such a space corresponds to a particular configuration of an object, *e.g.*, its location on the image plane, and the detection task boils down to locating the highest peaks of accumulated votes. The height of a peak additionally provides a measure of the detection's confidence.

Hough transform-based detection methods model the shape of the object implicitly, gathering the spatial information from a large set of different object patches annotated with the location of objects of interest. Learning involves the construction of the appearance codebook and, for each codebook entry, the distribution of object parameters which generated

it. Thanks to its additive nature, the generalized Hough transform can handle noisy measurements, partial occlusions, and large variations in shape and appearance. This is the case for the mouth, capable of great appearance changes through its many possible configurations (*e.g.*, open and closed).

Decision trees [Breiman *et al.* 1984] split a hard problem into easier ones, solvable using simple rules. As such, a tree can perform highly non-linear mappings from complex input spaces to simpler output spaces. All non-leaf nodes in a tree contain a binary test, which guides a data sample towards the left or the right child. The tests are chosen in a supervised learning framework and building a tree boils down to selecting the tests which cluster the annotated training samples such as to allow good predictions using simple models. The leaves store such models, constructed using the annotated samples left at train time. Random forests are collections of trees [Amit and Geman 1997], each trained on a randomly selected subset of the available data; this reduces over-fitting in comparison to trees trained on the whole dataset, as shown by [Breiman 2001]. Randomness can be introduced also in the pool of binary tests available for optimization at each node.

At run time, a test sample visits all the trees, ending up in a leaf in each of them. The final output of the forest is computed by averaging the results of all trees, according to the models stored at the leaves. Figure 2.2 shows a toy example of random forest: A data sample is guided through the trees until a leaf; there, actions are taken depending on the retrieved models.

Random forests have become a popular method in computer vision; they have been successful in semantic segmentation [Shotton *et al.* 2008], key-point recognition [Lepetit *et al.* 2005], object detection [Gall and Lempitsky 2009, Gall *et al.* 2011], action recognition [Yao *et al.* 2010, Gall *et al.* 2011], or real-time human pose estimation [Shotton *et al.* 2011, Girshick *et al.* 2011]. They are well suited for time-critical applications, since they are very fast at both training and testing, lend themselves to parallelization [Sharp 2008], and are inherently multi-class. For the interested reader, the tutorial of [Criminisi *et al.* 2011] offers a detailed introduction to the use of decision forests in computer vision.

Hough forests [Gall *et al.* 2011] combine the benefits of random forests with those of the generalized Hough transform. Successful applications of

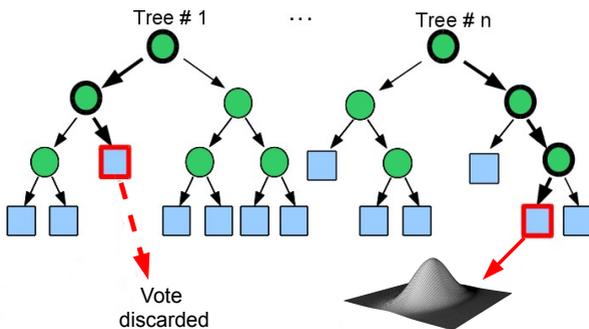


Figure 2.2: Example of a random forest. The binary tests at the nodes guide a sample down the trees. At the leaves, further actions are taken depending on the model stored at train time.

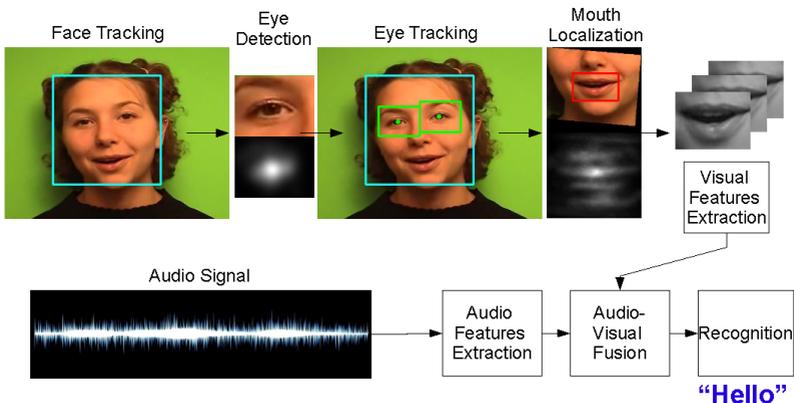


Figure 2.3: Overview of our AVSR system. The visual pipeline is shown at the top: The detected face bounding box is used to define search regions for tracking the eyes. The mouth detector is applied to images of the lower part of the face, scaled and rotated according to the eye positions. At the bottom right, the features extracted from the stream of normalized mouth images and from the audio signal are fused allowing for the actual speech recognition to take place.

Hough forests include class-specific object detection [Gall and Lempitsky 2009, Okada 2009] and action recognition [Yao *et al.* 2010].

2.2 System Overview

The pipeline of our audio-visual speech recognition system is depicted in Figure 2.3. Our work focuses on the visual pipeline, expanded in the upper region of the image.

The first necessary step for any AVSR algorithm is face detection, for which we use the popular method of [Viola and Jones 2004]. To cope with appearance changes, partial occlusions, and multiple faces, we employ the online-boosting tracker of [Grabner *et al.* 2006] to follow the head across consequent frames. The algorithm uses the current bounding box (containing the target face) and its surroundings as positive and respectively negative samples for updating the internal classifier.

Assuming the face to be pictured in a near-frontal view, the bounding box returned by the tracker allows us to estimate the rough positions of the eyes thanks to anthropometric relations. We then estimate scale and in-plane rotation of the face by filtering the positions of the irises, which we detect leveraging their circular shape as explained in Section 2.3. With this information at hand, we crop the lower part of the face and normalize it. In this way, the resulting picture contains a mouth which is horizontal and has a specific size. This normalization step allows us to run the mouth detection algorithm at only one scale and rotation, drastically reducing computation time. The actual method for mouth localization using a Hough forest is presented in Section 2.4. Finally, features are extracted from the stream of normalized mouth images and from the audio signal and fused in order to recognize the spoken words, as described in Section 2.5. A thorough set of experiments is presented Section 2.6.

2.3 Face Normalization

We use the method of [Valenti and Gevers 2011] for accurate eye center localization, based on isophote curvature. The main idea relies on

the radial symmetry and high curvature of the eyes' brightness patterns. An isophote is a curve going through points of equal intensity, its shape being invariant to rotations and linear changes in the lighting conditions [Lichtenauer *et al.* 2005]. Because isophotes never intersect each other, they can be used to fully describe a picture.

For each pixel p in an image, a displacement vector is computed as:

$$\{D_x, D_y\} = -\frac{\{L_x, L_y\} (L_x^2 + L_y^2)}{L_y^2 L_{xx} - 2L_x L_{xy} L_y + L_x^2 L_{yy}} \quad (2.1)$$

where L_x and L_y are the image derivatives along the x , respectively y axis. The value of an accumulator image at the candidate center $c = p + D$ is incremented by the curvedness of the original image measured at p , computed as:

$$\sqrt{L_{xx}^2 + 2L_{xy}^2 + L_{yy}^2}, \quad (2.2)$$

thus giving higher weights to center candidates coming from highly curved isophotes.

Knowing that the pupil and the iris are generally darker than the neighboring region (sclera), only transitions from bright to dark areas are considered, *i.e.*, situations where the denominator of Equation (2.1) is negative and the curvature agrees with the gradient's direction. The eye center is finally located by convolving the accumulator image with a smoothing Gaussian kernel and selecting the peak location.

The above method fails when the irises are not visible, *e.g.*, due to closed eyelids or strong reflections on the glasses. When tracking a video sequence, this can lead to sudden jumps of the detections. Such errors propagate through the whole pipeline, leading to wrong estimates of the mouth scale and rotation, and thus of its bounding box location. To mitigate the effects of these errors, we smooth the pupils' trajectories using Kalman filters [Welch and Bishop 2001], one for each eye center.

2.4 Hough Transform-based Mouth Localization

We use Hough forests [Gall *et al.* 2011] for the purposes of mouth localization. The trees learn the discriminative appearance of image feature patches and their corresponding mapping into votes in a Hough space $\mathcal{H} \subseteq \mathbb{R}^H$. For the task at hand, the Hough space encodes the hypothesis $\mathbf{h}(c, \mathbf{x})$ for class c (mouth versus rest of the face) and position on the image plane, \mathbf{x} .

The voting process is exemplified in Figure 2.4. (a) Probabilistic votes for the mouth center are cast based on the appearance of example patches. Note how the magenta patch is classified as negative, *i.e.*, uninformative about the mouth position, and thus not allowed to vote. (b) All votes are summed up into a Hough image, where (c) the peak is taken as the mouth center. The bounding box enclosing the mouth is finally scaled and rotated according to the detected eye positions. The implicit shape model (ISM) can be modeled by an explicit codebook as originally done in [Leibe *et al.* 2008], but the construction of codebooks is expensive due to the required clustering techniques and the linear matching complexity. We therefore choose to follow a random forest framework, where both learning and matching are less computationally demanding.

Following [Gall *et al.* 2011], we denote with \mathcal{I} a mapping from the input domain $\mathbf{y} \in \Omega \subseteq \mathbb{R}^2$ (the image plane) to the set of various feature channels $(I^1(\mathbf{y}), I^2(\mathbf{y}), \dots, I^F(\mathbf{y}),) \in \mathbb{R}^F$. After training, the leaves in the forest $\{L\}$ model the mapping from the appearance $\mathcal{I}(\mathbf{y})$ of an image patch centered at \mathbf{y} to the probabilistic Hough vote:

$$\mathcal{L} : (\mathbf{y}, \mathcal{I}) \rightarrow p(\mathbf{h} | L(\mathbf{y})), \quad (2.3)$$

where $p(\mathbf{h} | L(\mathbf{y}))$ is the distribution of Hough votes in the space \mathcal{H} .

In the following, we describe how the mapping is learned and the forest built from annotated training data (Sec. 2.4.1) and how it is used to localize the mouth in a new image (Sec. 2.4.2).

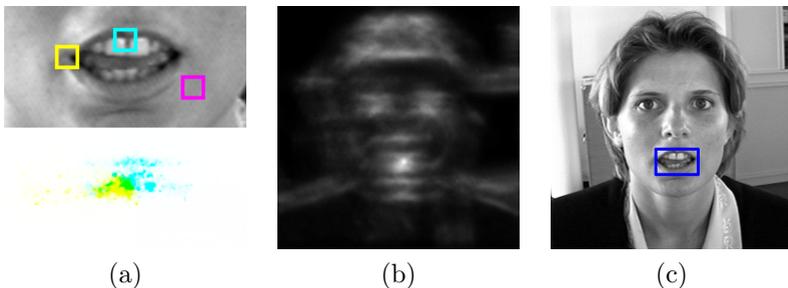


Figure 2.4: (a) For each of the emphasized patches (top), votes are cast for the mouth center (bottom). While lips (yellow) and teeth (cyan) provide valuable information, the skin patch (magenta) casts votes with a very low probability. (b) Hough image after accumulating the votes of all image patches. (c) The mouth is localized by the maximum in the Hough image.

2.4.1 Learning

Building a forest is a supervised learning problem, *i.e.*, training data need to be annotated with labels on the desired output space. In our case, we are given images of the lower part of the face, normalized with respect to scale and orientation, and annotated with the mouth center location (the bounding box enclosing the mouth has the same size for all samples). The following procedure applies similarly to multi-class or higher-dimensional problems.

Each tree in a forest is constructed based on a set of patches $\{\mathcal{P}_i = (\mathcal{I}_i, c_i, \mathbf{d}_i)\}$, where \mathcal{I}_i represents the appearance of patch i and c_i its class label. The 2D offset vector \mathbf{d}_i represents the patch’s relative displacement with respect to the object center. In our case, a patch can belong to a mouth region (positive, $c_i = 1$), or not (negative, $c_i = 0$). The feature channels I_i^f include raw image intensities and derivative filter responses. We compute such features from fixed size image patches, as the ones shown in Figure 2.4 (a). The training patches are randomly sampled from mouth regions (positives) and non-mouth regions (negatives).

Following the random forest framework [Breiman 2001], we build each tree by recursively optimizing its nodes, starting from the root.

For each non-leaf node we choose a binary test out of a set of randomly generated tests $\{\phi^k\}$, after having evaluated them on the set of training patches $S = \{\mathcal{P}_i = (\mathcal{I}_i, c_i, \mathbf{d}_i)\}$ available at that node. The chosen test splits the patches into two new subsets which are passed to the children: A patch which satisfies the test goes to the right, otherwise to the left. Given the appearance \mathcal{I} of a patch, a test $\phi(\mathcal{I}) \rightarrow \{0, 1\}$ compares the difference of channel values I^f for a pair of locations \mathbf{p} and \mathbf{q} against a threshold τ :

$$\phi_{f,\mathbf{p},\mathbf{q},\tau}(\mathcal{I}) = \begin{cases} 0, & \text{if } I^f(\mathbf{p}) - I^f(\mathbf{q}) < \tau \\ 1, & \text{otherwise.} \end{cases} \quad (2.4)$$

The process iterates until a leaf is created when either the maximum tree depth is reached (15), or less than a minimum number of training samples are left (20). A leaf node L stores the following information, according to the patches which are left at the time of its creation:

- The probability of belonging to a mouth $p(c = 1)$, approximated by the proportion of positive samples reaching the leaf at train time;
- The displacement vectors associated with all positive patches, *i.e.*, $D_m^L = \{\mathbf{d}_i\}_{c_i=1}$.

The leaves build an implicit codebook and model the spatial probability of the mouth center \mathbf{x} given the appearance $\mathcal{I}(\mathbf{y})$ of a patch located at position \mathbf{y} on the image. Such probability is represented by a non-parametric density estimator computed over the set of positive samples D_m^L and by the probability that the image patch belongs to the mouth $p(c = 1 | \mathcal{I}(\mathbf{y}))$:

$$p(\mathbf{x} | \mathcal{I}) = \frac{1}{Z} p(c = 1 | \mathcal{I}) \left(\frac{1}{|D_m^L|} \sum_{\mathbf{d}_i \in D_m^L} \frac{1}{2\pi\sigma^2} \exp\left(-\frac{\|(\mathbf{y} - \mathbf{x}) - \mathbf{d}_i\|^2}{2\sigma^2 \mathbf{I}_{2 \times 2}}\right) \right). \quad (2.5)$$

In Equation (2.5), we omitted the dependence of \mathcal{I} on \mathbf{y} for simplicity, $\sigma^2 \mathbf{I}_{2 \times 2}$ is the covariance of the isotropic Gaussian Parzen window, and

Z is a normalization constant. The probabilities for three patches are illustrated in Figure 2.4 (a): only patches extracted from the mouth region are actually casting votes.

Because the quantity in (2.5) is the product of a class and a spatial probability, the optimization procedure at the nodes is designed to try and form clusters of patches with increasingly lower class and spatial uncertainty as the tree deepens. For this reason, the splits produced by the tests $\{\phi^k\}$, generated by randomly sampling $f, \mathbf{p}, \mathbf{q}$ and τ , are evaluated with respect to both a measure of the class uncertainty \mathcal{U}_1 and of the spatial uncertainty \mathcal{U}_2 on all patches S available at that node. In practice, at each node we randomly select one of the two measures with equal probability and finally pick the test which minimizes the sum of the chosen measure computed over the left and right clusters:

$$\phi^* = \underset{k}{\operatorname{argmin}} \left(\mathcal{U}_*(S_l) + \mathcal{U}_*(S_r) \right), \quad (2.6)$$

where $\star = 1$ or 2 indicates the measure type. $S_l = \{S_i | \phi^k(\mathcal{I}_i)=0\}$ and $S_r = \{S_i | \phi^k(\mathcal{I}_i)=1\}$ represent the clusters of patches sent to the left, respectively right child.

We use the same measures proposed by [Gall *et al.* 2011], *i.e.*, we define the class uncertainty based on the entropy over the class labels:

$$\mathcal{U}_1(S) = -|S| \cdot \sum_{c \in \{0,1\}} p(c|S) \ln p(c|S), \quad (2.7)$$

where $|S|$ is the number of patches at the current node and $p(c|S)$ is approximated by the ratio of patches with class label c in the set S .

For the spatial uncertainty measure \mathcal{U}_2 , we use the impurity of the offset vectors \mathbf{d}_i :

$$\mathcal{U}_2(S) = \sum_{i:c_i=1} (\mathbf{d}_i - \bar{\mathbf{d}})^2, \quad (2.8)$$

where $\bar{\mathbf{d}}$ is the mean of the spatial vectors \mathbf{d}_i computed over all positive patches in the set.

2.4.2 Localization

The detection process is illustrated in Figure 2.4. Given a face image, normalized with respect to scale and orientation as explained in Section 2.3, we densely extract feature patches. For each tree in the forest $\{\mathcal{T}_t\}_{t=1}^T$, non-leaf nodes guide a test patch extracted at position \mathbf{y} all the way down to a leaf L . The path undertaken by a patch depends on the results of the nodes' binary tests applied to the appearance $\mathcal{I}(\mathbf{y})$. Based on the models stored at the leaves, a vote is cast onto the Hough space using Equation (2.5), averaged over the whole forest:

$$p(\mathbf{x}|\mathcal{I}(\mathbf{y})) = \frac{1}{T} \sum_{t=1}^T p(\mathbf{x}|\mathcal{I}(\mathbf{y}); \mathcal{T}_t), \quad (2.9)$$

The probabilistic votes produced by the patches extracted at all possible image locations \mathbf{y} are then accumulated in the Hough image, see Figure 2.4 (b). In practice, we add the discrete votes $p(c = 1|\mathcal{I})/|D_m^L|$ to the pixels $\{(\mathbf{y} - \mathbf{d})|\mathbf{d} \in D_m^L\}$ for each tree and apply the Gaussian kernel after voting. The location where the generalized Hough transform gives the strongest response is considered to be the center of the mouth, as shown in Figure 2.4 (c). The value at the peak measures the confidence of the detection. In a standard object detection framework, a threshold on such confidence would define when to trigger the detection, but in our case we assume a mouth to be always present in the image.

2.5 Audio-Visual Speech Recognition

Our work focuses on the mouth localization part of a AVSR system, independently from the chosen approaches for the actual recognition and feature fusion methods. In order to show the applicability of our method, we resort to the widely used multi-stream hidden Markov models (MSHMM), a generalization of standard HMMs [Young *et al.* 1999].

HMMs are used to model the behavior of systems which stochastically switch between discrete states. In a first order discrete Markov chain, the probability of being in a specific state depends only on the state itself and on the previous one. The term *Hidden* stands for the fact that states

are not directly observable: What we can measure are some products of the system’s states, called observations.

HMMs are commonly used in ASR as models of the temporal dynamics of the speech signal. A state of a HMM can either represent a word in the dictionary, or rather a single speech sound, also called phoneme, *i.e.*, “the smallest segmental unit of sound employed to form meaningful contrasts between utterances” [Association and Corporate 1999]. Most systems use left-right models [Bakis 1976], where states are only visited from left to right, *i.e.*, from an initial state to a final one. In an ASR system, the observations are features extracted from the audio channel. Given a sequence of observations, the goal is to recognize the most likely word model which might have generated them, for which the Viterbi algorithm is generally used [Rabiner 1990]. While typical HMMs use one Gaussian mixture (GMM) to model the observation probabilities at each state, a multi-stream HMM has several GMMs per state, *e.g.*, one for each input modality s .

In our system, the joint probability of the multimodal observations $O = (o_1, \dots, o_t)$ and the states $Q = (q_1, \dots, q_t)$, is given by:

$$p(O, Q) = \prod_{q_i} b_{q_i}(o_i) \prod_{(q_i, q_j)} a_{q_i q_j} \quad (2.10)$$

where the probability of a transition from q_i to q_j is given by $a_{q_i q_j}$ and

$$b_j(o) = \prod_{s=1}^2 \left(\sum_{m=1}^{M_s} c_{j_s, m} N(o_s; \mu_{j_s, m}, \Sigma_{j_s, m}) \right)^{\lambda_s}. \quad (2.11)$$

In the above equation, $N(o; \mu, \Sigma)$ are the multi-variate Gaussians with mean μ and covariance Σ , weighted by $c_{j_s, m}$. We learn the model parameters independently for each modality. The weights $\lambda_s \in [0, 1]$ steer the influence of the modalities, with $\lambda_1 + \lambda_2 = 1$.

As observations, we extract features commonly employed in AVSR: Mel-frequency cepstral coefficients from the audio stream and DCT features from the normalized mouth images, where only the odd columns are used due to symmetry [Potamianos *et al.* 2004, Potamianos and Scanlon 2005]. For both types of features, we add the first and second temporal derivatives and normalize the sets as to have zero mean.

2.6 Experiments

We evaluated our system testing each component separately, from the eye detection part to the actual speech recognition performance.

In a first set of experiments, we initially assessed the quality of scale and orientation estimation method which relies on the detection of the eyes, then moved on to the mouth localization accuracy. We compared our results with the methods of [Vukadinovic and Pantic 2005] and [Valstar *et al.* 2010] for facial feature points detection, for which executables are available. We used the publicly available BioID face database [Jesorsky *et al.* 2001], which is often employed for comparing eye detection algorithms. The database is comprised of 1521 greyscale images of 23 individuals, acquired at different points in time under uncontrolled office illumination, with a resolution of 384x288 pixels. Subjects often show closed eyes, different facial expressions, and many of them wear glasses. Manually annotated ground truth is provided for the pupils and for 18 other facial points. As ground truth mouth center, we take the centroid of the 4 fiducial points on the mouth provided with the database (lip corners and outer lips' midpoints).

We then evaluated the system in terms of speech recognition performance on the CUAVE database [Patterson *et al.* 2002]. This is a standard database within the AVSR community, consisting of videos recorded in controlled conditions, at 29.97fps interlaced, with a resolution of 740x480. Each of the 36 subjects repeats the digits from “zero” to “nine” in American English.

2.6.1 Estimation of Scale and Orientation

Because the mouth localization performance directly depends on the quality of the estimation of the face's scale and orientation, as explained in Section 2.3, we performed the following experiment on the full BioID database. First we detected the face in each image (taking the largest bounding box in case of multiple detections) and searched for the eye centers within the two upper quarters of the face rectangle. Then, we computed the errors with respect to scale (inter-ocular distance) and in-plane rotation (angle formed by the line connecting the eyes and the horizontal image axis).

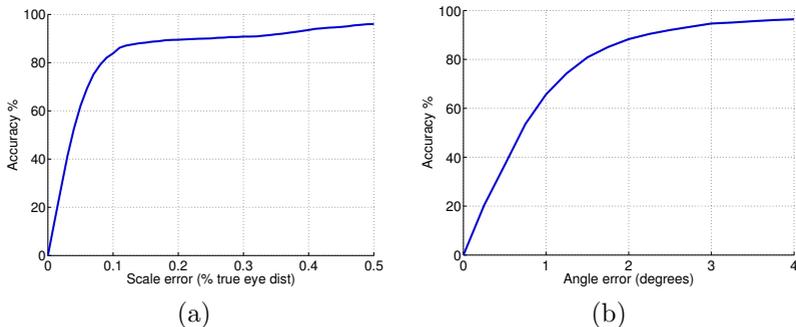


Figure 2.5: a) Accuracy vs. eye distance error (scale). b) Accuracy vs. angle error (rotation). The plots show the percentage of correctly estimated images as the threshold defining success increases.

Figure 2.5 shows the accuracy for the two measures, *i.e.*, the percentage of correct estimations as the error thresholds defining success increase. In Figure 2.5 (a) the accuracy is plotted against the error between the detected eye distance $dEye$ and the ground truth dGT , as $err = \frac{abs(dEye-dGT)}{dGT}$. In Figure 2.5 (b) instead, the accuracy is a function of the angle error in degrees. In 1.12% of the cases, no face was detected at all, moreover, sometimes the face detector gave wrong results, getting stuck on some clutter in the background. This partly explains why the curves in Figure 2.5 never reach 100%.

2.6.2 Mouth Localization

To evaluate the goodness of the mouth detection pipeline, we ran a 4-fold cross validation on the BioID database, *i.e.*, training the mouth detector on three quarters of the data, testing on the fourth, iterating, and averaging the results. We compared our results to the output of the facial feature detectors of [Vukadinovic and Pantic 2005] and [Valstar *et al.* 2010], using the authors' source code. As we localize the mouth center rather than the corners, we considered the centroid of the four mouth corners provided by the above detectors. For fairness to the competing methods, we have to specify that, because the BioID database does not contain labels of the subjects' identity, our experiments are

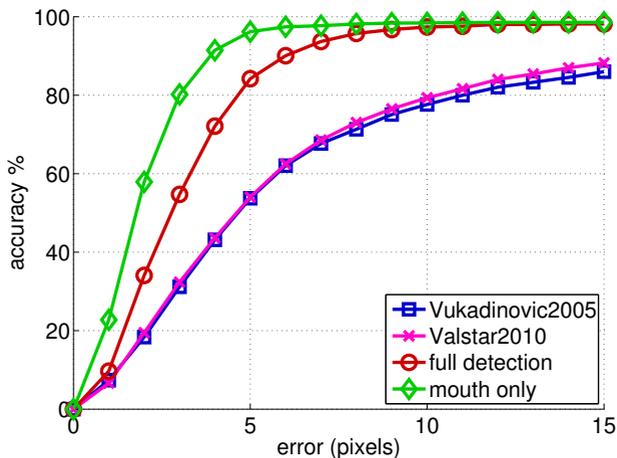


Figure 2.6: Accuracy vs. mouth center localization error (in pixels) between the methods of [Vukadinovic and Pantic 2005] (blue), [Valstar et al. 2010] (magenta), our full pipeline (red), and the mouth localization given the eye position from ground truth (green).

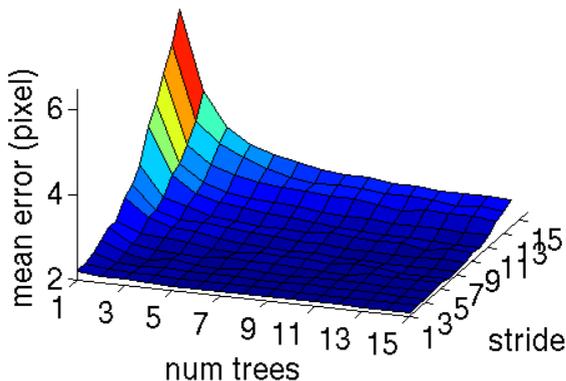


Figure 2.7: Average mouth localization error in pixels plotted against stride and number of trees in the forest. The error only increases noticeably when few trees are loaded and the stride parameter is large.

not subject-independent. This means that test images could contain subjects present in the set used to train the mouth detector. Moreover, the competitor facial feature point detectors were trained on different datasets altogether.

As already mentioned, face detection does not always succeed. Indeed, the detector of [Vukadinovic and Pantic 2005] did not return results in 9.67% of the cases, while the detector of [Valstar *et al.* 2010] often produced false positives. In order to remove the influence of errors originated in the face or eye detection parts of the pipeline, we performed a second test concentrating on the mouth localization alone, using the ground truth eye positions as initialization.

The curves in Figure 2.6 show the accuracy of the tested algorithms, in percentage of the correctly localized mouths, as the error threshold (in pixels) increases. Our method outperformed the competitor facial feature point detectors for the mouth localization task, both in the “full detection” (face, eyes, mouth), and “mouth only” type of experiment.

We additionally ran the “mouth only” test while varying two important parameters of the Hough-based detector: The number of trees in the forest and the stride controlling the density of the patches being sampled from a test image. The results in Figure 2.7 show that the average error remains low (around 2 pixels) even for a large stride and when only a few trees are employed. Steering these intuitive parameters thus allows the user to find a trade-off between accuracy of the detection and computation demand.

Figure 2.8 shows some successfully processed frames out of the BioID database. It can be seen that the full pipeline can cope with difficult situations like the presence of glasses, facial hair, and head rotations. On the other hand, Figure 2.9 shows some of the failure cases; in all the examples shown, the source of error is to be found in either the face or eye detection steps.

Table 2.10 summarizes the average errors and standard deviations produced by the tested methods, on the BioID dataset. For each method, we only considered images where the face detector returned a bounding box, *i.e.*, the executables of [Vukadinovic and Pantic 2005] and [Valstar *et al.* 2010] produced a text file containing the results. However, some false detections remained which increased the variance in the errors; this is



Figure 2.8: Some example successes of the system. Difficult situations like reflections on the glasses, facial hair and head pose changes are handled.

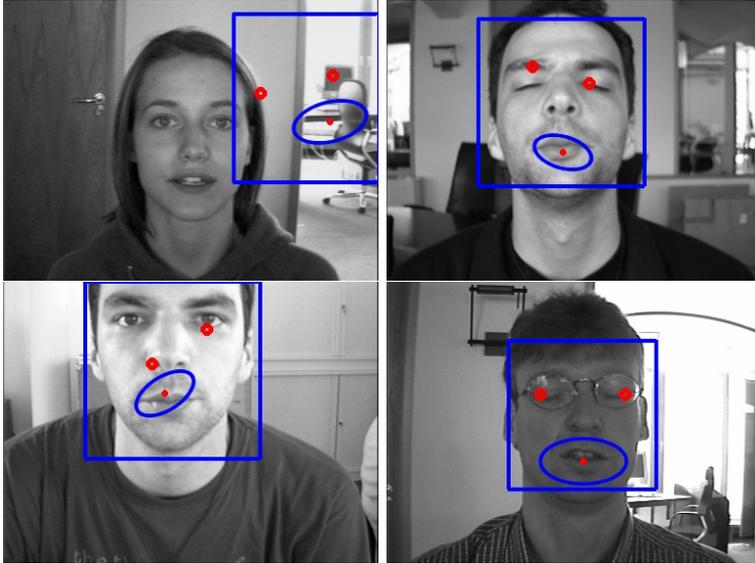


Figure 2.9: Some examples failures of the system on the BioID database. Here the mouth detection is doomed to fail if the face detection and/or the eye detection stages fail.

Full detection	Mouth only	[Vukadinovic and Pantic 2005]	[Valstar <i>et al.</i> 2010]
3.77 ± 6.88	2.10 ± 3.0	5.39 ± 4.35	11.65 ± 29.0

Figure 2.10: Mean and standard deviation of the errors (in pixels) for the mouth localization task on the BioID database.

particularly true for the face detector employed by [Valstar *et al.* 2010]. The “mouth only” error presents a low variance because of the use of ground truth eye locations. Indeed, because it relies on the shape of the irises, which are commonly occluded during blinking or by reflections on the glasses, the employed eye detector is an important cause of failure in a database like BioID.

2.6.3 Speech Recognition

As the goal of our system is to automatically provide normalized mouth images for the purposes of audio-visual speech recognition, we tested it on the CUAVE database [Patterson *et al.* 2002]. We concentrate on the subset of the database where subjects appear alone, keeping the face nearly frontal.

We use a mouth detector trained on the full BioID database. The videos were de-interlaced and linearly interpolated in order to match the frequency of the audio samples (100Hz).

The focus of this work is mouth localization, so we did not try to optimize the speech recognition system. As our approach is independent of the actual recognition system, it does not necessarily have to be coupled with multi-stream hidden Markov models. In all of the following experiments, we used the implementation of [Gurban and Thiran 2009], without the automatic feature selection part.

Being the power of AVSR evident especially when the audio channel is unreliable, we added white noise to the audio stream. We trained on clean audio and tested at different levels of Signal to Noise Ratios (SNRs). For the audio-visual fusion, we kept the audio and video weights λ_1 and λ_2 fixed for each test, and ran several trials varying the weights from 0.00 to 1.00 in 0.05 steps, finally picking the combination which gave the best recognition rate at each SNR.

Following [Young *et al.* 1999], we defined the accuracy as the number of correctly recognized words, C , minus the number of insertions, I (false positives detected during silence), divided by the number of words, N :

$$Accuracy = \frac{C - I}{N}. \quad (2.12)$$

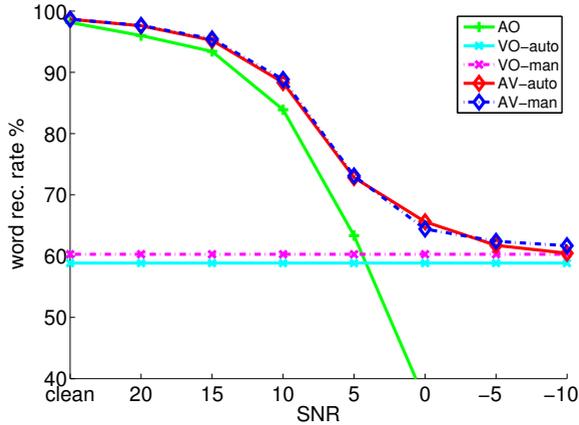


Figure 2.11: Word recognition rate for the audio-visual system using 80 visual features at different SNRs, for automatically and manually extracted mouth images. AV results always outperform monomodal settings.

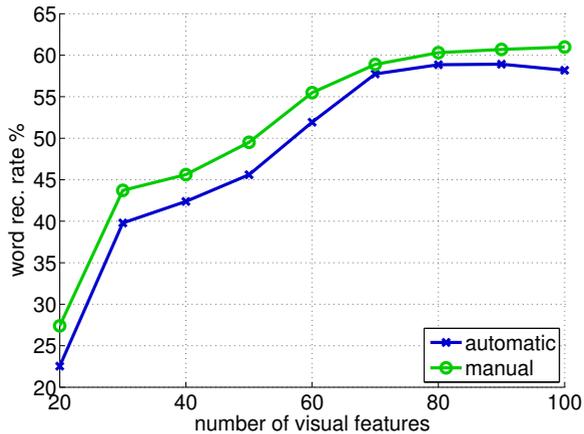


Figure 2.12: Influence of the number of features in video-only speech recognition systems. The accuracy of the recognizer using automatically extracted mouth images is only slightly lower than when manual intervention is employed for the mouth ROI extraction task.

We split the CUAVE sequences into 6, speaker-independent sets and performed a cross-validation by training on five groups while testing on the sixth and averaging the results of all combinations.

Figure 2.11 shows the performance of the system when we used a fixed number of visual features (80), at several SNR levels. We compared to the results obtained from semi-automatically extracted mouth-regions, which give the upper bound for the accuracy obtained with our automatic method. We also compared to the results of a audio-only (AO) and video-only (VO) recognizer. It can be noted that the multimodal approaches always outperform the monomodal ones. This is particularly clear for the audio-only system, which presents a steep decrease in recognition performance as the noise level increases. It is also interesting to notice that our automatic method for mouth ROI extraction performed only slightly worse than when manual intervention was used to annotate the mouth positions.

In Figure 2.12, we show the accuracy of the recognizer when only video features are used, as their number increases: Our approach performed best with 80 visual features (58.85%), while for greater sets the performance decreased slightly.

The images in 2.13 show some frames extracted from CUAVE videos with audio corrupted by white noise at 0 SNR. The yellow ellipse represents the localized mouth region fed to the AVSR systems and subtitles indicate the output words. In these examples, the multimodal recognizer always gets the correct word (red, right), while the video-only system gets confused or completely misses the utterance (yellow, left).

2.6.4 Processing Speed

When analyzing videos on a 2.8 GHz machine, the presented system (implemented in C++ without particular optimization efforts or the use of multi threading) runs at about 4fps. Most of the computation is concentrated in the mouth localization part: The face plus eyes tracking parts together run at 53fps.

A sensible decrease in processing time with a low price in accuracy can easily be achieved by loading a smaller number of trees and introducing a stride: For 10 trees and a stride of 4, the system runs at 15fps.



Figure 2.13: Some example frames showing the recognition capability of our system. The sequences were taken from the CUAVE database and white noise added to the audio channel (0 SNR). The yellow ellipse represents the localized mouth and the subtitles show the automatically recognized words. In these examples, the audio-visual system (AV - right) recognizes the word correctly, while the audio-only one (AO - left) makes mistakes or misses the utterance completely.

2.7 Conclusion

In this chapter, we have presented a novel and robust method for mouth localization which proved accurate enough for audio-visual speech recognition purposes. Even though speech recognition software has greatly improved in the last years, the visual modality will always be a valuable addition in noisy environments. Using the Hough forest algorithm [Gall *et al.* 2011], our method maps feature patches extracted from the lower part of the face to probabilistic votes in a Hough image, the peak of which is considered to be the mouth center. Compared to existing algorithms which rely on the detection of specific facial feature points, most notably mouth corners and lip contours, our voting approach is not jeopardized by the occlusion of any such key points. The proposed method is not only relevant for AVSR but also for lip reading and facial expression recognition or identification, where a normalized region-of-interest of the mouth can be required.

Our experiments show that our method outperforms recent facial feature detectors on near-frontal facial images and that the achieved word recognition rate for ASVR is near to the boundary obtained by employing mouth regions cropped using manual intervention. The system can achieve real time processing speed thanks to the estimation of scale and in-plane orientation of the face from filtered irises' detections. An additional speed-up with a small price in accuracy can be achieved by reducing the number of trees and the sampling rate of the mouth detector by introducing a stride.

The main shortcoming of the current system is its reliance on the detection of the eyes to make the mouth localization fast enough for interactive scenarios. This means that the algorithm will likely fail when the irises are not visible, because the eyes are closed or covered by sunglasses. Large out of plane rotations also compromise the results of the system, eventually because one of the eyes might become occluded.

Deaf people can achieve a reasonable speech perception using the visual modality alone [Summerfield 1992]. This suggests that research in the field of lip reading and AVSR has not completed its task and substantial progress can be expected in the future.

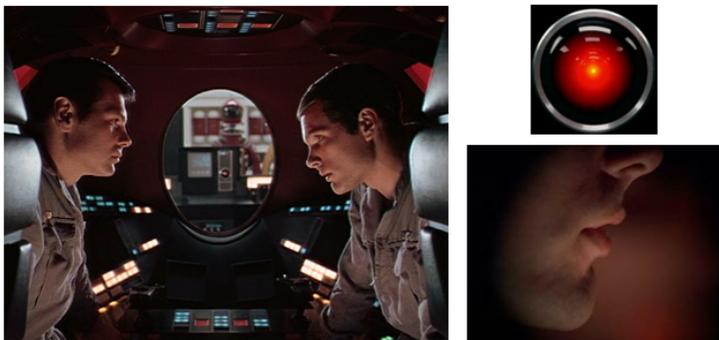


Figure 2.14: When will machines read lips using only visual information, even from profile view, as the villain computer HAL9000 did in Stanley Kubrick’s “2001: A Space Odyssey”?

3

Facial Expression Recognition from Video Sequences

The large number of subtle movements our faces can perform is one of the key components of human communication. Through facial expressions, we continuously transmit and read feelings and intentions (whether real or pretended) and support verbal communication. Moreover, facial deformations provide cues about a person's alertness, personality, generic health state, etc. The subject has fascinated many fields of research, last but not least computer science. Especially in the computer vision and machine learning communities, automatic facial expression recognition has long been advocated as a key feature of any interface aiming to be perceived as natural.

In his 1872's book *The Expression of the Emotions in Man and Animals* [Darwin 1872], Charles Darwin wrote:

...the young and the old people of widely different races, both with man and animals, express the same state of mind by the same movements.

Darwin was a precursor of a trend in psychology and anthropology started with Paul Ekman's studies conducted in the 1960's. Against the mainstream idea that facial expressions of emotions are fully learned and can thus differ among cultures, Ekman found that isolated primitive tribes perceive some facial expressions as connected with the same prototypical emotional states [Ekman and Friesen 1971].

Ekman's conclusion was that some particular facial movements correspond to specific emotions, independently of cultural background. These

“basic” emotions are *anger*, *disgust*, *fear*, *happiness*, *sadness*, and *surprise* (none of them with a clear social component, such as shame, or pride), exemplified in Figure 3.1. Such discrete and small set of emotions, even though far from being complete or capable of describing every day’s human feelings, has been an appealing choice for computer scientists ever since they started trying to design automatic methods for facial expression analysis.

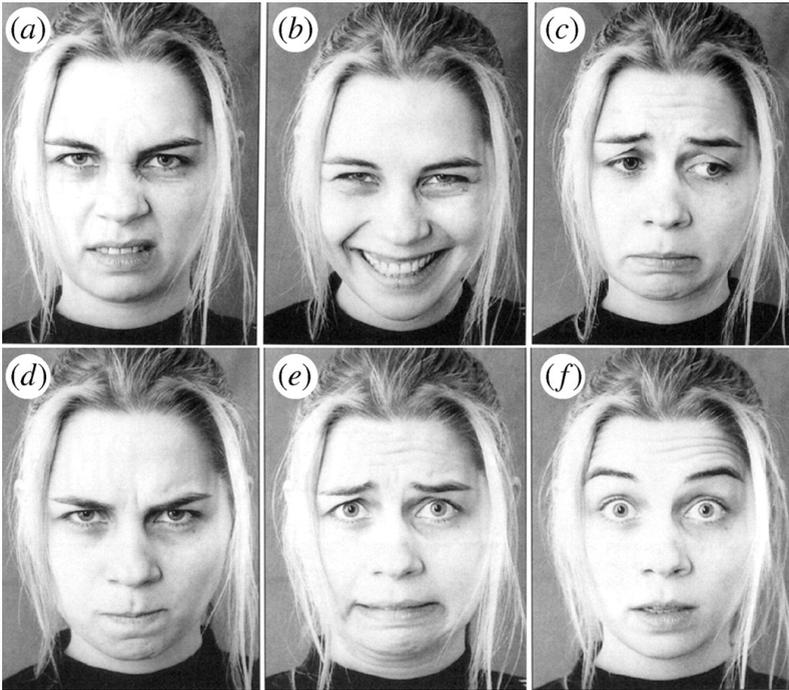


Figure 3.1: Facial deformations corresponding to the six basic emotions. (a-f): disgust, happiness, sadness, anger, fear and surprise. Figure reproduced with permission from [Pantic 2009]. ©The Royal Society

The ability for a computer to recognize the user’s facial expression opens a wide range of applications in different research areas, including security, marketing, medicine, education, telecommunications, and drowsy driver detection. However, it is important not to confuse human emotion recognition from facial expression recognition: The latter is merely a classification of facial deformations into a set of abstract classes, solely based on visual information. Instead, human emotions can only be inferred from context, self-report, physiological indicators, and expressive behavior which may or may not include facial expressions [Cohn 2006]. For example, a smile can appear both as an expression of joy and embarrassment [Ambadar *et al.* 2009].

According to the survey of [Fasel and Luetttin 2003], there are two main methodological approaches to the automatic analysis of facial expressions in the literature. *Judgment-based approaches* attempt to directly map visual inputs into one of a set of categories, while *sign-based approaches* describe facial expressions by means of coded facial actions, *e.g.*, Ekman’s Facial Action Coding System [Ekman and Friesen 1978]. FACS represents face deformations by activations of a set of Action Units corresponding to single facial muscles movements and it has inspired the facial animation parameters of the MPEG-4 standard [Pandzic and Forchheimer 2002].

This chapter presents a judgment-based method for the classification of videos of expressive faces into one of the basic emotion labels. We investigate the Hough transform voting approach of [Yao *et al.* 2010, Gall *et al.* 2011], originally designed for human action recognition, applied to the task of facial expression recognition. After having localized and normalized the faces with respect to the eyes’ centers, the image sequences are arranged into cuboids, or, extending the notation of [Yao *et al.* 2010], *expression tracks*. These are a representation of the face which is invariant to location, scale, and in-plane rotation. On such tracks, we perform classification by casting votes for the expression label and the temporal location of the apex, *i.e.*, the highest peak in the facial expression intensity.

As in [Yao *et al.* 2010, Gall *et al.* 2011], the voting is performed by a Hough forest (see Section 2.1.1), which learns a mapping from densely sampled spatio-temporal features to the center (apex) of the expression in the video sequence. The trees are trained in a multi-class fashion

and can therefore discriminate between different classes simultaneously. The leaf nodes can vote for each class and represent a discriminative codebook sharing features across classes.

Compared to the task of action recognition from video, facial expressions present subtler differences and are therefore more difficult to classify. Contributions in additions to the work of [Yao *et al.* 2010] include the normalization of the tracks with respect to rotation and the use of more discriminative features.

In the experiment section, we evaluate our system on standard databases of facial expressions. Our results are comparable to state-of-the-art methods, supporting our idea that Hough-voting approaches are promising tools for advancing in the field of automatic facial expression recognition.

3.1 Related work

Automatic facial expression recognition dates back to [Suwa *et al.* 1978]. Since then, the field of research has seen a steady growth, gaining momentum in the 1990's, thanks to the advances in algorithms for face detection and the availability of cheaper computing power, as the surveys of [Fasel and Luetttin 2003] and [Zeng *et al.* 2009] show. In this section we review some of the works forming a context for our proposed approach, pointing the interested reader to the recent publications of [De La Torre and Cohn 2011], [Tian *et al.* 2011], and [Valstar *et al.* 2011] for more information.

The initial face localization and normalization step, common to virtually all facial expression recognition approaches, serves to achieve a representation of the face invariant to scale, translation, and in-plane rotation. The literature is rich with approaches which normalize the images based on the location of the face [Buenaposada *et al.* 2008], of the eyes [Bartlett *et al.* 2005], or thanks to facial features tracking methods [Aleksic and Katsaggelos 2006, Dornaika and Davoine 2008], among which Active Appearance Models [Cootes *et al.* 2001, Matthews and Baker 2003] and 3D Morphable Models [Blaiz and Vetter 1999] stand out.

After the normalization stage, the remainder of an automatic facial expression recognizer consists of feature extraction, followed by the actual classification. Features are designed to minimize variation within the expression classes while maximizing it between different classes. Geometric measurements can be employed, *e.g.*, from the locations of specific points tracked on the face throughout the sequence [Shang and Chan 2009, Aleksic and Katsaggelos 2006]. Alternatively, image-based features can be extracted from texture patches covering either the whole face (holistic) or specific sub-regions of it (local). Commonly employed feature extraction methods from facial textures and their temporal variations include optical flow [Essa 1998, Yeasin *et al.* 2006], Gabor filter responses [Bartlett *et al.* 2005, Wu *et al.* 2010], and Linear Binary Patterns [Shan *et al.* 2009, Zhao and Pietikäinen 2009]. For the actual classification, AdaBoost and its combination with Support Vector Machines have recently gained a lot of attention [Bartlett *et al.* 2005, Littlewort *et al.* 2006]. Other popular approaches include nearest-neighbor searches [Buenaposada *et al.* 2008] and Hidden Markov Models [Cohen *et al.* 2003, Shang and Chan 2009, Zhao and Pietikäinen 2009, Aleksic and Katsaggelos 2006].

Most of the work in the literature has been concentrating on the analysis of posed expressions, usually classifying them into the six prototypical emotions or single Action Units. This is a consequence of the inherent difficulty of acquiring annotated databases of spontaneous facial expressions. However, there is a recent trend of works focusing on naturalistic expressions [Sebe *et al.* 2007], trying for example to discriminate authentic versus posed emotions [Pantic 2009].

Decision trees and forests have been previously used for action recognition, but only as indexing structures for speeding up nearest neighbor searches, as in [Lin *et al.* 2009, Reddy *et al.* 2009]. Works related to the Hough forest algorithm were presented in Section 2.1.1.

Inspired by the approach of [Yao *et al.* 2010] for human action classification, we build a holistic, image-based method for recognizing facial expressions which uses a random forest to learn the mapping between 3D video patches and votes in a Hough space for the label and the temporal location of the expression.

3.2 Voting Framework

Having seen the successful application of Hough forests (Section 2.1.1) to the task of human action recognition [Yao *et al.* 2010, Gall *et al.* 2011], we investigate their performance on facial expressions recognition. We assume our data to be already arranged into expression tracks, *i.e.*, the face images to be cropped and aligned as shown in Figure 3.2 (a). Section 3.3 provides insights on how this normalization is performed.

3.2.1 Training

We start from the assumption of having a set of training expression tracks available for each class $c \in C$, annotated with the expression label and the temporal location of the apex. We want to learn a mapping between 3-dimensional patches extracted from the expression tracks and a voting space for class label and time. To this end, we use the Hough forest method, originally developed for 2D single-class object detection [Gall and Lempitsky 2009], and later extended to handle multi-class detection in the spatio-temporal domain and applied to the task of action recognition [Yao *et al.* 2010, Gall *et al.* 2011].

Having already covered the Hough forest learning in Section 2.4.1, we now limit ourselves to the modifications needed in order to extend the algorithm to handle multiple classes and higher-dimensional input and output spaces.

We build a tree from a set of cuboids $\{\mathcal{P}_i = (\mathcal{I}_i, c_i, \mathbf{d}_i)\}$, randomly sampled from the training sequences. Figure 3.2 shows an expression track (a) and sample 3D patches extracted from it (b), with the corresponding displacement vectors. The features $\mathcal{I}_i = (I_i^1, I_i^2, \dots, I_i^F) \in \mathbb{R}^4$ are now in space plus time, and the expression label ($c_i \in C = \{0, 1, \dots, 5\}$) can represent one of the six basic emotions. Displacement vectors \mathbf{d}_i are correspondingly 3-dimensional, and stretch from the cuboid center to the center of the expression (apex) in the sequence.

Training the forest follows the same procedure explained in Section 2.4.1. The binary tests are still simple comparisons of two pixels, but this time localized both in space and time, *i.e.*, $\mathbf{p} \in \mathbb{R}^3$ and $\mathbf{q} \in \mathbb{R}^3$ in Eq. 2.4.

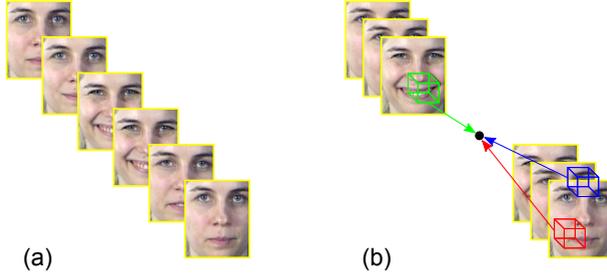


Figure 3.2: (a) Sample facial expression track. (b) Sample 3D patches drawn from the track, voting for the expression label and its spatio-temporal center.

The measures to minimize during training are readily modified for the multi-class problem at hand. For a set of cuboids S available at a node, the measure of class uncertainty equivalent to Equation 2.7 is

$$\mathcal{U}_1(S) = -|S| \cdot \sum_{c \in \mathcal{C}} p(c|S) \ln p(c|S), \quad (3.1)$$

and the spatial uncertainty measure equivalent to Equation 2.8 becomes

$$\mathcal{U}_2(S) = \sum_{c \in \mathcal{C}} \sum_{i: c_i=c} \|\mathbf{d}_i - \bar{\mathbf{d}}_c\|^2, \quad (3.2)$$

where $\bar{\mathbf{d}}_c$ is the average offset vector for class c .

When the training process is over, a leaf L stores a probability p_c for each expression class, approximated by the proportion of patches with class label c which ended in L during training. Moreover, for each class c , a leaf contains the training patches' respective displacement vectors, $D_c = \{\mathbf{d}_i\}_{i: c_i=c}$. Patches extracted from different classes can end up in the same leaf, thus sharing the same features; the probabilities p_c indicate the degree of sharing among classes.

3.2.2 Facial Expression Classification

At test time, similarly to the mouth localization case (Section 2.4.2), cuboids are densely extracted from the track being analyzed and sent through all trees in the forest. When reaching a leaf L , a patch casts votes in a 4D Hough accumulator (x and y location, time, and class label), proportional to the probabilities p_c stored at the leaf. For each class c , the corresponding vote is directed towards the expression spatio-temporal center, according to a 3D Gaussian Parzen window estimate of the vectors D_c .

Figure 3.3 exemplifies the voting process for a sequence expressing anger. The dark spots correspond to the probabilistic votes that have been cast by the patches and accumulated in the four-dimensional space. Because the track has already been localized in space, we marginalize the votes into a 2D accumulator for only class label and time. The local maximum in the remaining Hough image finally leads to the classification prediction, as displayed in Fig. 3.4. For a more formal description of the voting process, we refer the reader to [Gall *et al.* 2011].

Time-scale invariance can theoretically be achieved by up-sampling or down-sampling the tracks, and then applying the Hough forest to label expressions displayed at different speeds. However, the system has some tolerance built in through the variations in speed observed in the training data and we therefore did not consider multiple time scales.

3.3 Building the Expression Tracks

In order to arrange the data in the required normalized expression tracks, we align the faces based on the eye positions. Faces are rotated and scaled so that the eyes lie on the same horizontal line and present the same inter-ocular distance. The invariance to rotation, an addition to the work of [Yao *et al.* 2010], is a necessary step for the recognition of facial expressions, which are subtler and harder to recognize than human actions. When ground truth annotation of the eye locations is not available, we employ the automatic method described in Section 2.3. In the following, we normalize the facial images to an inter-ocular distance of 25 pixels, resulting in 55×45 images.

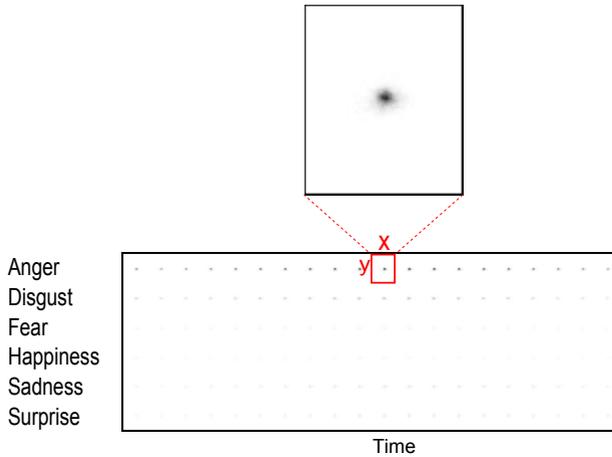


Figure 3.3: An example of the 4D Hough image (x , y , time, expression) output of the voting process for a clip displaying anger. The dark dots represent clusters of votes.

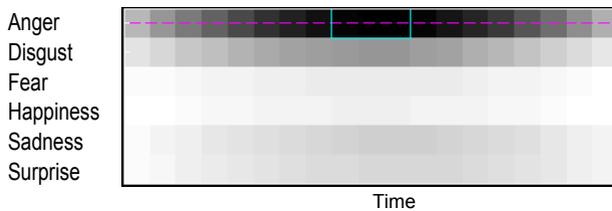


Figure 3.4: Example Hough voting space reduced to the two dimensions expression class and time. The maximum (in dark) is taken as the expression label and temporal location.

3.3.1 Feature Extraction

For the classification part of their work, [Yao *et al.* 2010] used simple features such as color, greyscale intensity, spatial gradients along the x and y axis, and frame to frame optical flow. In our approach, inspired by the work of [Schindler and Van Gool 2008], we extract more sophisticated features separately representing the shape and the motion of the face in the expression track.

The information about shape comes from the responses of a bank of log-Gabor filters. In comparison to standard (linear) filters, log-Gabor filters show an improved spectrum coverage with fewer scales [Field 1987]. The response g at position (x, y) and spatial frequency w is:

$$g^w(x, y) = \frac{1}{\mu} e^{-\frac{\log(w(x, y)/\mu)}{2 \log \sigma}}, \quad (3.3)$$

where μ is the preferred frequency and σ a constant used to achieve an even coverage of the spectrum. We use a bank with 3 scales ($\mu \in \{2, 4, 8\}$ pixels) and 6 orientations ($\phi \in \{0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ\}$), keeping only the response's magnitude $\|g^w(x, y)\|$ as descriptor. Example responses of the filters applied to one frame of an expression track are shown in Figure 3.5.

For the information regarding motion, we compute dense optic flow at every frame by template matching, using the L_1 -norm, considering 4 directions. Assuming that our expression tracks always start with a neutral face, we compute the optical flow with respect to both the previous frame (frame2frame) and the first frame in the track (frame2first). Figure 3.6 shows examples of the two types of optical flow fields extracted from one expression track.

In order to increase robustness to translation and reduce the dimensionality of the feature space, both the shape and motion feature images are down-sampled by max-pooling, also known as winner-takes-all [Fukushima 1980]:

$$h(x, y) = \max_{(i, j) \in \mathcal{G}(x, y)} [g(i, j)], \quad (3.4)$$

where $\mathcal{G}(x, y)$ denotes a 3×3 neighborhood of pixel (x, y) .

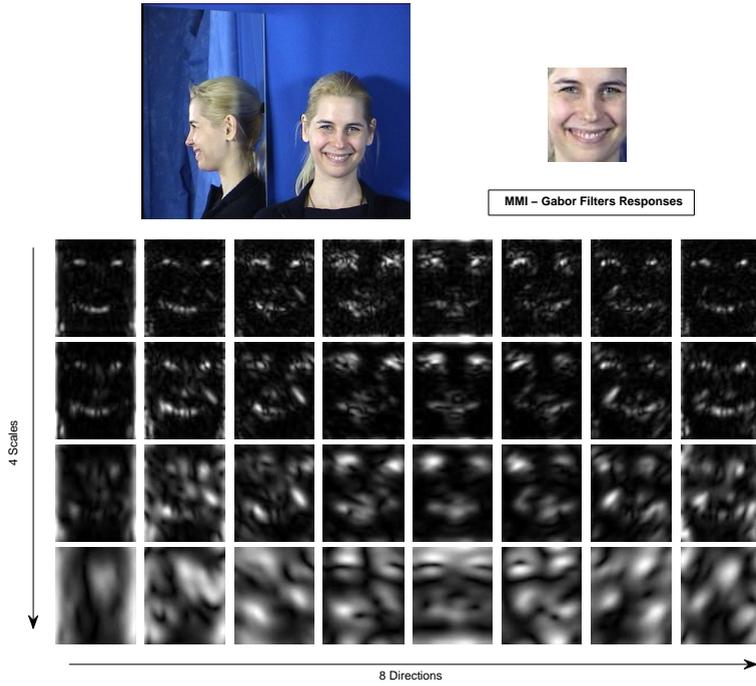


Figure 3.5: Example log-Gabor responses extracted from a normalized expressive face.

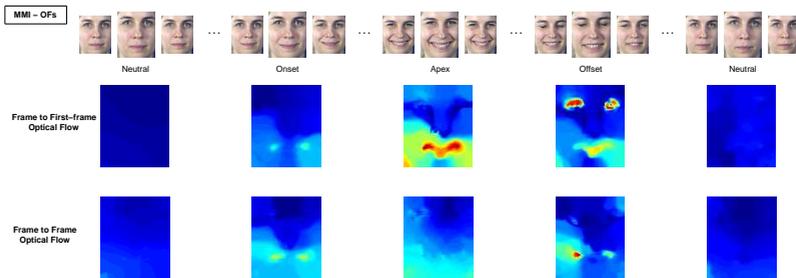


Figure 3.6: Example optical flow computed from an expression track.

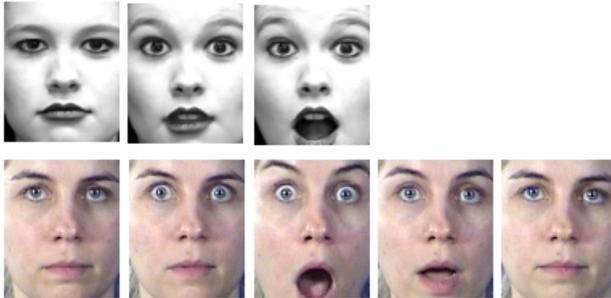


Figure 3.7: Sample frames extracted from sequences depicting surprise in the Cohn-Kanade database (top) and MMI database (bottom). Note how the MMI database contains not only the transition from the neutral face to the apex of the expression, but also the offset leading back to the neutral state at end of the sequence.

3.4 Experiments

We trained and tested our facial expression recognition system on the Cohn-Kanade [Kanade *et al.* 2000] and the MMI [Pantic *et al.* 2005, Valstar and Pantic 2010] databases. Both contain videos of posed facial expressions, with subjects facing the camera under controlled lighting conditions.

The Cohn-Kanade database consists of greyscale video sequences of 100 university students, 65% of which female. The videos always start with a neutral face and end at the apex, *i.e.*, the maximum intensity of the expression. For our study, we selected sequences which can be labeled as one of the basic emotions and which are longer than 13 frames, for a total of 344 videos of 97 subjects, each performing 1 to 6 facial expressions. Recently, the same authors collected a new database, the Extended Cohn-Kanade Dataset, containing more sequences and some examples of genuine smiles [Lucey *et al.* 2010].

The MMI database [Pantic *et al.* 2005, Valstar and Pantic 2010] is a constantly growing, web-searchable set of color videos containing both posed and spontaneous emotions. We selected the subset of (posed)

videos labeled as one of the six basic emotions, while discarding all others labeled only in terms of Action Units. The resulting set is comprised of 176 videos of 29 people displaying 1 to 6 expressions. The subjects differ in sex, age, and ethnic background; moreover, facial hair and glasses are sometimes present. The main difference between the MMI and Cohn-Kanade databases is that the MMI sequences do not end at the expression’s apex, but return to a neutral face. Example sequences of surprise from both datasets are shown in Figure 3.7, with the Cohn-Kanade at the top and MMI database at the bottom.

As explained in Section 3.3, both databases have been aligned to the eye center locations. For the Cohn-Kanade database, ground truth manual annotations are provided by [Lipori 2010], while no such labeling is available for the MMI database, on which we used the eye tracking method described in Section 2.3.

Expression tracks need to be labeled with both spatial and temporal center of the expression. The center in the image plane is assumed to correspond to the center of the face. The temporal center should ideally be located at the expression apex, therefore we took the last frame for the Cohn-Kanade database and the middle frame in the case of the MMI database. We trained and tested on all frames from the Cohn-Kanade dataset, which has an average sequence length of 18 frames, while we selected only 20 frames in the middle of each sequence for the MMI database, which has an average length of 79 frames.

For all of the following experiments, we performed subject-independent 5-fold cross validations, *i.e.*, making sure that the same subjects did not occur in both training and test sets, and present here the results averaged over all five iterations. Forests always contained only 5 trees; indeed, adding more trees improved the results only slightly.

Among the parameters of the proposed method are the size and shape of the 3D patches. We ran some experiments varying the patches’ spatial size and shape, while keeping their number fixed to 100 and the temporal dimension to 2 frames. In Figure 3.8, the bars represent the recognition rate as a function of the size and shape of the sampled patches, as achieved on the Cohn-Kanade database. Larger patches produced better results than smaller ones. The best results (86.7%) were achieved with 20×50 patches, *i.e.*, vertical rectangles covering almost half of the face.

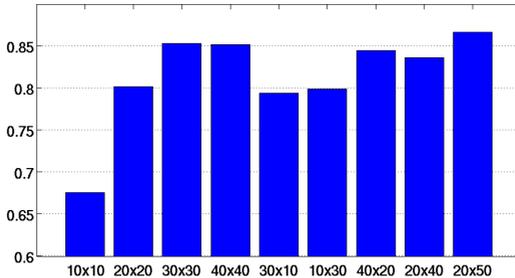


Figure 3.8: Influence of the patch size on the overall recognition rate. Larger, rectangular patches, give the best results.

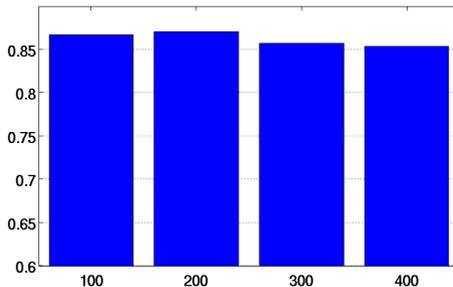


Figure 3.9: Recognition accuracy as a function of the number of (20×50) patches sampled from each sequence during training.

Increasing the number of training patches per sequence did not influence much the recognition accuracy. Figure 3.9 shows that the accuracy increased slightly only when moving from 100 to 200 patches, while it actually decreased when more patches were used. We also tested the influence of the temporal length of the patches, but did not experience significant changes in the expression recognition accuracy. All results shown in the rest of the section were achieved by sampling 200 patches of size $20 \times 50 \times 2$.

Figure 3.10 shows the confusion matrix obtained by our method when applied to the Cohn-Kanade dataset. On average, we recognized the

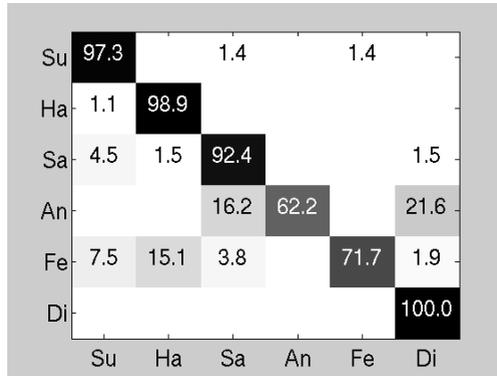


Figure 3.10: Confusion matrix for the Cohn-Kanade database, normalized using the provided ground truth eye locations. Expressions such as disgust and surprise are well recognized, while most of the confusion arises from the anger/disgust and fear/happiness classes.

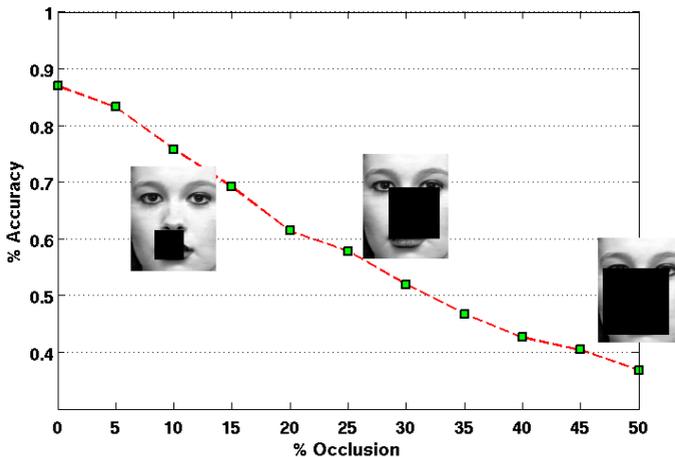


Figure 3.11: Recognition rate for the Cohn-Kanade database, as a function of the percentage of occlusion. The images along the curve help in understanding the amount of occlusion introduced. Even when 50% of the face is deleted, the system still performs better than chance (16.7%), close to 40% recognition rate.

correct expression 87.1% of the time; in particular, disgust was always correctly classified. Fear and anger were the most confused labels, mainly mistaken for happiness, respectively disgust.

To assess the robustness of the method to corrupted test images, we removed (set to zero) the information in each feature channel falling under a cuboid. The cuboids were as long as the sequences, and covered a specific percentage of the image plane. For each sequence, the cuboid location on the 2D image plane was randomly chosen. We ran 5 trials for each percentage of occlusion, and present the averaged results in Figure 3.11. It can be noted how the performance decreases slowly as the amount of missing data. At 15% occlusion, the accuracy is still around 70%, falling below 50% only when more than 30% of the face is removed. Sample frames help visualizing the amount of facial image removed.

	HF	[Yeasin06]	[Buenaposada08]	[Aleksic05]
SUR	97.3%	100%	100%	100%
HAP	98.9%	96.6%	98.8%	98.4%
SAD	92.4%	96.2%	82.0%	96.2%
ANG	62.2%	100%	78.4%	70.6%
FEA	71.7%	76.4%	73.9%	88.2%
DIS	100%	62.5%	87.9%	97.3%
AVG	87.1%	90.9%	89.1%	93.6%

Table 3.1: *The results of our method (HF) are comparable with other works on automatic expression recognition. The accuracy is given for each expression class separately and on average.*

Table 3.1 lists the results of our Hough forest-based algorithm (HF) next to the performance of other methods which used the Cohn-Kanade database and which published their recognition rates for each label. Results are comparable.

In an attempt to assess the contribution of each feature channel to the recognition, Figure 3.12 plots the accuracy achieved on the Cohn-Kanade database (using the ground truth annotations for alignment) when each feature was used separately and in all their possible combinations. As can be seen, frame to first optical flow alone gave the best results, followed by

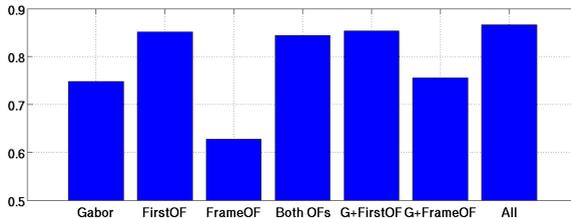


Figure 3.12: Average recognition accuracy on the Cohn-Kanade database plotted against single features and their combinations. The optical flow between the current and the first frame gave the best results, followed by Gabor filter responses and frame to frame optical flow.

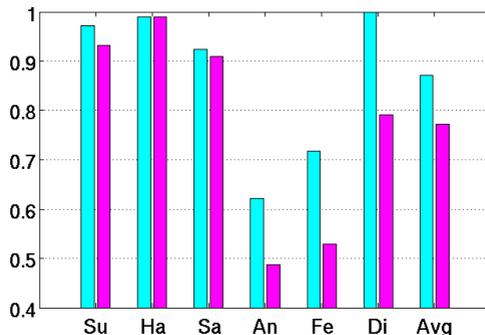


Figure 3.13: Accuracy for each class label, as recognized from the tracks created thanks to the ground truth annotation (cyan bars on the left) and automatically extracted by the eye tracker (magenta, right).

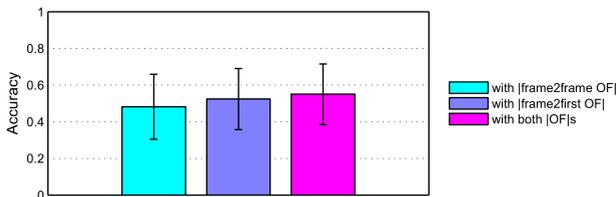


Figure 3.14: Results obtained on the MMI database using the set of features originally employed by [Yao et al. 2010] with the addition of frame to frame and frame to first optical flow.

log-Gabor responses and by optical flow computed between consecutive frames. The combination of all three features led to the best results, though for a small amount.

Su	84.6				15.4	
Ha		97.1				2.9
Sa			73.3	13.3	6.7	6.7
An		3.0	21.2	69.7		6.1
Fe	25.9	3.7	14.8	7.4	44.4	3.7
Di		7.4	3.7	7.4	3.7	77.8
	Su	Ha	Sa	An	Fe	Di

Figure 3.15: Confusion matrix for the MMI database. The higher rate of confusion with respect to the results obtained on the Cohn-Kanade database can be partly explained by the fact that manual annotations of the eye locations were not available.

In Figure 3.13, the performance for each class is plotted, depending on whether the tracks were extracted using the manual ground truth annotations of the eye locations (cyan bars on the left) or automatically, using the eye tracker described in Section 2.3 (magenta, on the right). Results clearly worsened when the fully automatic method was employed, but not in the same extent for each class. Surprise, happiness, and sadness were less affected by errors in the tracking than the other classes.

When training and testing on the MMI database, again in a 5-fold cross-validation fashion and with 200 patches of size $20 \times 50 \times 2$, we obtained the confusion matrix shown in Figure 3.15. We notice a higher rate of misclassification compared to the results achieved on the Cohn-Kanade database, especially for fear. This could be partly explained by the fact that manual annotations of the eye centers were not available, but also by the lack of a precise annotation of the expression center in the sequences. Also, the expressions in the MMI database are subtler than in the Cohn-Kanade dataset. On average, our method achieved a recognition rate of 76% on the MMI database and, to the best of our knowledge, we are the

first ones to attempt at classifying the expressions directly (rather than Action Units) on this dataset.

Figure 3.14 shows the average results obtained on the MMI sequences when using the features originally proposed by [Yao *et al.* 2010], with the addition of the two kinds of optical flow. The poor results of the original feature set serves as convincing support for the introduction of the log-Gabor filter responses, as explained in section 3.3.1

3.5 Conclusion

In this chapter, we investigated the use of a Hough voting method for facial expression recognition. Our system extends previous work aimed at human action classification [Gall *et al.* 2011] to the task of discriminating facial expressions from video sequences, which are subtler and harder to classify.

We chose features which separately encode the form and motion of the face, allowing us to capture the subtle differences in the facial expressions which the original action recognition system could not. We evaluated the system on two standard databases achieving results comparable to the state of the art. A valuable feature of Hough voting methods is their robustness to occlusions, property which we experimentally demonstrated by synthetically removing parts of the image from the testing sequences.

As for the mouth localization method of Chapter 2, the main limitation of the proposed system is its current dependence on eye tracking for the normalization of the expression tracks. This makes our system unusable when the eyes are not visible. However, robust tracking of a larger number of facial features might be possible, reducing the sensitivity of the algorithm to partial occlusions of the face. Such improved tracking would also allow the application of our Hough voting method to the recognition of the expression from non frontal facial videos.

Future work includes the investigation of additional image features and the application of the method to the recognition of more naturalistic facial expressions or of Action Units activation. Because the system appears to perform better when using large patches (see Figure 3.8), it would be interesting to try and directly define the binary tests of the

forest over the whole normalized face region. In order to be able to produce a large number of votes, forests would have to contain more trees. Even though this should not hinder processing time, memory requirements would be higher.

4

Acquisition of a Multimodal Corpus of Affective Communication

As computers become ever more ubiquitous tools in our everyday lives, increasing efforts go into technologies aiming to improve our experience of the interaction with them. Emotions play a crucial role in human cognition [Damasio 1995] and artificial agents will never be perceived as fair interlocutors unless they can read and express feelings, similarly to how we do it. For these reasons, the field of affective computing has seen a boost in recent years [Zeng *et al.* 2009]. Algorithms for both recognition and synthesis of emotional states are being designed and developed; however, what is often missing are annotated corpora displaying affective communication, needed for training and evaluating such systems. Acquiring such datasets is challenging, as human affective displays are multimodal, rare, often masked in real life interactions, and highly context- and culture-related.

When designing the acquisition of a new corpus, a first question to be addressed is which modalities should be captured. The research community has converged towards the idea that affect-aware interfaces should use several modalities, in a way imitating humans [Jaimes and Sebe 2007, Sebe *et al.* 2006, Sebe 2009, Zeng *et al.* 2007]. Even though emotional cues can be rather reliably extracted from physiological measurements (e.g. [Picard 2000]), the invasiveness of these methods can influence the subject's state and rule out most application scenarios. Humans, on the other hand, can easily guess someone's affective state only from cues

such as facial expressions, voice modulation, and body pose. Speech and facial expression, in particular, appear to encode most of the information used by humans to communicate emotions [Mehrabian 1968] and therefore have been often preferred in the computer science community [Zeng *et al.* 2009, Sebe *et al.* 2006].

Another important aspect to be taken into consideration when acquiring affective data is the desired degree of naturalness. A good trade-off between quality and naturalness needs to be found: Corpora collected in controlled environments are by definition unnatural, but moving towards unconstrained settings increases the amount of noise, making the data unusable for many applications such as visual speech synthesis [Mueller *et al.* 2005, Zhang *et al.* 2004, Edge *et al.* 2009, Deng and Neumann 2007]. Many studies on affective computing concentrated so far on posed data, which is proven to differ from spontaneous behavior [Whissell 1972, Frigo 2006, Valstar *et al.* 2007]. An affective-aware agent will be of no use, or even harm, if it can recognize uninteresting expressions but fails to identify key emotions which we instead can easily spot. In cases where the accuracy of the data is crucial, *e.g.*, for computer graphics purposes, induction methods represent a good compromise, and the literature is rich of examples where videos [Gross and Levenson 1995], still photographs [Bradley *et al.* 1996], music [Clark 1983], or manipulated games [Scherer *et al.* 1998] have been used to elicit emotions. These methods are far from being a replacement of pure naturalism, but they are well established and have shown to evoke a range of authentic emotions in a laboratory environment [Cowie and Cornelius 2003], in particular the use of videos [Westermann *et al.* 1996].

The evaluation and annotation of the recorded data requires a definition of emotion, which is still an open issue in itself. Quoting from [Izard 2009]:

The term “emotion” has defied definition mainly because it is multifaceted and not a unitary phenomenon or process. Use of the unqualified term “emotion” makes for misunderstandings, contradictions, and confusions in theory and research.

Many of the works on affective computing so far are limited to the six basic emotions of Ekman [Ekman 1971]. These few discrete categories actually stand for a family of emotions, bounded to Ekman's stringent criteria [Cowie *et al.* 2002]. Moreover, everyday experience suggests that emotions do come in combinations, common examples being a sad versus a joyful surprise. A popular alternative are continuous representations where affective states are mapped onto a low-dimensional space, *e.g.*, a 2D space based on activation or arousal (strength) and evaluation or valence (positive vs. negative) [Russell 1980, Craggs and Wood 2004]. A more complex wheel of emotions was suggested in [Plutchik 2001], which consisted of 8 basic bipolar emotions (joy versus sorrow, anger versus fear, acceptance versus disgust and surprise versus expectancy) and 8 advanced emotions each composed of 2 basic ones. In general, collapsing the multidimensional space of possible emotional states onto a homogeneous, low-dimensional space inevitably incurs in information loss, and different ways of performing the collapse lead to different results. Such representations are also not intuitive and difficult to use for inexperienced users.

In this chapter, we present a framework for collecting and annotating a novel multimodal corpus aimed at the research fields of automatic synthesis and recognition of expressive verbal communication. Together with speech, we acquire high-quality dense dynamic 3D facial geometries. The 3D information is highly desirable in the mentioned research fields for its informative power, allowing to extract features more easily and reliably than 2D video.

Because of the necessary recording setup, we settle for elicited emotions and resort to video clips to induce affective states, as it was done, among others, in [Sebe *et al.* 2007]. While the video clips provide a context in the spirit of film-based induction methods, the repetition of the emotional sentences serves in itself as an eliciting method [Velten 1968]. We also introduce a consistency check by asking our speakers to evaluate the emotion in the video clip. Similarly to [Morlec *et al.* 2001], we label the corpus using a list of affective adjectives to be weighted according to their perceived strength, allowing multiple labels for each sentence. Both the eliciting videos and the recorded data were evaluated by independent online surveys.

The resulting BIWI 3D Audiovisual Corpus of Affective Communication, $B3D(AC)^2$, is valuable for applications like emotional visual speech modeling, but also for view-independent facial expression recognition, or audio-visual emotion recognition. The corpus is made available to the community for research purposes ¹.

4.1 Related Work

One way to categorize databases for training and evaluating affection-aware systems is based on whether the recorded emotions are naturalistic, artificially induced, or fully posed. A comprehensive overview of the existing audio-visual corpora can be obtained from [Cowie *et al.* 2005], [Douglas-Cowie *et al.* 2007], and [Zeng *et al.* 2009]. In this section, we list some of the available datasets which can be related to ours, with a specific focus on affective communication.

The HUMAINE Network of Excellence has produced important steps forward in the field of affective computing, gathering a collection of databases [Douglas-Cowie *et al.* 2007] containing a large number of audio-visual recordings, divided into naturalistic and elicited.

Among the naturalistic databases, the Vera am Mittag dataset [Grimm *et al.* 2008] consists of recordings from a German TV talk show, containing spontaneous emotional speech coming from authentic discussions. Most of the data was labeled by a large number of human evaluators using a continuous scale for three emotion primitives: valence, activation, and dominance. The Belfast naturalistic database [Douglas-Cowie *et al.* 2003] contains TV recordings and interviews judged relatively emotional, annotated using the FEELTRACE [Cowie and Cornelius 2003] system. The EmoTV corpus [Martin *et al.* 2009] contains interactions extracted from French TV interviews, both outdoor and indoor, with a wide range of body postures.

Moving on to elicited datasets, the Sensitive Artificial Listener (SAL) database [Douglas-Cowie *et al.* 2007] contains audio-visual recordings of humans conversing with a computer. The SAL interface is designed to let the user work through a range of emotional states. The SmartKom

¹www.vision.ee.ethz.ch/datasets/

database [Türk 2001], comprises recordings of people interacting with a machine asking them to solve specific tasks provoking different affective states. In the Activity Data and Spaghetti Data sets [Douglas-Cowie *et al.* 2007], volunteers were recorded while respectively engaging in outdoor activities and feeling inside boxes containing various objects (*e.g.*, spaghetti or buzzers going off when touched). The subjects recorded the emotions they felt during the activities. The eWiz database [Morlec *et al.* 2001] contains 322 sentences pronounced by the same speaker with varying prosodic attitudes suggested by reading a text specifying the affective context. In [Zara *et al.* 2007], the EmoTaboo protocol is introduced, consisting in letting pairs of people (one being a confederate) play the game “Taboo” while their faces, upper bodies, and voices are recorded.

The work of [Sun *et al.* 2011] recently introduced a multimodal database focused on mimicry and its relationship with human affect. The 40 participants were recorded with 18 synchronized audio and video sensors while engaging in a political discussion and in a role-playing game, always in pair with a confederate. The corpus is annotated with dialogue acts, turn-takings, affect, body movements and facial expressions; additionally, the participants self-reported their felt experiences. The MAHNOB-HCI database [Soleymani *et al.* 2012] was recorded in response to affective stimuli. The database contains synchronized recordings of face videos, audio, and physiological signals coming from the peripheral and central nervous system (ECG, GSR, respiration amplitude and skin temperature). The 27 participants watched eliciting emotional videos, self-reporting their emotions using arousal, valence, dominance, predictability, as well as affective keywords.

Going towards acted corpora, the GEMEP corpus [Bänziger and Scherer 2007] comprises recordings of the voices, faces, and full bodies of professional stage actors uttering meaningless sentences, following the method of [Banse and Scherer 1996]. The set of displayed emotions is an extension of the six basic ones, and the actors were guided by reading introductory scenarios for each emotion. In [Chen 2000], students were filmed while pronouncing a set of sentences, each representing one of eleven affective states, once again an extension of the six basic emotions.

Annotating video recordings is difficult and time consuming. For example, the popular Facial Action Coding System (FACS) labeling [Ek-

man and Friesen 1978] takes a trained expert about two hours for one minute of video footage. An alternative are marker-based motion capture systems, used to obtain 3D information. An example use of motion capture is the IEMOCAP database [Busso *et al.* 2008], where actors were recorded in dyadic sessions with markers on the face, head, and hands while performing affective communication scenarios. Motion capture techniques were also employed to record actors engaged in affective speech for corpora aimed at visual speech modeling for synthesis purposes, as in [Cao *et al.* 2005], [Beskow and Nordenberg 2005], and [Wampler *et al.* 2007]. Despite the accuracy and robustness of such methods, placing markers on someone’s face is error prone and might influence the subject’s emotional state like other invasive physiological measurements.

When dense 3D face geometry data is desired, most of the available datasets only target face recognition and therefore contain only still scans of neutral faces. Exceptions are the databases of [Yin *et al.* 2006] and [Yin *et al.* 2008], where a large number of subjects were recorded by a 3D scanner while posing the six basic emotions (without speech), only the latter containing dynamics. The Bosphorus 3D face database [Savran *et al.* 2008] also includes facial expressions, composed of selected subsets of Action Units and the six basic emotions. The recently proposed 3D Relightable Facial Expression (ICT-3DRFE) database of [Stratou *et al.* 2011] focuses on illumination invariance: 3D models of 23 subjects performing 15 different expressions (FACS annotated) come together with photometric information allowing for photo realistic rendering. Also the database of [Cosker *et al.* 2011] is FACS annotated and contains dynamic 3D facial expressions of 10 subjects performing between 19 and 97 different AUs, both individually and in combination.

To the best of our knowledge, our corpus represents the first dataset combining audio and dense, dynamic 3D facial deformations of affective communication.

4.2 Data Acquisition

In order to simultaneously record audio-visual speech data, we employed the real-time 3D scanner described in [Weise *et al.* 2007] and a studio condenser microphone. To keep the noise level as low as possible, we acquired the data in an anechoic room, with walls covered by sound wave-absorbing materials. Figure 4.1 shows the setup, with a speaker being scanned while watching an eliciting video on the screen.

4.2.1 Corpus Definition

Our database consists of 40 short English sentences extracted from feature films. The clips were selected by the authors, trying to cover a wide range of emotions and ensuring that the speech was clear, without music or other voices in the background. The movie clips do not just contain the sentence to be pronounced, but are longer (about 30 seconds on average) and are supposed to build the emotional state in the viewer.

Our volunteers satisfied the sole requirement of being native English speakers: a total of 14 subjects, 8 females and 6 males, aged between 21 and 53 (average 33.5); example identities are shown in Figure 4.2. Each sentence was recorded twice: with and without emotion. After removing some wrong recordings, we got 1109 sequences, 4.67 seconds long on average.

4.2.2 Recording Protocol

Each speaker sat alone in the anechoic chamber, in front of the scanner and the microphone, while the authors could give instructions and control the recordings from a separate room.

For the first part of the corpus, the speaker was asked to read the sentences from text displayed on a computer screen, keeping a neutral tone. In a second stage, the speaker watched the eliciting video and was asked to rate its emotional content by means of a paper questionnaire, as explained in Section 4.3.1. The videos could be seen more than once if requested. In order to capture the emotional version of each sentence,



Figure 4.1: Recording setup: one speaker sits in front of the 3D scanner in the anechoic room while watching one of the eliciting videos clips.

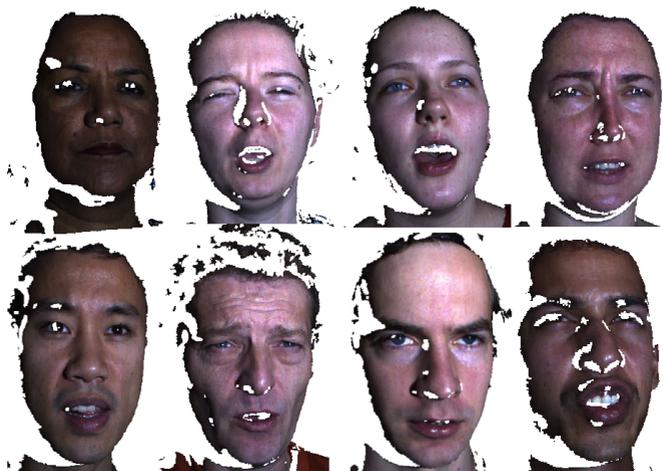


Figure 4.2: Examples of the different identities present in the corpus.

the speaker was finally asked to repeat the sentence using the emotional tone perceived from the video.

4.2.3 Video Processing

The real-time 3D scanner was employed to capture detailed 3D geometry and texture of the performances of each speaker, as shown by the first two images in figure 4.3. Facial expression analysis, however, requires full spatial and temporal correspondences of the 3D data. To achieve this goal, we used the two-step procedure introduced in [Weise *et al.* 2009a]: First, a generic template mesh is warped to the reconstructed 3D model of the neutral facial expression of a speaker. Second, the resulting personalized template is automatically tracked throughout all facial expression sequences.



Figure 4.3: From left to right, the image shows the 3D reconstruction of a person’s face, the corresponding texture mapped on it, and the personalized face template deformed to fit the specific frame.

Personalized Face Template

In order to build a person-specific face template, each speaker was asked to turn the head with a neutral expression and as rigidly as possible in front of the real time 3D scanner. The sequence of scans was registered and integrated into one 3D model using the online modeling algorithm proposed in [Weise *et al.* 2009b]. Small deformations arising during head

motion violate the rigidity assumption, but in practice do not pose problems for the rigid reconstruction. Instead of using the 3D model directly as a personalized face template, a generic face template was warped to fit the reconstructed model. Besides enabling a hole-free reconstruction and a consistent parametrization, using the same generic template has the additional benefit of providing full spatial correspondence between different speakers.

Warping the template to the reconstructed 3D models was achieved by means of non-rigid registration, where for each mesh vertex \mathbf{v}_i of the generic template a deformation vector \mathbf{d}_i is determined in addition to a global rigid alignment. This is formulated as an optimization problem, consisting of a smoothness term minimizing bending of the underlying deformation [Botsch and Sorkine 2008], and a set of data constraints minimizing the distance between the warped template and the reconstructed model. As the vertex correspondences between generic template and reconstructed model are unknown, closest point correspondences are used as approximation similarly to standard rigid ICP registration. A set of manually labeled correspondences were used for the initial global alignment and to start the warping procedure. The landmarks were mostly concentrated around the eyes and mouth, but a few correspondences were selected on the chin and forehead to match the global shape. The manual labeling was necessary only once per speaker and took at most a couple of minutes. The resulting personalized template accurately captures the facial 3D geometry of the corresponding person.

The diffuse texture map of the personalized template was automatically extracted from the rigid registration scans by averaging the input textures. The face was primarily illuminated by the 3D scanner, and we could therefore compensate for lighting variations using the calibrated position of the projection. Surface parts likely to be specular were removed based on the half angle. The reconstructed texture map is typically over smoothed, but sufficient for the tracking stage.

Facial Expression Tracking

The personalized face templates were used to track the facial deformations of each performance. For this purpose, non-rigid registration was

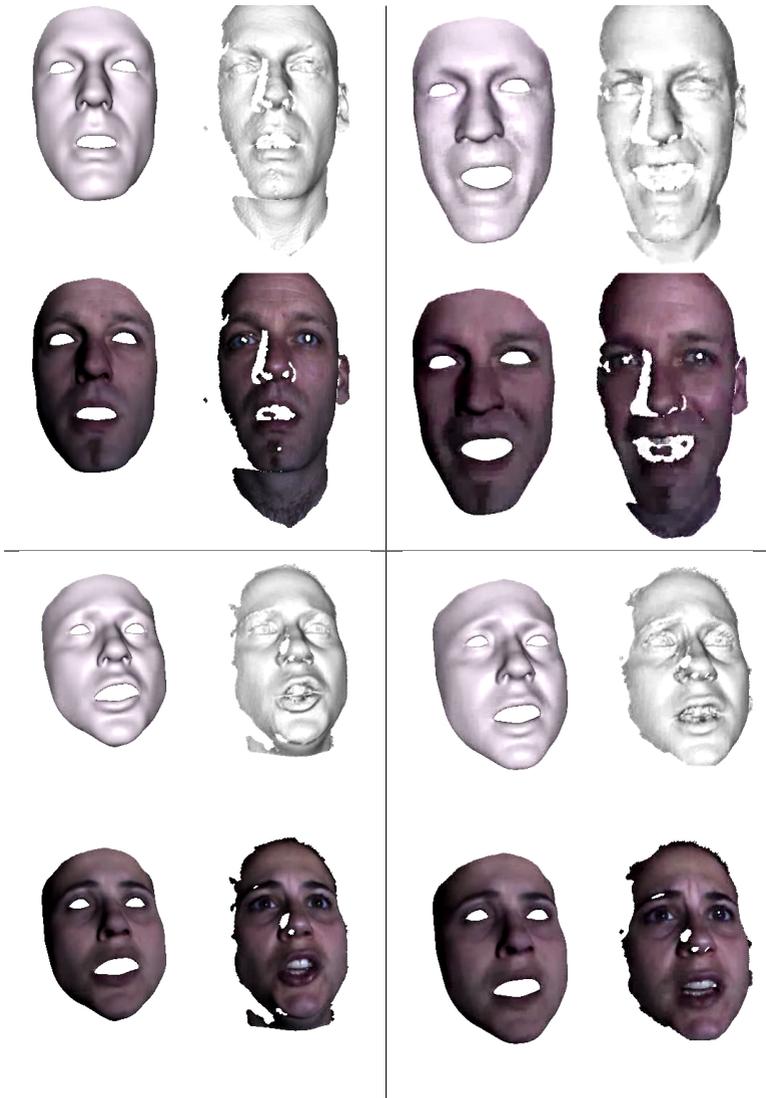


Figure 4.4: Example frames from the database, with and without texture. The recorded data is shown on the right, while the deformed generic templates are rendered on the left.

employed, in a similar manner as during the template creation, minimizing the distances between the template vertices and the 3D scans. To ensure temporal continuity, optical flow constraints were also included in the optimization as the motion of each vertex from frame to frame should coincide with the optical flow constraints. During speaking, the mouth region deforms particularly quickly, and non-rigid registration may drift and ultimately fail. This was compensated for by employing additional face-specific constraints such as explicitly tracking the chin and mouth regions, making the whole process more accurate and robust to fast deformations. Figure 4.4 shows visual examples of the corpus where the personalized models are adapted to specific frames.

4.2.4 Audio Processing

Different affective states are manifested in speech by changes in the prosody, see [Schröder 2008] for an overview. As certain prosodic differences are small but still audible, a careful setup of an audio-visual corpus requires accurate extraction of prosodic parameters from the audio signal.

Speech prosody can be described at the perceptual level in terms of pitch, sentence melody, speech rhythm, and loudness. The physically measurable quantities of a speech signal are the following acoustic parameters: fundamental frequency (F_0), segment duration, and signal intensity. F_0 correlates with pitch and sentence melody, segment duration correlates with speech rhythm, and signal intensity with loudness.

The annotation process necessary for obtaining the physical prosodic parameters of the utterances in the corpus included a number of steps: First, the sentence's text was transcribed into the phonological representation of the utterance. Then accurate phoneme segmentation, fundamental frequency extraction, and signal intensity estimation were achieved by analyzing the speech data. In the following, we give an overview of the extraction of fundamental frequency, signal intensity, and segment duration, for which we used the automatic procedures provided by SYNVO Ltd [Synvo 2012].

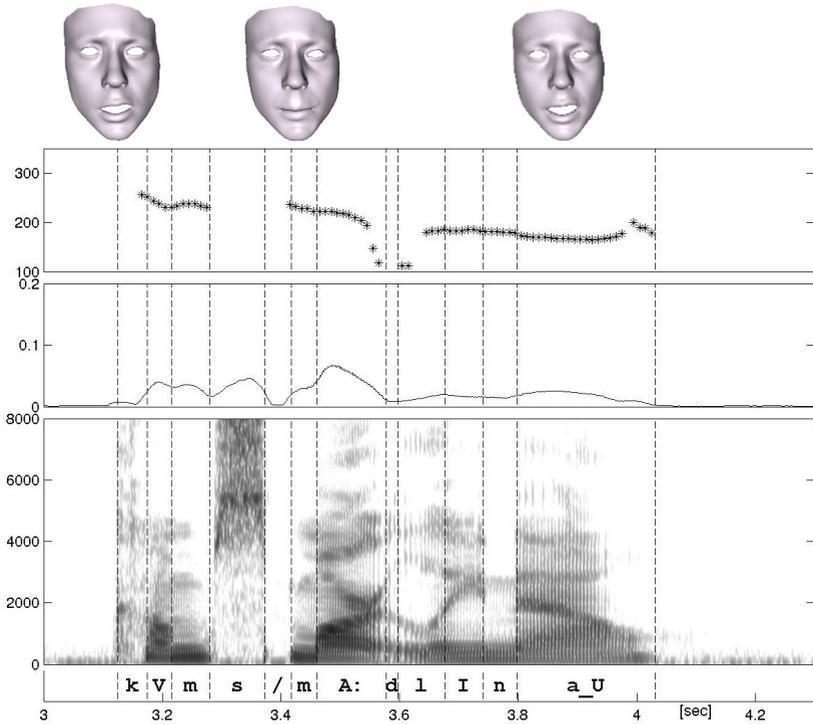


Figure 4.5: A sentence of the corpus (“Come smartly now”), neutrally pronounced. Phoneme segmentation, spectrogram, signal intensity contour, fundamental frequency contour of the speech signal, and sample faces are shown from bottom to top.

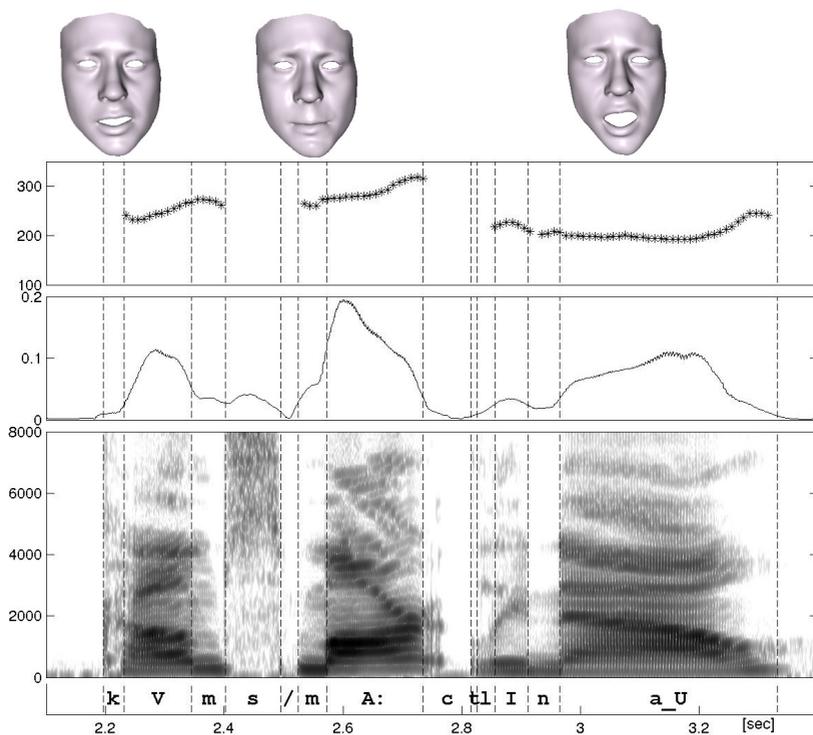


Figure 4.6: Same sentence as Figure 4.5, pronounced by the same speaker, but after having seen the eliciting video. The emotional utterance clearly shows higher signal intensity and a high rising fundamental frequency contour at the second syllable, in contrast to the low falling one of the neutral utterance shown in Figure 4.5. Syllable nucleus durations also appear longer in the emotional version of the sentence. Also the 3D faces at the top suggest higher emotional content.

Transcription

The phonological representation contains the sequence of phonemes for the sentences in the corpus, the syllables' stress level, the position and strength of phrase boundaries, plus the indicators of phrase types. Initial phonological representations of the sentences were obtained from the text version of the corpus, thanks to the transcription component of the text-to-speech system described in [Romsdorfer and Pfister 2007]. These initial phonological representations contain the standard phonetic transcription of the sentences, or canonical phonetic transcription.

The phonological part (phrase type, phrase boundary, and sentence accentuation) of the automatically generated representations was then adapted to the speech signals. Neural network-based algorithms were employed for automatic phrase type, phrase boundary, and syllable accent identification [Romsdorfer 2009].

vowels	i: I U u: e @ q 3 3: V O: A A: Q
diphthongs	@_U a_I a_U e_I E_@ I_@ O_I o_U U_@
consonants	p p_h b t t_h d k k_h g m n N r f v T D s z S Z x h j w l
affricates	t_S d_Z
pauses	c_u c_v /

Table 4.1: Segment types of English phonemes and speech pauses used for transcription of the speech data of the audio-visual corpus.

Fundamental Frequency Extraction

Fundamental frequency (F_0) values of the speech data were computed every 10 ms using a pitch detection algorithm based on combined information taken from the cepstrogram, the spectrogram, and the autocorrelation function of the speech signal [Romsdorfer 2009]. Signal sections judged as unvoiced by the algorithm were assigned no F_0 values. Figures 4.5 and 4.6 show examples of such fundamental frequency contours.

Signal Intensity Extraction

Signal intensity values of the speech are computed every 1 ms. We used the root mean square value of the signal amplitude calculated over a window of 30 ms duration. Signal intensity contours of the same sentence pronounced by the same speaker in neutral and emotional mode are displayed in figures 4.5, respectively 4.6. The intensity contours of the two utterances suggest the difference in their emotional content.

Segment Duration Extraction

An accurate extraction of phoneme and speech pause durations requires an exact segmentation of the speech into adjacent, non-overlapping segments, and a correct assignment of labels to these segments indicating the segment type. This assignment is commonly termed “labeling”.

Because the phonological representation contains the standard phonetic transcription of an utterance, it is convenient to use this standard transcription for automatic segmentation and labeling. However, a close phonetic transcription, also referred to as matched phonetic transcription, indicating pronunciation variants made by the speaker, results in a much better segmentation and labeling.

Segment Types

Segment types correspond to the phoneme types determined in the transcription. Plosives were additionally segmented into their hold and burst parts, which were labeled separately. While the burst part of a plosive

was denoted by the same symbol used for the plosive phoneme type, a “c” denoted the hold part, also called closure or preplosive pause. Speech pauses corresponding to phrase boundaries were labeled with the symbol “/”. For a plosive following a speech pause, no preplosive pause was segmented. Table 4.1 lists all segment types used for the transcription.

Automatic Segmentation Procedure

Manual transcription and segmentation of the speech would have taken too much time. We applied a segmentation procedure first presented in [Romsdorfer 2004], which simultaneously delivers a highly accurate phonetic segmentation and a close phonetic transcription.

This segmentation procedure relies on iterative Viterbi search for best-matching pronunciation variants and on iterative retraining of phoneme hidden Markov models (HMMs). This procedure does not require elaborate features, just standard mel-frequency cepstral coefficients (MFCCs) and voicing information.

The segmentation was performed in two steps:

1. Context-independent, three-state, left-to-right, phoneme HMMs with 8 Gaussian mixtures per state were trained on the speech data of the corpus using the standard phonetic transcription of the utterances by applying a so-called “flat start” initialization [Young *et al.* 1999].
2. A small set of language- and speaker-dependent pronunciation variation rules was applied to the canonical transcriptions and a recognition network generated for each utterance. Such a network included all pronunciations allowed by the rules.

A Viterbi search then determined the most likely path through the networks and thus delivered an adapted phonetic transcription of each utterance. These new transcriptions were used to retrain the HMMs that were in turn used in the next iteration for the Viterbi search. The procedure stopped when the number of insertions, deletions, and replacements

of phonemes between the current and the previously adapted transcriptions fell below some predefined threshold. Details on this segmentation procedure can be found in [Romsdorfer and Pfister 2005].

Because the length of the analysis window restricts the accuracy of boundary detection of certain segments, *e.g.*, preplosive pauses, an additional post-processing step was added to the second stage, correcting segment boundary placement of specific segment classes based on the speech signal amplitude and voicing information.

4.3 Evaluation

In order to assess the quality of the corpus, we resorted to human observers for evaluating both the eliciting movie clips (4.3.1), and the acquired data in the form of videos containing renderings of the 3D template tracking coupled with the original audio signal (4.3.2). In Section 4.3.3, a preliminary analysis of the data is presented.

4.3.1 Eliciting Videos Evaluation

The speakers themselves were asked to rate the induction videos, just after having watched them and before pronouncing the emotional version of the sentences. A paper form was filled out, allowing grades between 0 and 5 to a set of 11 suggested emotional labels (“Negative”, “Anger”, “Sadness”, “Stress”, “Contempt”, “Fear”, “Surprise”, “Excitement”, “Confidence”, “Happiness”, and “Positive”), where 0 means “I don’t know”, 1 corresponds to “Not at all”, and 5 to “Very”. An additional field was provided, allowing the suggestion of new labels considered appropriate for the clip. The original list of labels was built starting from the six basic emotions and adding/removing labels by a preliminary screening of the eliciting videos (*e.g.*, “Disgust” was never observed and thus removed). We do not claim that these labels represent the space of emotions in general, only that they are adequate for describing the selected video clips.

The eliciting videos were also shown to a larger audience, by means of an online survey, presenting the same structure of the paper form given

to the speakers. The order was randomized, allowing the user to quit the evaluation at any time. In total, 122 people took part in the survey (20.5% of which were native English speakers), labeling over 8 video clips each on average. The mean inter-rater correlation was 0.622 for the speakers and 0.646 for the online survey. We use the Pearson product-moment correlation coefficient throughout the chapter: $\rho_{xy} = \frac{\text{cov}(X,Y)}{\sigma_x \sigma_y}$.

Figs. 4.7 and 4.8 compare the results of the two separate evaluations of the eliciting clips: the cyan bars (left) correspond to the answers given by the speakers, while the magenta bars (right) are the results of the online survey. In Figure 4.7, the histograms show how many times (in percentages of all movie clips) a label was given the grade on the x-axis. The distributions of the grades are very similar, indicating that the laboratory environment had only a minor impact on the perception of affective states.

In Figure 4.8(a), for each emotional label, mean and standard deviation of its perceived strength are plotted over all sentences. Figure 4.8(b) compares the number of sentences labeled as the corresponding emotion on the x-axis (*i.e.*, with an average grade > 3), giving an idea of the affective content of the eliciting videos. In general, we note a predominance of negative labels, and, for the online survey, a slightly higher standard deviation and a tendency to give higher grades to negative emotions. Fear and contempt were the least perceived affective states from our eliciting videos. The most suggested additional labels were “Nervousness”, “Disappointment”, and “Frustration”.

Some of the labels naturally depend on each others, as can be seen in Figure 4.9, plotting the correlation between the evaluations of the online survey, where the brighter upper-left and lower-right corners indicate a high correlation among positive and negative states. Correlation exists between some of the basic emotions (e.g., “Sadness” and “Fear”, or “Surprise” and “Happiness”), indicating that a single label procedure based on the basic emotions would have been insufficient to describe the affective states present in our eliciting videos, and thus supporting our choice of an expanded label set.

Figure 4.10 tries to judge the suggested affective states: For each label, the bar represents how many times (in percentage of all evaluations) it was given the value 0 (“I don’t know”). “Contempt” and “Confidence”

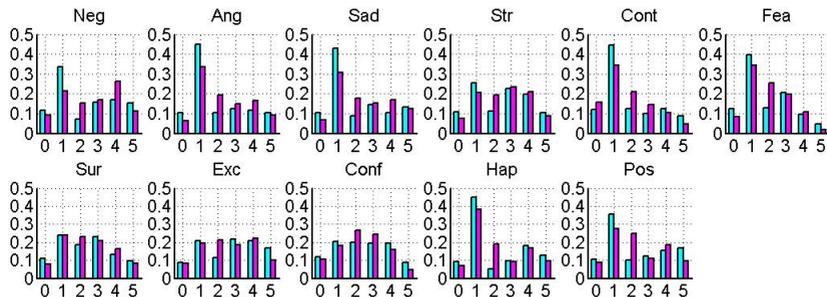
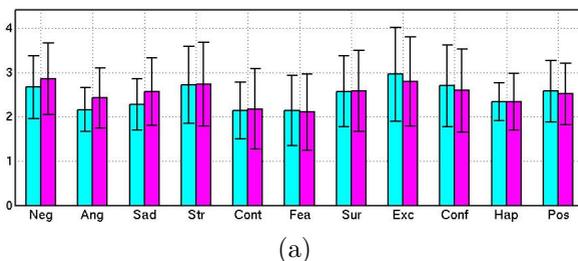
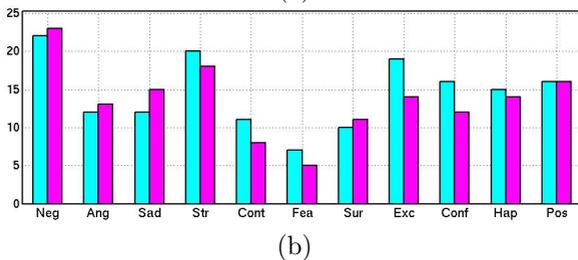


Figure 4.7: The histograms show the contents of the eliciting videos, each graph corresponds to one of the allowed eliciting adjectives. The bars show how many times (in percentage) the label was given the corresponding grade (0 to 5, on the x-axis) by the speakers (left, cyan), respectively by the online survey users (right, magenta)



(a)



(b)

Figure 4.8: The eliciting videos evaluated by the speakers (cyan-left) and by users of the online survey (magenta-right); for each emotional label, (a) shows mean and standard deviation of the received grades, while (b) represents the number of sentences given an average grade > 3 for that label.

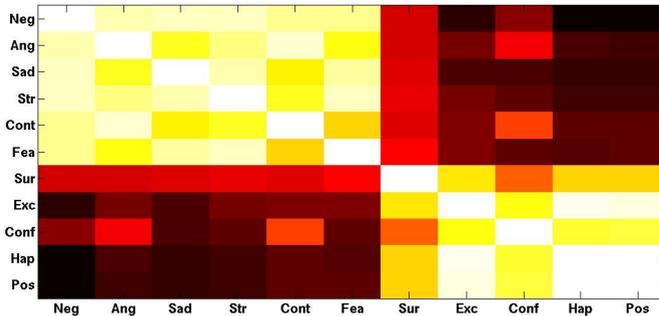


Figure 4.9: Correlations between the affective adjectives, given the evaluations of the eliciting videos in the online survey. There is a high correlation (bright fields) among positive and negative emotions.

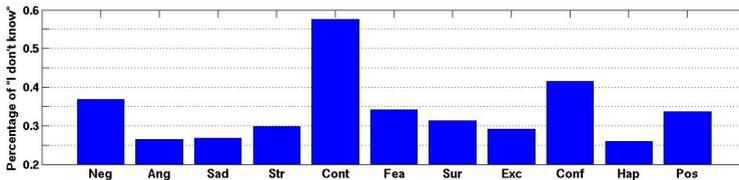
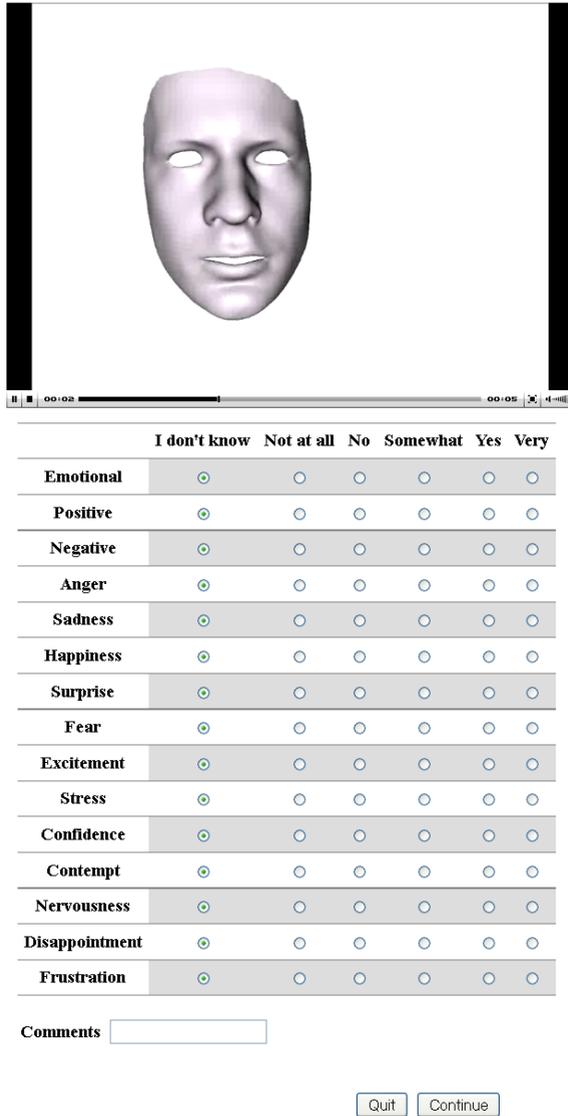


Figure 4.10: For each label (on the x-axis) the number of times it was rated 0 (“I don’t know”) is shown in percentage of all the evaluations of the eliciting videos by the online survey. “Contempt” and “Confidence” were the labels of which people were least certain.

were given zeros most often, possibly being the states which the observers of the online survey were least certain of.

4.3.2 Corpus Evaluation

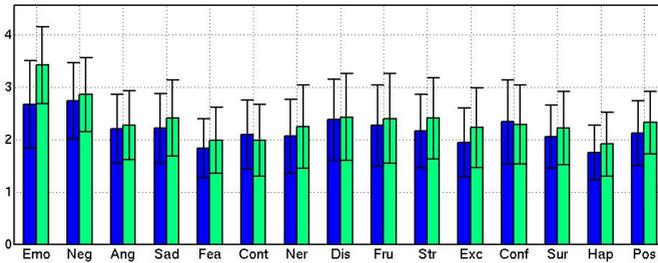
In order to assess the quality of the acquired data, videos were created containing renderings of the tracked 3D faces and the original audio signals. A new survey was designed, where the suggested emotional label set was enriched by the three states most commonly suggested during the evaluations of the eliciting videos (“Nervousness”, “Disappointment”, “Frustration”), and by the additional label “Emotional”.



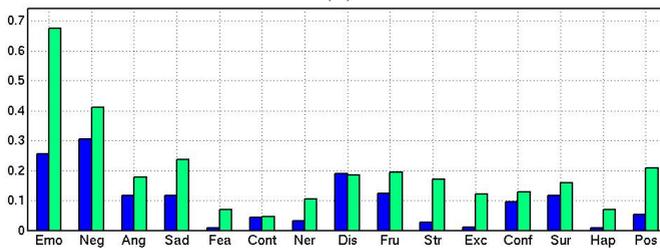
	I don't know	Not at all	No	Somewhat	Yes	Very
Emotional	<input checked="" type="radio"/>	<input type="radio"/>				
Positive	<input checked="" type="radio"/>	<input type="radio"/>				
Negative	<input checked="" type="radio"/>	<input type="radio"/>				
Anger	<input checked="" type="radio"/>	<input type="radio"/>				
Sadness	<input checked="" type="radio"/>	<input type="radio"/>				
Happiness	<input checked="" type="radio"/>	<input type="radio"/>				
Surprise	<input checked="" type="radio"/>	<input type="radio"/>				
Fear	<input checked="" type="radio"/>	<input type="radio"/>				
Excitement	<input checked="" type="radio"/>	<input type="radio"/>				
Stress	<input checked="" type="radio"/>	<input type="radio"/>				
Confidence	<input checked="" type="radio"/>	<input type="radio"/>				
Contempt	<input checked="" type="radio"/>	<input type="radio"/>				
Nervousness	<input checked="" type="radio"/>	<input type="radio"/>				
Disappointment	<input checked="" type="radio"/>	<input type="radio"/>				
Frustration	<input checked="" type="radio"/>	<input type="radio"/>				

Comments

Figure 4.11: Screenshot of the anonymous online survey set up for evaluating the contents of the corpus.



(a)



(b)

Figure 4.12: Evaluations of the neutral (blue-left, read from text) and emotional (green-right, pronounced after having watched the eliciting video) sentences of the corpus. For each label, mean and standard deviation of the received grades are plotted in (a), while (b) shows the percentage of sentences given an average grade > 3 for that label. The plots show that the emotional part of the corpus was indeed evaluated as such by the anonymous observers.

The anonymous users of the survey were presented with the sentences in a randomized order; Figure 4.11 shows a screen shot of the survey page, with the video and the available annotations. In the following, we present results related to sentences which were rated at least 3 times.

The plots in Figure 4.12 compare how the users of the survey (over 800 people) perceived the two parts of the corpus, *i.e.*, the sentences read from text (blue-left) and pronounced after watching the eliciting video (green-right). In (a), average grade and standard deviation over all sentences are given for each label, while in (b) the percentage of sentences which were given an average grade greater than 3 for the label on x-axis is shown. There is evidence of a general increase in the grades given to the

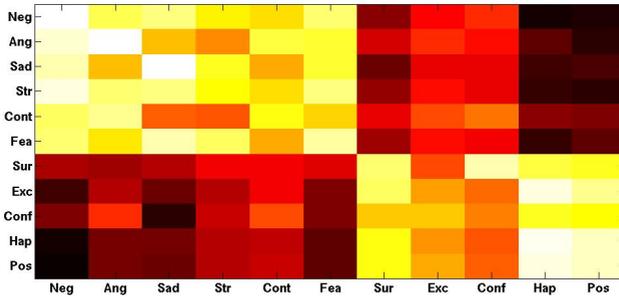


Figure 4.13: Correlations between the evaluations of the eliciting videos (y-axis) and of the videos containing the renderings of the emotional sentences of the corpus (x-axis). The correlation (bright fields) among positive and negative emotions is visible.

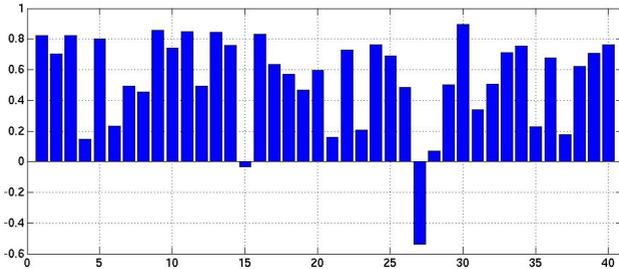


Figure 4.14: For each corpus sentence, the correlation is shown between the average evaluation of the corresponding eliciting video and the average evaluation of the sentence pronounced by the speakers after watching the video. High values correspond to agreement in the evaluations.

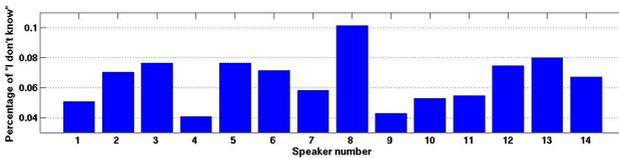


Figure 4.15: Number of “I don’t know” (in percentage) received by all emotional sentences pronounced by the speaker specified on the x-axis

emotional labels for the sentences pronounced after watching the eliciting videos, showing the effectiveness of the induction method; however, the result is unclear for labels like “Contempt” and “Confidence”, supporting the intuition of Figure 4.10.

Figure 4.13 compares the sentences uttered after watching the eliciting videos and the videos themselves by plotting the correlation between the subset of labels shared by the two surveys. Correlation is still noticeable among positive and negative emotions, but not as much as in Figure 4.9, *e.g.*, for “Confidence”. Figure 4.14 shows the correlation between the evaluations of an eliciting video and the evaluations of the corresponding sentence as pronounced by the speakers after watching the video. Most of the 40 utterances show high correlation (1 means full agreement), but some specific sentences show lower agreement, notably number 27, where apparently the emotional state perceived from the video was not similar to the one conveyed by the speakers’ performance. This is not surprising since the eliciting videos are longer than the sentences in the corpus and thus can more easily build the emotional states in the viewer. Also the absence of eyes, facial texture, and rest of the body, makes the renderings of the tracked faces less effective in conveying the emotions.

Figure 4.15 shows the number of times (in percentage) “I don’t know” was chosen when evaluating the emotional sentences pronounced by the speaker specified on the x-axis. Speaker number 8 was given zeros about 10% of the time, appearing to be the least effective in conveying the affective states.

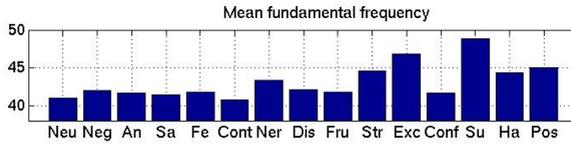
4.3.3 Data Analysis

In order to perform some preliminary studies on the acquired data and demonstrate possible uses of it, we proceeded by selecting as neutral the utterances with an average grade smaller than 3 for the label “Emotional”. For each other affective label, we considered sequences which were given a mean grade greater than 3 for that label. The plots in Figure 4.16 show the relations of the affective adjectives and simple audio and video features, averaged over all sequences labeled according to the above rule. In particular, Figure 4.16 (a) refers to the fundamental frequency, suggesting that positive emotions manifest themselves in

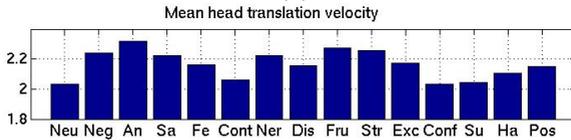
higher values of F_0 . Figure 4.16 (b) shows the mean first derivatives computed over the magnitude of the rigid translations of the heads, *i.e.*, average head velocity. Emotional sentences present on average higher velocities, especially for affective states like “Anger” and “Frustration“. These plots indicate that a single feature is not enough to recognize the affective state but that already several low-level audio-visual cues can give some evidence for the emotion.

Figure 4.17 demonstrates that some correlation exists between auditory and visual channel of our corpus. The plots show the correlation (over all acquired frames) between F_0 , first 12 mel frequency cepstral coefficients ($m_0 - m_{11}$), and mean Gaussian curvature calculated over the facial surfaces at the cheeks, mouth, and eyebrows regions (c_c , c_m , and c_b), for the sentences labeled as “Sad“ (a), and “Happy“ (b). As expected, there is strong correlation (bright areas) within features extracted from the same modality (especially for some of the audio features). However, correlation is also present between features extracted from different modalities. Note that the strength of the correlation between audio and visual features differs for the two labels.

Thanks to the accurate phoneme segmentation and spatio-temporal correspondences among all facial scans, we arranged the 3D face scans into groups corresponding to particular phonemes, and thus built a statistical model of the phonemes’ visual appearance (visemes). Figure 4.18 shows the result of applying Principal Component Analysis to the scans corresponding to the phoneme “P”, as uttered by the same subject. The three rows show the three main modes of variation observed in the data, with the average in the middle and the faces generated by setting the corresponding weights to -3 std. on the left, and respectively $+3$ std. on the right. The example suggests that most of the variation spanned by the first modes corresponds to changes in the expressiveness of the speech (coarticulation effects are reduced to a minimum by selecting the central frame for each phoneme segment). This simple example shows the power of our facial representation, which paves the way for automatic visual speech synthesis and recognition.

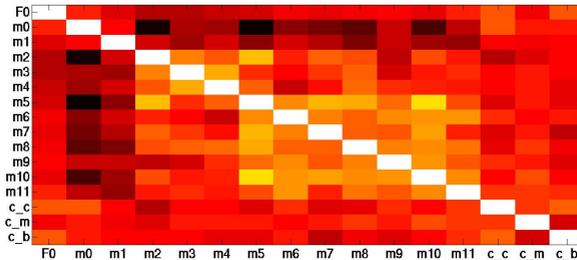


(a)

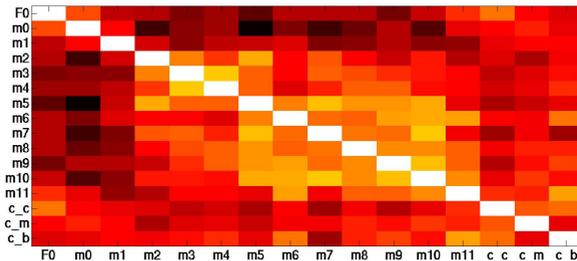


(b)

Figure 4.16: (a) F_0 averaged over each adjective (x-axis), i.e., over the sentences with a mean grade > 3 for that label. (b) Mean head translation velocity for sequences labeled as the adjectives on the x-axis.



(a)



(b)

Figure 4.17: Correlation between audio (F_0 , $m_0 - m_{11}$) and geometric features (c_c , c_m , and c_b) for (a) “sad”, and (b) “happy” sentences. Correlation is present both within and across the two modalities.



Figure 4.18: First three modes of the PCA model of the phoneme “T”. The middle column shows the average face, while the left and right columns represent the result of setting the mode’s weight to -3 std. and $+3$ std., respectively.

4.4 Conclusion

This chapter presented an audio-visual corpus comprised of affective speech and corresponding dense dynamic 3D face geometries. We also explained in details the system which we employed for the corpus' acquisition and automatic annotation. The setup was designed for the recording of high quality data, targeting applications like visual speech modeling for synthesis and recognition purposes.

The recordings of naturalistic emotions being unfeasible in the required studio environment, we resorted to eliciting videos to induce the affective states in the speakers. Our corpus stands out from all currently available datasets, which are either completely posed, limited to dynamic facial expressions without speech, or lacking 3D information.

The corpus comprises 1109 sentences uttered by 14 native English speakers, in the form of audio plus dense dynamic face depth data. For the speech signal, a phonological representation of the utterances, phoneme segmentation, fundamental frequency, and signal intensity are provided. The depth signal is converted into a sequence of 3D meshes, providing full spatial and temporal correspondences across all sequences and speakers, a vital requirement for generating advanced statistical models which can be used for animation or recognition applications.

Current time-consuming steps of our setup are the recording of the raw data, which takes about 1.5 hours for 80 short sentences spoken by one person, and the evaluation of the affective states in the processed data. While the recording process cannot be speeded up, the evaluation was widely spread using a web-based survey.

Although the evaluation shows that similar affective states are perceived by human observers when watching the eliciting videos and the processed data from the corpus, the used induction method is not a replacement of naturalism. This is the price to pay for high quality data. Another limitation is the fact that the 3D visual modality does not include eyes, eyelids, inner mouth, and other body parts beside the face. The raw 3D data being part of the corpus, better templates could be used to track the faces and fill some of the above gaps.

5

Random Forests for Real Time 3D Face Analysis

Despite recent advances, people still interact with machines through devices like keyboards and mice, which are not part of natural human-human communication. As people interact by means of many channels, including body posture and facial expressions, an important step towards more natural interfaces is the visual analysis of the user's movements by the machine. Besides the interpretation of full body movements, as done by systems like the Kinect for gaming, new interfaces would highly benefit from automatic analysis of facial movements.

Recent work has mainly focused on the analysis of standard images or videos; see the survey of [Murphy-Chutorian and Trivedi 2009] for an overview of head pose estimation from video. The use of 2D imagery is very challenging though, not least because of the lack of texture in some facial regions. On the other hand, depth-sensing devices have recently become affordable (*e.g.*, Microsoft Kinect or Asus Xtion) and in some cases also accurate (*e.g.*, [Weise *et al.* 2007]).

The newly available depth cue is key for solving many of the problems inherent to 2D video data. Yet, 3D imagery has mainly been leveraged for face tracking [Weise *et al.* 2009a, Weise *et al.* 2011, Breidt *et al.* 2011, Cai *et al.* 2010], often leaving open issues of drift and (re-)initialization. Tracking-by-detection, on the other hand, detects the face or its features in each frame, thereby providing increased robustness.

The definition of 3D head pose estimation generally means localizing a specific facial feature point (*e.g.*, the nose) and determining the head

orientation (*e.g.*, as Euler angles). When 3D data is used, most methods rely on geometry to localize prominent facial points like the nose tip [Lu and Jain 2006, Chang *et al.* 2006, Sun and Yin 2008, Breitenstein *et al.* 2008, Breitenstein *et al.* 2009] and thus becoming sensitive to its occlusion. Moreover, most of the available algorithms are either not real time, rely on some assumption for initialization like starting with a frontal pose, or cannot handle large rotations.

We propose to use random regression forests for real time head pose estimation and facial feature localization from depth images. We introduce a voting framework where patches extracted from the whole depth image can contribute to the estimation task, similarly to what was presented in chapters 2 and 3. The proposed method does not rely on specific hardware and can easily trade-off accuracy for speed. We estimate the desired, continuous parameters directly from the depth data, through a learnt mapping from depth to parameter values. Our system works in real time, without manual initialization. In our experiments, we show that it also works for unseen faces and that it can handle large pose changes, variations in facial hair, and partial occlusions due to glasses, hands, or missing parts in the 3D reconstruction. It does not rely on specific features like the nose tip.

Random forests show their power when using large datasets, on which they can be trained efficiently. Because the accuracy of a regressor depends on the amount of annotated training data, the acquisition and labeling of a training set are key issues. Depending on the expected scenario, we either synthetically generate annotated depth images by rendering a face model undergoing large rotations, or record real sequences using a consumer depth sensor, automatically annotating them using state-of-the-art tracking methods.

A preliminary version of this chapter was published in [Fanelli *et al.* 2011a], where we introduced the use of random regression forests for real-time head pose estimation from high quality range scans. In [Fanelli *et al.* 2011b], we extended the forest to cope with depth images where the whole body can be visible, *i.e.*, discriminating depth patches that belong to a head and only using those to predict the pose, jointly solving the classification and regression problems involved. In this chapter we provide a thorough experimental evaluation and extend the random forest

framework with the important localization of several facial landmarks on the range scans.

5.1 Related work

Here we present existing works related to head pose estimation and facial features detection. Methods related to random forests were listed in Sec. 2.1.1.

5.1.1 Head pose estimation

With application ranging from image normalization for recognition to driver drowsiness detection, automatic head pose estimation is an important problem. Several approaches have been proposed in the literature [Murphy-Chutorian and Trivedi 2009]; before introducing 3D approaches, which are more relevant for our work, we present a brief overview of algorithms that take 2D images as input. Methods based on 2D images can be subdivided into appearance-based and feature-based classes, depending on whether they analyze the face as a whole or instead rely on the localization of some specific facial features.

2D Appearance-based methods. Such methods usually discretize the head pose space and learn separate detectors for subsets of poses [Jones and Viola 2003]. The works of [Chen *et al.* 2003] and [Balasubramanian *et al.* 2007] present head pose estimation systems with a specific focus on the mapping from the high-dimensional space of facial appearance to the lower-dimensional manifold of head poses. The latter paper considers face images with varying poses as lying on a smooth low-dimensional manifold in a high-dimensional feature space. The proposed Biased Manifold Embedding uses the pose angle information of the face images to compute a biased neighborhood of each point in the feature space, prior to determining the low-dimensional embedding. In the same vein, [Osadchy *et al.* 2005] instead use a convolutional network to learn the mapping, achieving real time performance for the face detection problem, while also providing an estimate of the head pose. A very popular family of methods use statistical models of the face shape and appearance,

like Active Appearance Models (AAMs) [Cootes *et al.* 2001], multi-view AAMs [Ramnath *et al.* 2008], and 3D Morphable Models [Blanz and Vetter 1999, Storer *et al.* 2009]. Such methods usually focus on tracking facial features rather than estimating the head pose, however. In this context, the authors of [Martins and Batista 2008] coupled an Active Appearance Model with the POSIT algorithm for head pose tracking.

2D Feature-based methods. These methods rely on some specific facial features to be visible, and therefore are sensitive to occlusions and to large head rotations. The authors of [Vatahska *et al.* 2007] use a face detector to roughly classify the pose as frontal, left, or right profile. After this, they detect the eyes and nose tip using AdaBoost classifiers, and the detections are fed into a neural network which estimates the head orientation. Similarly, the authors of [Whitehill and Movellan 2008] present a discriminative approach to frame-by-frame head pose estimation. Their algorithm relies on the detection of the nose tip and both eyes, thereby limiting the recognizable poses to the ones where both eyes are visible. In [Morency *et al.* 2008], a probabilistic framework is proposed, called Generalized Adaptive View-based Appearance Model, which integrates frame-by-frame head pose estimation, differential registration, and keyframe tracking.

3D methods. In general, approaches relying solely on 2D images are sensitive to illumination changes and lack of distinctive features. Moreover, the annotation of head poses from 2D images is intrinsically problematic. Since 3D sensing devices have become available, computer vision researchers have started to leverage the additional depth information for solving some of the inherent limitations of image-based methods. Some of the recent works thus use depth as primary cue [Breitenstein *et al.* 2008] or in addition to 2D images [Cai *et al.* 2010, Morency *et al.* 2003, Seemann *et al.* 2004].

The authors of [Seemann *et al.* 2004] presented a neural network-based system fusing skin color histograms and depth information. It tracks at 10 fps but requires the face to be detected in a frontal pose in the first frame of the sequence. The approach of [Mian *et al.* 2006] uses head pose estimation only as a pre-processing step to face recognition, and the low reported average errors are only calculated on subjects present in the training set. Still in a tracking framework, the authors of [Morency *et al.* 2003] use instead an intensity and depth input image to build a prior

model of the face using 3D view-based eigenspaces. Then, they use this model to compute the absolute difference in pose for each new frame. The pose range is limited and manual cropping is necessary. In [Cai *et al.* 2010], a 3D face model is aligned to an RGB-depth input stream for tracking features across frames, taking into account the very noisy nature of depth measurements coming from commercial sensors.

Considering instead pure detectors on a frame-by-frame basis, the authors of [Lu and Jain 2006] create hypotheses for the nose position in range images based on directional maxima. For verification, they compute the nose profile using PCA and a curvature-based shape index. In [Breitenstein *et al.* 2008], a real time system working on range scans provided by the scanner of [Weise *et al.* 2007] is presented. Their system can handle large pose variations, facial expressions, and partial occlusions, as long as the nose remains visible. The method relies on several candidate nose positions, suggested by a geometric descriptor. Such hypotheses are all evaluated in parallel on a GPU, which compares them to renderings of a generic template with different orientations, finally selecting the orientation which minimizes a predefined cost function. Real time performance is only met thanks to the parallel GPU computations. Unfortunately, GPUs are power-hungry and might not be available in many scenarios where portability is important, *e.g.*, for mobile robots. Breitenstein *et al.* also collected a dataset of over 10K annotated range scans of heads. The subjects, both male and female, with and without glasses, were recorded using the scanner of [Weise *et al.* 2007] while turning their heads around, trying to span all possible yaw and pitch rotation angles they could. The scans were automatically annotated, tracking each sequence using ICP in combination with a personalized face template. The same authors also extended their system to use lower quality depth images from a stereo system [Breitenstein *et al.* 2009]. Yet, the main shortcomings of the original method remain.

5.1.2 Facial features localization

2D Facial Features. Facial feature detection from standard images is a well studied problem, often performed as preprocessing for face recognition. Previous contributions can be classified into two categories, depending on whether they use global or local features. Holistic methods,

e.g., Active Appearance Models [Cootes *et al.* 2001, Cootes *et al.* 2002, Matthews and Baker 2003], use the entire facial texture to fit a generative model to a test image. They are usually affected by lighting changes and a bias towards the average face. The complexity of the modeling is an additional issue. Moreover, these methods perform poorly on unseen identities [Gross *et al.* 2005] and cannot handle low-resolution images well.

In recent years, there has been a shift towards methods based on independent local feature detectors [Valstar *et al.* 2010, Amberg and Vetter 2011, Belhumeur *et al.* 2011]. Such detectors are discriminative models of image patches centered around the facial landmarks, often ambiguous because the limited support region cannot cope with the large appearance variations present in the training samples. To improve accuracy and reduce the influence of wrong detections, global models of the facial features configuration like pictorial structures [Felzenszwalb and Huttenlocher 2005, Everingham *et al.* 2006] or Active Shape Models [Cristinacce and Cootes 2008] are needed.

3D Facial Features. Similar to the 2D case, methods focusing on facial feature localization from range data can be subdivided into categories using global or local information. Among the former class, the authors of [Mpiperis *et al.* 2008] deform a bi-linear face model to match a scan of an unseen face in different expressions. Yet, the paper's focus is not on the localization of facial feature points and real time performance is not achieved. Also the authors of [Kakadiaris *et al.* 2007] non-rigidly align an annotated model to face meshes. Constraints need to be imposed on the initial face orientation, however. Using high quality range scans, the work of [Weise *et al.* 2009a] presented a real time system, capable of tracking facial motion in detail, but using personalized templates. The same approach has been extended to robustly track head pose and facial deformations using RGB-depth streams provided by commercial sensors like the Kinect [Weise *et al.* 2011].

Most works that try to directly localize specific feature points from 3D data take advantage of surface curvatures. For example, the authors of [Sun and Yin 2008, Segundo *et al.* 2010, Chang *et al.* 2006] all use curvature to roughly localize the inner corners of the eyes. Such an approach is very sensitive to missing depth data, particularly for the regions around the inner eye corners, frequently occluded by shadows. Surface

curvatures are also used in [Mehryar *et al.* 2010], first by extracting ridge and valley points, which are then clustered. The clusters are refined using a geometric model imposing a set of distance and angle constraints on the arrangement of candidate landmarks. In [Colbry *et al.* 2005], curvature is used in conjunction with the Shape Index proposed by [Dorai and Jain 1997] to locate facial feature points from range scans of faces. The reported execution time of this anchor point detector is 15 sec per frame. The authors of [Wang *et al.* 2002] use point signatures [Chua and Jarvis 1997] and Gabor filters to detect some facial feature points from 3D and 2D data. The method needs all desired landmarks to be visible, thus restricting the range of head poses while being sensitive to occlusions. Genetic algorithms are used in [Yu and Moon 2008] to combine several weak classifiers into a 3D facial landmark detector. The authors of [Ju *et al.* 2009] detect the nose tip and the eyes using binary neural networks, and propose a 3D shape descriptor invariant to pose and expression.

The authors of [Zhao *et al.* 2011] propose a 3D Statistical Facial Feature Model (SFAM), which models both the global variations in the morphology of the face and the local structures around the landmarks. The low reported errors for the localization of 15 points in scans of neutral faces come at the expense of processing time: over 10 minutes are needed to process one facial scan. In [Nair and Cavallaro 2009], fitting the proposed PCA shape model containing only the upper facial features, *i.e.*, without the mouth, takes on average 2 minutes per face.

In general, prior work on facial feature localization from 3D data is either sensitive to occlusions, especially of the nose, requires prior knowledge of feature map thresholds, cannot handle large rotations, or does not run in real time.

5.2 Random forests for 3D face analysis

In Section 5.2.1 we first summarize a general random forest framework [Breiman 2001], then give specific details for face analysis based on depth data in Sections 5.2.2 and 5.2.3.

5.2.1 Random forest

Random forests were already introduced in Chapter 2 for the task of mouth localization and in Chapter 3 for facial expression recognition. Fig. 5.1 illustrates a random regression forest mapping feature patches extracted from a depth image to a distribution stored at each leaf. In our framework, these distributions model the head orientation or locations of facial features. In the following, we outline the general training approach of a random forest and give the application specific details in Sections 5.2.2 and 5.2.3.

A tree T in a forest $\mathcal{T} = \{T_i\}$ is built from a set of annotated patches, randomly extracted from the training images: $\mathcal{P} = \{P_i\}$, where \mathcal{I}_i is the appearance of the patch. Starting from the root, each tree is built recursively by assigning a binary test $\phi(\mathcal{I}) \rightarrow \{0, 1\}$ to each non-leaf node. Such test sends each patch (according to its appearance) either to the left or right child, in this way, the training patches \mathcal{P} arriving at the node are split into two sets, $\mathcal{P}_L(\phi)$ and $\mathcal{P}_R(\phi)$.

The best test ϕ^* is chosen from a pool of randomly generated ones ($\{\phi\}$): all patches arriving at the node are evaluated by all tests in the pool and a predefined information gain of the split $IG(\phi)$ is maximized:

$$\phi^* = \arg \max_{\phi} IG(\phi) \quad (5.1)$$

$$IG(\phi) = \mathcal{H}(\mathcal{P}) - \sum_{i \in \{L, R\}} w_i \mathcal{H}(\mathcal{P}_i(\phi)), \quad (5.2)$$

where $w_i = \frac{|\mathcal{P}_i(\phi)|}{|\mathcal{P}|}$ is the ratio of patches sent to each child node and $\mathcal{H}(\mathcal{P})$ is a measure of the patch cluster \mathcal{P} , usually related to the entropy of the clusters' labels. The measure $\mathcal{H}(\mathcal{P})$ can have different forms, depending on whether the goal of the forest is regression, classification, or rather a combination of the two. The measures that are relevant for face analysis are discussed in Sections 5.2.2 and 5.2.3. The process continues with the left and the right child using the corresponding training sets $\mathcal{P}_L(\phi^*)$ and $\mathcal{P}_R(\phi^*)$ until a leaf is created when either the maximum tree depth is reached, or less than a minimum number of training samples are left.

In order to employ such a random forest framework for face analysis from depth data, we have to

- acquire annotated training data \mathcal{P} ,
- define binary tests ϕ ,
- define a measure $\mathcal{H}(\mathcal{P})$,
- define a distribution model to be stored at the leaves.

These issues are discussed in the following sections.

5.2.2 Head pose estimation

Training data

Building a forest is a supervised learning problem, *i.e.*, training data needs to be annotated with labels on the desired output space. In our head pose estimation setup, a training sample is a depth image containing a head, annotated with the 3D locations of a specific point, *i.e.*, the tip of the nose, and the head orientation. Fix-sized patches are extracted from a training image, each annotated with two real-valued vectors: While $\boldsymbol{\theta}^1 = \{\theta_x, \theta_y, \theta_z\}$ is the offset computed between the 3D point falling at the patch center and the nose tip, the head orientation is encoded as Euler angles, $\boldsymbol{\theta}^2 = \{\theta_{ya}, \theta_{pi}, \theta_{ro}\}$. In order for the forest to be more scale-invariant, the size of the patches can be made dependent on the depth (*e.g.*, at its center), however, in this work we assume the faces to be within a relatively narrow range of distances from the sensor.

In order to deal with background like hair and other body parts, fixed-sized patches are not only sampled from faces but also from regions around them. A class label c_i is thus assigned to each patch P_i , where $c_i = 1$ if it is sampled from the face and 0 otherwise. The set of training patches is therefore given by $\mathcal{P} = \{P_i = (\mathcal{I}_i, c_i, \boldsymbol{\theta}_i)\}$, where $\boldsymbol{\theta} = (\boldsymbol{\theta}^1, \boldsymbol{\theta}^2)$. \mathcal{I}_i represents the image features \mathcal{I}_i^f computed from a patch P_i . Such features include the original depth values plus, optionally, the geometric normals, *i.e.*, for a depth pixel $d(u, v)$, the average of the normals of the planes passing through $d(u, v)$ and pairs of its 4-connected neighbors. The x, y, and z coordinates of the normals are treated as separate feature channels.

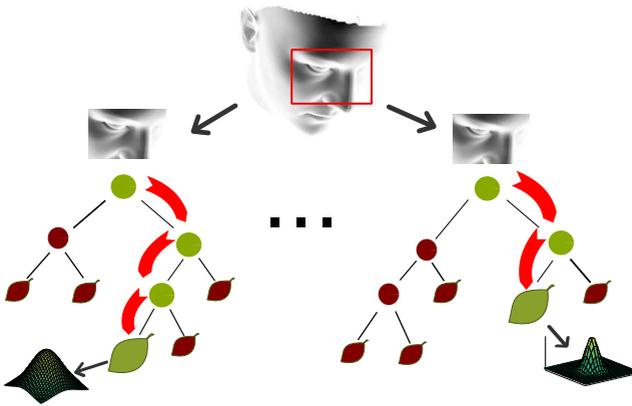


Figure 5.1: Example of regression forest for head pose estimation. For each tree, the tests at the non-leaf nodes direct an input sample towards a leaf, where a real-valued, multivariate distribution of the output parameters is stored. The forest combines the results of all leaves to produce a probabilistic prediction in the real-valued output space.

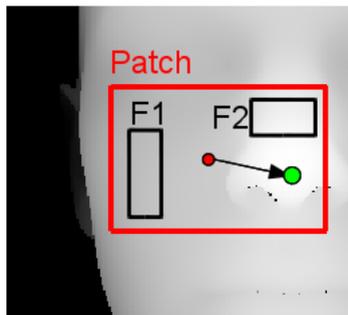


Figure 5.2: Example of a training patch (larger, red rectangle), offset vector (arrow) between the 3D point at the patch's center (red dot) and the ground truth nose location (green). F1 and F2 represent a possible choice for the regions over which to compute a binary test.

Binary tests

Our binary tests $\phi_{f, F_1, F_2, \tau}(\mathcal{I})$ are defined as:

$$|F_1|^{-1} \sum_{\mathbf{q} \in F_1} \mathcal{I}^f(\mathbf{q}) - |F_2|^{-1} \sum_{\mathbf{q} \in F_2} \mathcal{I}^f(\mathbf{q}) > \tau, \quad (5.3)$$

where f is the feature channel's index, F_1 and F_2 are two asymmetric rectangles defined within the patch, and τ is a threshold. We use the difference between the average values of two rectangular areas as in [Fanelli *et al.* 2011a, Criminisi *et al.* 2010], rather than single pixel differences as in [Gall *et al.* 2011] in order to be less sensitive to noise; the additional computation is negligible when integral images are used. Tests defined as Equation (5.3) represent a generalization of the widely-used Haar-like features [Papageorgiou *et al.* 1998]. An example test is shown in Figure 5.2: A patch is marked in red, containing the two regions F_1 and F_2 defining the test (in black); the arrow represents the 3D offset vector (θ^1) between the 3D patch center (in red) and the ground truth location of a feature point, the nose tip in this case (green).

Goodness of split

A regression forest can be applied to head pose estimation from depth images containing only faces [Fanelli *et al.* 2011a]; in this case, all training patches are positive ($c_i = 1 \forall i$) and the measure $\mathcal{H}(\mathcal{P})$ is defined as the entropy of the continuous patch labels. Assuming θ^n , where $n \in \{1, 2\}$, to be realizations of 3-variate Gaussians, we can represent the labels in a set \mathcal{P} as $p(\theta^n) = \mathcal{N}(\theta^n; \bar{\theta}^n, \Sigma^n)$, and thus compute the differential entropy $H(\mathcal{P})^n$ for n :

$$H(\mathcal{P})^n = \frac{1}{2} \log \left((2\pi e)^3 |\Sigma^n| \right). \quad (5.4)$$

We thus define the regression measure:

$$\mathcal{H}_r(\mathcal{P}) = \sum_n \log(|\Sigma^n|) \propto \sum_n H(\mathcal{P})^n. \quad (5.5)$$

Substituting Equation (5.5) into Equation (5.2) and maximizing it actually favors splits which minimize the covariances of the Gaussian distributions computed over all label vectors θ^n at the children nodes, thus intuitively decreasing the regression uncertainty.

Goal of the forest, however, is not only to map image patches into probabilistic votes in a continuous space, but, as in [Fanelli *et al.* 2011b], also to decide which patches are actually allowed to cast such votes. In order to include a measure of the classification uncertainty in the information gain defined by Equation (5.2), we use the measure $\mathcal{H}_c(\mathcal{P})$ of the cluster's class uncertainty, defined as the entropy:

$$\mathcal{H}_c(\mathcal{P}) = - \sum_{k=0}^K p(c = k|\mathcal{P}) \log(p(c = k|\mathcal{P})), \quad (5.6)$$

where $K = 1$. The class probability $p(c = k|\mathcal{P})$ is approximated by the ratio of patches with class label k in the set \mathcal{P} .

The two measures (5.5) and (5.6) can be combined in different ways.

One approach, used in the previous chapters 2 and 3, is to randomly select one or the other at each node of the trees, denoted in the following as the *interleaved* method.

A second approach (*linear*) was proposed by [Okada 2009], *i.e.*, a weighted sum of the two measures:

$$\mathcal{H}_c(\mathcal{P}) + \alpha \max(p(c = 1|\mathcal{P}) - t_p, 0) \mathcal{H}_r(\mathcal{P}). \quad (5.7)$$

When minimizing (5.7), the optimization is steered by the classification term alone until the purity of positive patches reaches the activation threshold t_p . From that point on, the regression term starts to play an ever important role, weighted by the constant α , until the purity reaches 1. In this case, $\mathcal{H}_c = 0$ and the optimization is driven only by the regression measure \mathcal{H}_r .

In [Fanelli *et al.* 2011b], we proposed a third approach, where the two measures are weighted by an *exponential* function of the depth:

$$\mathcal{H}_c(\mathcal{P}) + (1 - e^{-\frac{d}{\lambda}}) \mathcal{H}_r(\mathcal{P}), \quad (5.8)$$

where d is the depth of the node in the tree. In this way, the regression measure is given increasingly higher weight as we descend deeper in the tree towards the leaves, with the parameter λ specifying the steepness of the change.

Note that, when only positive patches are available, $\mathcal{H}_c = 0$, *i.e.*, Equations (5.7) and (5.8) are both proportional to the regression measure \mathcal{H}_r .

alone, and both lead to the same selected test ϕ^* , according to Equation (5.1).

In our experiments (see Section 5.3.2), we evaluate the three possibilities for combining the classification measure \mathcal{H}_c and the regression measure \mathcal{H}_r for training.

Leaves

For each leaf, the class probabilities $p(c = k | \mathcal{P})$ and the distributions of the continuous head pose parameters $p(\boldsymbol{\theta}^1) = \mathcal{N}(\boldsymbol{\theta}^1; \overline{\boldsymbol{\theta}^1}, \boldsymbol{\Sigma}^1)$ and $p(\boldsymbol{\theta}^2) = \mathcal{N}(\boldsymbol{\theta}^2; \overline{\boldsymbol{\theta}^2}, \boldsymbol{\Sigma}^2)$ are stored. The distributions are estimated from the training patches that arrive at the leaf and are used for estimating the head pose as explained in the following section.

Testing

When presented with a test depth image, patches are densely sampled from the whole image and sent down through all trees in the forest. Each patch is guided by the binary tests stored at the nodes, as illustrated in Fig. 5.1. A stride parameter controls how densely patches are extracted, thus easily steering speed and accuracy of the regression.

The probability $p(c = k | \mathcal{P})$ stored at the leaf judges how informative the test patch is for class k . This probability value tells whether the patch belongs to the head or other body parts. Since collecting all relevant negative examples is harder than collecting many positive examples, we only consider leaves with $p(c = k | \mathcal{P}) = 1$. For efficiency and accuracy reasons, we also filter out leaves with a high variance, which are less informative for the regression, *i.e.*, all leaves with $\text{tr}(\boldsymbol{\Sigma}^1)$ greater than a threshold max_v . The currently employed threshold ($max_v = 400$) has been set based on a validation set. Although the two criteria seem to be very restrictive, the amount of sampled patches and leaves is large enough to obtain reliable estimates.

The remaining distributions are used to estimate $\boldsymbol{\theta}^1$ by adding the mean offsets $\overline{\boldsymbol{\theta}^1}$ to the patch center $\boldsymbol{\theta}^1(\mathcal{P})$:

$$\mathcal{N}(\boldsymbol{\theta}^1; \boldsymbol{\theta}^1(\mathcal{P}) + \overline{\boldsymbol{\theta}^1}, \boldsymbol{\Sigma}^1). \quad (5.9)$$

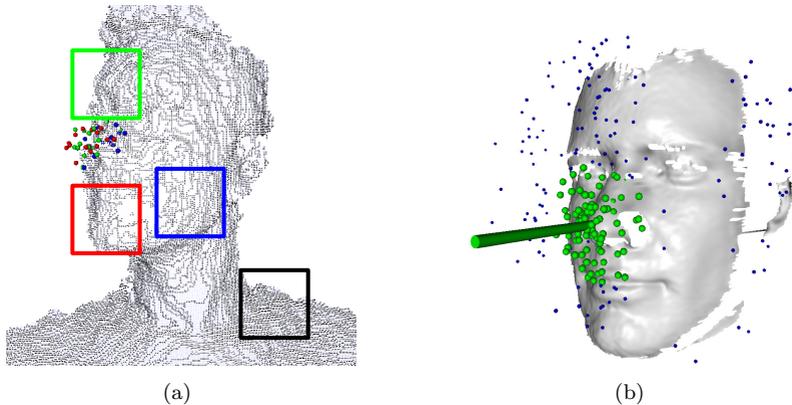


Figure 5.3: (a) Example votes, casted by different patches extracted from a Kinect depth image. The green, red, and blue patch are classified as positives and therefore cast votes for the nose position (correspondingly colored spheres). On the other hand, the black patch at the shoulder is classified as negative and does not vote. (b) Example (high resolution) test image: the green spheres represent the votes selected after outliers (blue spheres) are filtered out by mean shift. The large green cylinder stretches from the final estimate of the nose center in the estimated face direction.

The corresponding means for the position of the nose tip are illustrated in Fig. 5.3. The votes are then clustered, and the clusters are further refined by mean shift in order to remove additional outliers. As kernel for the mean shift, we use a sphere with a radius defined as one sixth of the radius of the average face in the model of [Paysan *et al.* 2009]. A cluster is declared as a head if it contains a large enough number of votes. Because the number of votes is directly proportional to the number of trees in the forest (a tree can contribute up to one vote for each test patch), and because the number of patches sampled is inversely proportional to the square of the stride, we use the following threshold:

$$\beta \frac{\#trees}{stride^2}. \quad (5.10)$$

For our experiments, we use $\beta = 300$.

For each cluster left, *i.e.*, each head detected, the distributions in the clusters are averaged, where the mean gives an estimate for the position of the nose tip θ^1 and the head orientation θ^2 and the covariance measures the uncertainty of the estimates.

5.2.3 Facial features localization

Since the framework for head pose estimation is very general and can be used in principle for predicting any continuous parameter of the face, the modifications for localizing facial features are straightforward. Instead of having only two classes as in Section 5.2.2, we have $K + 1$ classes, where K is the number of facial feature points we wish to localize. The set of training patches is therefore given by $\mathcal{P} = \{\mathcal{P}_i = (\mathcal{I}_i, c_i, \theta_i)\}$, where $\theta_i = \{\theta_i^1, \theta_i^2, \dots, \theta_i^K\}$ are the offsets between the patch center and the 3D locations of each of the K feature points. Accordingly, (5.5) is computed for the K fiducials and (5.6) is computed for the $K + 1$ classes, where $c = 0$ is the label for the background patches.

The testing, however, slightly differs. In Section 5.2.2, all patches are allowed to predict the location of the nose tip and the head orientation. While this works for nearly rigid transformations of the head, the location of the facial features depends also on local deformations of the face, *e.g.*, the mouth shape. In order to avoid a bias towards the average face due to long distance votes that do not capture local deformations, we reduce the influence of patches that are more distant to the fiducial. We measure the confidence of a patch P for the location of a feature point n by

$$\exp\left(-\frac{\|\overline{\theta}^n\|^2}{\gamma}\right), \quad (5.11)$$

where $\gamma = 0.2$ and $\overline{\theta}^n$ is the average offset relative to point n , stored at the leaf where the patch P ends. Allowing a patch to vote only for feature points with a high confidence, *i.e.*, above a feature-specific threshold, our algorithm can handle local deformations better, as our experiments show. The final 3D facial feature points' locations are obtained by performing mean-shift for each point n .

5.3 Evaluation

In this section, we thoroughly evaluate the proposed random forest framework for the tasks of head pose estimation from high quality range scans (Section 5.3.1), head pose estimation from low quality depth images (Section 5.3.2), and 3D facial features localization from high resolution scans (Section 5.3.3). Since the acquisition of annotated training data is an important step and a challenge task itself, we first present the used databases¹ in each subsection.

5.3.1 Head pose estimation - high resolution

Dataset

The easiest way to generate an abundance of training data with perfect ground truth is to synthesize head poses. To this end, we synthetically generated a very large training set of 640x480 range images of faces by rendering the 3D morphable model of [Paysan *et al.* 2009]. We made such model undergo 50K different rotations, uniformly sampled from $\pm 95^\circ$ yaw, $\pm 50^\circ$ pitch, and $\pm 20^\circ$ roll. We also randomly varied the model's distance from the camera and further perturbed the first 30 modes of the PCA shape model sampling uniformly within ± 2 standard deviation, thus introducing variations also in identity².

Such a dataset was automatically annotated with the 3D coordinates of the nose tip and the applied rotations, represented as Euler angles. Figure 5.4 shows a few of the training faces, with the cylinder pointing out from the nose indicating the annotation in terms of nose position and head direction. Note that the shape model captures only faces with neutral expression and closed mouth. Furthermore, important parts of the head like hair or the full neck are missing. This will be an issue in Section 5.3.2, where we discuss the limitations of synthetic training data.

¹Most of the datasets are publicly available at <http://www.vision.ee.ethz.ch/datasets>.

²Because of the proprietary license for [Paysan *et al.* 2009], we cannot share the above database. The PCA model, however, can be obtained from the University of Basel.

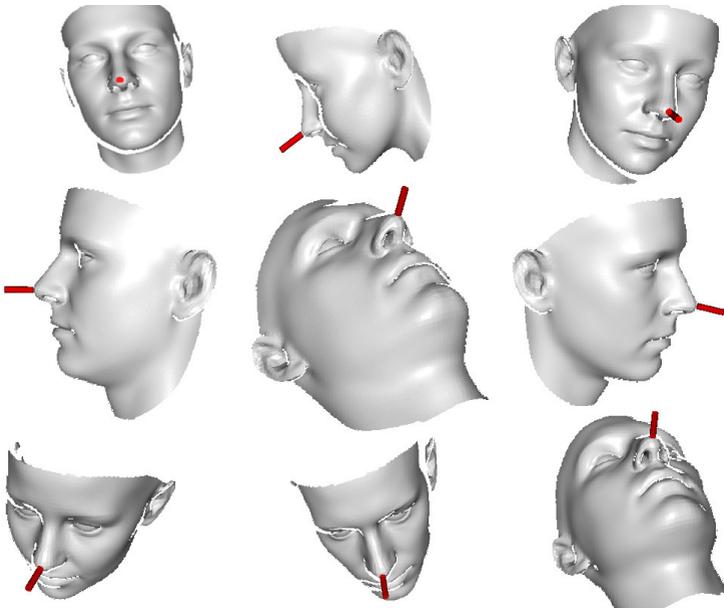


Figure 5.4: Sample images from our synthetically generated training set. The heads show large 3D rotations and variations in the distance to the camera and also in identity. The cylinder attached to the nose represents the ground truth face orientation.

For testing, we use the real sequences of the ETH Face Pose Range Image Data Set [Breitenstein *et al.* 2008]. The database contains over 10K range images of 20 people (3 females, 6 subjects recorded twice, with and without glasses) recorded using the scanner of [Weise *et al.* 2007] while turning their head around, trying to cover all pitch and yaw rotations. The images have a resolution of 640x480 pixels, and a face typically consists of around 150x200 pixels. The heads undergo rotations of about $\pm 90^\circ$ yaw and $\pm 45^\circ$ pitch, while no roll is present. The data was annotated using person-specific templates and ICP tracking, in a similar fashion as what will be later described in 5.3.2 and shown in Figure 5.15. The provided ground truth contains the 3D coordinates of the nose tip and the vector pointing from the nose towards the facing direction.

Experiments

In this section, we assume a face to be the prominent object in the image. That means that all leaves in a tree contain a probability $p(c = 1|\mathcal{P}) = 1$ and thus all patches extracted from the depth image will be allowed to vote, no matter their appearance.

Training a forest involves the choice of several parameters. In the following, we always stop growing a tree when the depth reaches 15, or if there are less than 20 patches left for training. Moreover, we randomly generate 20K tests for optimization at each node, *i.e.*, 2K different combinations of f , F_1 , and F_2 in Equation (5.3), each with 10 different thresholds τ . Other parameters include the number of randomly selected training images, the number of patches extracted from each image (fixed to 20), the patch size, and the maximum size of the sub-patches defining the areas F_1 and F_2 in the tests (set to be half the size of the patch). Also the number of feature channels available is an important parameter; in the following, we use all features (depth plus normals) unless otherwise specified.

A pair of crucial test-time parameters are the number of trees loaded in the forest and the stride controlling the spatial sampling of the patches from an input image. Such values can be intuitively tuned to find the desired trade-off between accuracy and temporal efficiency of the estimation process, making the algorithm adaptive to the constraints of different applications.

In all the following experiments, we use the Euclidean distance in millimeters as the nose localization error. For what concerns the orientation estimation, the ETH database does not contain large roll variations, and in fact these rotations are not encoded in the directional vector provided as ground truth. We therefore evaluated our orientation estimation performance computing the head direction vector from our estimates of the yaw and pitch angles and report the angular error in degrees with the ground truth vector.

Figure 5.5 describes the performance of the algorithm when we varied the size of the training patches and the number of samples used for training each tree. In Figure 5.5(a), the blue, continuous line shows the percentage of correctly classified images as a function of the patch size, when

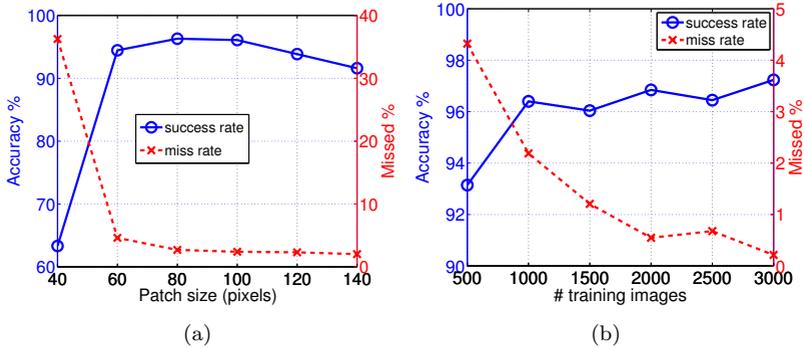


Figure 5.5: (a) Success rate of the system depending on the patch size (when using 1000 training samples), overlaid to the missed detection rate. (b) Success and missed detection rate depending on the number of training data (for 100x100 patches). Success is defined for a nose error below 20 mm and angular error below 15 degrees.

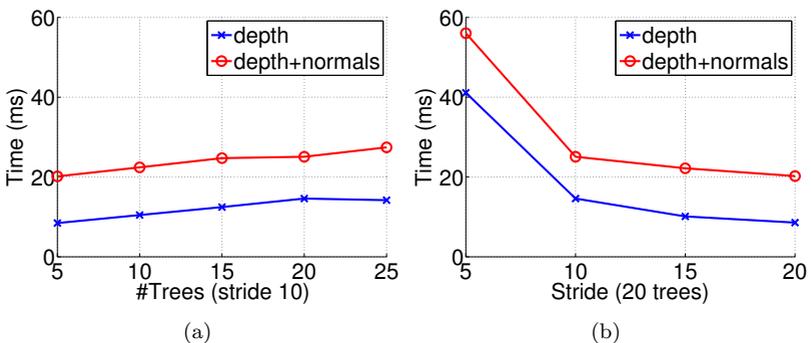


Figure 5.6: Processing time: a) Regression time as a function of the number of trees in the forest when the stride is fixed to 10 pixels. b) Run time for a forest of 20 trees as a function of the stride parameter.

1000 training images are used. Success is declared if the nose error is smaller than 20 mm and the angular error is below 15 degrees. Although this measure might be too generous for some applications, it reflects the relative estimation performance of the approach and is therefore a useful measure for comparing different settings of the proposed approach. The red, dashed line shows instead the percentage of false positives, *i.e.*, missed detections, again varying with the size of the patch. The plot shows that a minimum size for the patches is critical since small patches can not capture enough information to reliably predict the head pose. However, there is also a slight performance loss for large patches. In this case, the trees become more sensitive to occlusions and strong artifacts like holes since the patches cover a larger region and overlap more. Having a patch size between 80x80 and 100x100 pixels seems to be a good choice where the patches are discriminative enough to estimate the head pose, but they are still small enough such that an occlusion affects only a subset of patches. Figure 5.5(b) also shows accuracy and missed detections rate, this time for 100x100 patches, as a function of the number of training images. It can be noted that the performance increases with more training data, but it also saturates for training sets containing more than 2K images. For the following experiments, we trained on 3000 images, extracting 20 patches of size 100x100 pixels from each of them.

In all the following graphs, red circular markers consistently represent the performance of the system when all available feature channels are used (*i.e.*, depth plus geometric normals), while the blue crosses refer to the results achieved employing only the depth channel.

The plots in Figure 5.6 show the time in milliseconds needed to process one frame, once loaded in the RAM. The values are reported as a function of the number of trees used and of the stride parameter. The numbers were computed over the whole ETH database, using an Intel Core i7 CPU @ 2.67GHz processor, without resorting to multithreading. Figure 5.6(a) plots the average run time for a stride fixed to 10 pixels, as a function of the number of trees, while in Figure 5.6(b) 20 trees are loaded and the stride parameter changes instead. For strides equal to 10 and greater, the system always performs in real time. Unless otherwise specified, we use these settings in all the following experiments. Obviously, having to compute the normals (done on the CPU using a 4-neighborhood) increases processing time, but, for the high-quality scans we are dealing

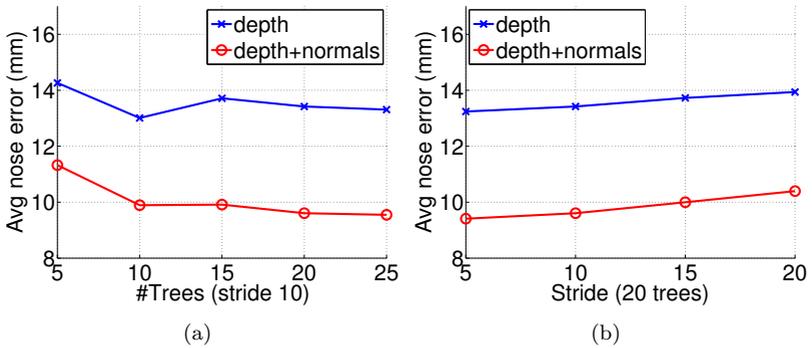


Figure 5.7: Mean errors (in millimeters) for the nose localization task, as a function of the number of trees (a) and of the stride (b).

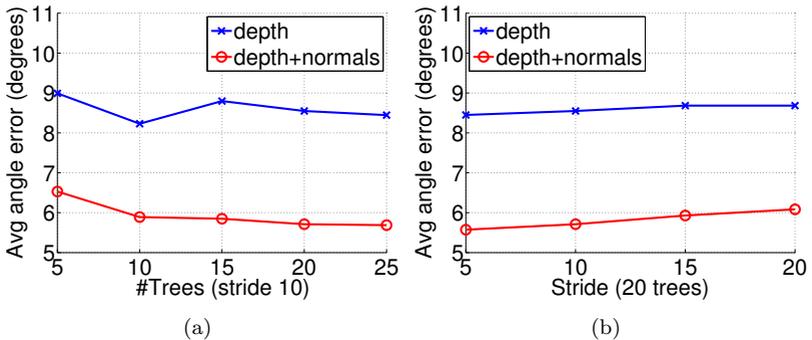


Figure 5.8: Mean errors (degrees) for the orientation estimation task, as a function of the number of trees (a) and of the stride (b).

with, the boost in accuracy justifies the loss in terms of speed, as can be seen in the next plots.

Figure 5.7(a) shows the average errors in the nose localization task, plotted as a function of the number of trees when the stride is fixed to 10, while in Figure 5.7(b) 20 trees are loaded and the stride is changed. Similarly, the plots in Figures 5.8(a) and 5.8(b) present the average errors in the estimation of the head orientation. When comparing Figures 5.6, 5.7, and 5.8, we can conclude that it is better to increase the stride than reducing the number of trees when the processing time needs to be reduced. Using normals in addition also improves the detection performance more than increasing the number of trees. In particular, using depth and normals with a stride of 10 gives a good trade-off between accuracy and processing time for our experimental settings.

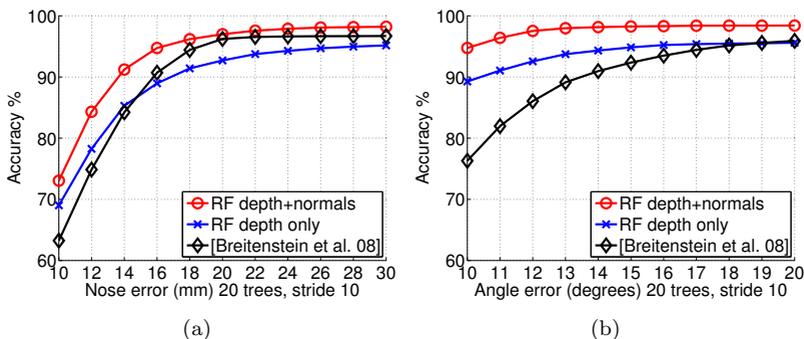


Figure 5.9: Accuracy: (a) Percentage of correctly estimated poses as a function of the nose error threshold. (b) Accuracy plotted against the angle error threshold. The additional information coming from the normals (red curves) consistently boosts the performance. The black curve represents the accuracy of [Breitenstein et al. 2008] on the same dataset.

In Figure 5.9, the plots show the accuracy of the system computed over the whole ETH database, when both depth and geometric normals are used as features. Specifically, the curves in Figure 5.9(a) and Figure 5.9(b) represent the percentage of correctly estimated depth images as functions of the success threshold set for the nose localization error, respectively for the angular error. Using all the available feature channels

performs consistently better than relying only on the depth information. The plots show also the success rate of the method of [Breitenstein *et al.* 2008], applied to the same data³; their algorithm uses information about the normals to generate nose candidates, but not for refining the pose estimation on the GPU, where a measure based on the normalized sum of squared depth differences between reference and input range image is used.

Our approach proves better at both the tasks of nose tip detection and head orientation estimation. We improve over the state-of-the-art especially at low thresholds, which are also the most relevant. In particular, for a threshold of 10 mm on the nose localization error, our improvement is of about 10% (from 63.2% to 73.0%), and even better for a threshold of 10 degrees on the angular error: Our system succeeded in 94.7%, compared to 76.3% of Breitenstein *et al.*

	[Breitenstein <i>et al.</i> 2008]	Random Forests
Nose error (mm)	10.3 ± 17.5	9.6 ± 13.4
Direction error (°)	9.1 ± 12.6	5.7 ± 8.6
Yaw error (°)	7.0 ± 13.4	4.4 ± 2.7
Pitch error (°)	4.8 ± 4.9	3.2 ± 2.7
Dir. acc. ($\leq 10^\circ$)	76.3%	94.7%
Nose acc. (≤ 10 mm)	63.2%	73.0%

Table 5.1: Comparison of our results with the ones of [Breitenstein *et al.* 2008]. Mean and standard deviation are given for the errors on nose localization, direction estimation, and singularly for yaw and pitch angles. The values in the last two rows are the percentages of correctly estimated images for a threshold on the angular error of 10 degrees, and on the nose localization error of 10 millimeters. We used a forest with 20 trees, leveraging both depth and normals as features, testing with a stride of 10 pixels.

³We used the source code provided by the authors.

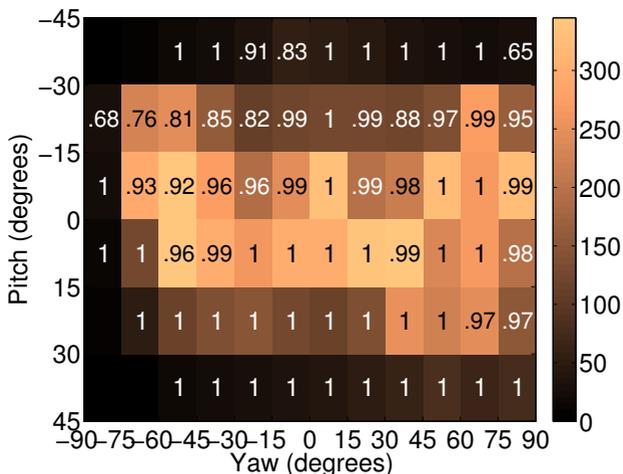


Figure 5.10: Normalized success rates of the estimation, equivalent of Figure 10 in [Breitenstein et al. 2008]. The database was discretized in $15^\circ \times 15^\circ$ areas and the accuracy computed for each range of angles separately. The color encodes the number of images falling in each region, as explained by the side bar. Success is declared when the nose error is below 20 mm and the angular error is below 15 degrees.

Table 5.1 reports mean and standard deviation of the errors, compared to the ones of [Breitenstein *et al.* 2008]. The first rows show mean and standard deviation for the Euclidean error in the nose tip localization task, the orientation estimation task, and for the yaw and pitch estimation errors taken singularly. The last two rows give the percentages of correctly estimated images for a threshold on the angular error of 10 degrees, and on the nose localization error of 10 millimeters. The average errors were computed from the ETH database, where our system did not return (*i.e.*, no cluster of votes large enough was found) an estimate in 0.4% of the cases, while the approach of Breitenstein failed 1.6% of the time; only faces where both the systems returned an estimate were used to compute the average and standard deviation values.

Figure 5.10 shows the success rate of the system applied to the ETH database (using 20 trees and a stride of 10) for an angular error threshold

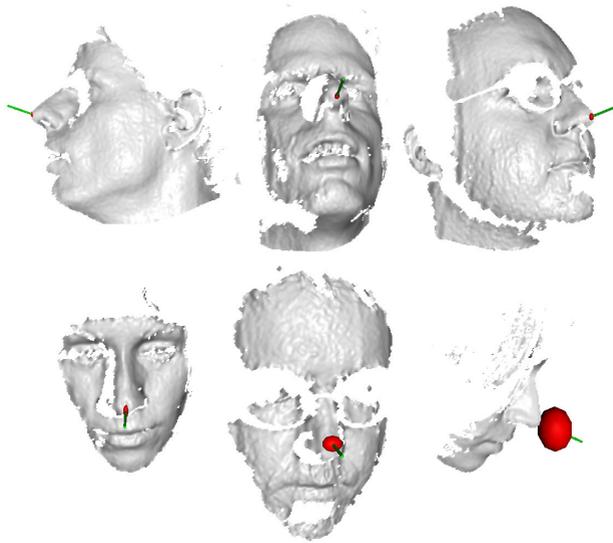


Figure 5.11: Correctly estimated poses from the ETH database. Large rotations, glasses, and facial hair do not pose major problems in most of the cases. The green cylinder represents the estimated head rotation, while the red ellipse is centered on the estimated 3D nose position and scaled according to the covariance provided by the forest (scaled by a factor of 10 to ease the visualization).

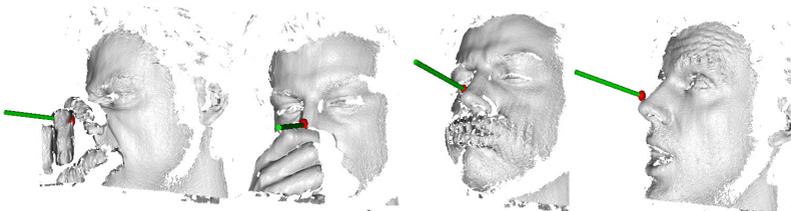


Figure 5.12: Example frames from a sequence acquired with the 3D scanner of [Weise et al. 2007]. Occlusions (even of the nose) and facial expressions can be handled by our system.

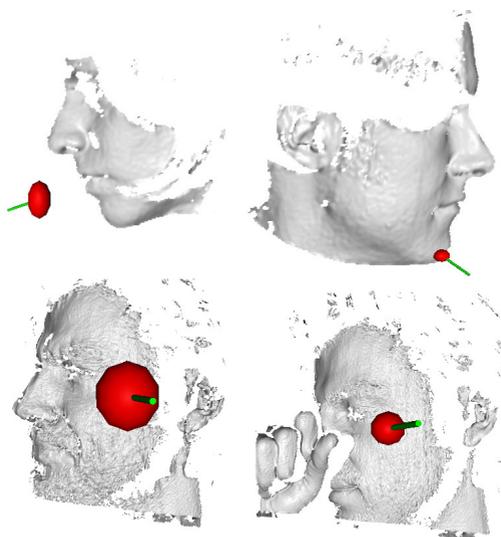


Figure 5.13: Example failure images from the ETH database. The large ellipse denotes a high variance for the estimate of the nose location.

of 15° and a nose error threshold of 20 mm. The heat map shows the database divided in $15^\circ \times 15^\circ$ bins depending on the head's pitch and yaw angles. The color encodes the amount of images in each bin, according to the side color bar. The results are 100% or close to 100% for most of the bins, especially in the central region of the map, which is where most of the images fall. Our results are comparable or superior to the equivalent plot in [Breitenstein *et al.* 2008].

Figure 5.11 shows some successfully processed frames from the ETH database. The red ellipse is placed on the estimated nose tip location and scaled according to the covariance output of the regression forest. The green cylinder stretches from the nose tip along the estimated head direction. Our system is robust to large rotations and partial facial occlusions (note the girl at the bottom right, with most of the face covered by hair, which is not reconstructed by the scanner). Additional results are shown in Figure 5.12, demonstrating how the proposed algorithm can



Figure 5.14: Example frames from the real time video, showing how the regression works even in the presence of partial occlusions, notably of the nose. Facial expressions also can be handled to a certain degree, even though we trained only on neutral faces.

handle a certain degree of facial expression and occlusion, maintaining an acceptable accuracy of the estimate.

We ran our real time system on a Intel Core 2 Duo computer @ 2GHz, equipped with 2GB of RAM, which was simultaneously used to acquire the range data as explained in [Weise *et al.* 2007]. Figure 5.14 shows some example frames from the video. Our method successfully estimates the head pose even when the nose is totally occluded and thus most of the other approaches based on 3D (*e.g.*, [Breitenstein *et al.* 2008]) would completely fail. Some degree of facial dynamics also does not seem to cause problems to the regression in many cases, even though the synthetic training dataset contains only neutral faces; only very large mouth movements like yawning result in a loss of accuracy.

Some example failures are rendered in Figure 5.13. Note how the red ellipse is usually large, indicating a high uncertainty of the estimate. These kind of results are usually caused by a combination of large rotations and missing parts in the reconstruction, *e.g.*, because of hair or occlusions; in those circumstances, clusters of votes can appear in the wrong locations and if the number of votes in them is high enough, they might be erroneously selected as the nose tip.

5.3.2 Head pose estimation - low resolution

Dataset

To train and test our head pose estimation system on low quality depth images coming from a commercial sensor like the Kinect, synthesizing a database is not an easy task. First of all, such a consumer depth camera is built specifically for being used in a living-room environment, *i.e.*, capturing humans with their full body. This means that heads are always present in the image together with other body parts, usually the torso and the arms. Because regions of the depth image different than the head are not informative about the head pose, we need examples of negative patches, *e.g.*, coming from the body, together with positive patches extracted from the face region. Lacking the human body model and MoCap trajectories employed by [Shotton *et al.* 2011], we resorted to record a new database using a Kinect. The dataset comprises 24

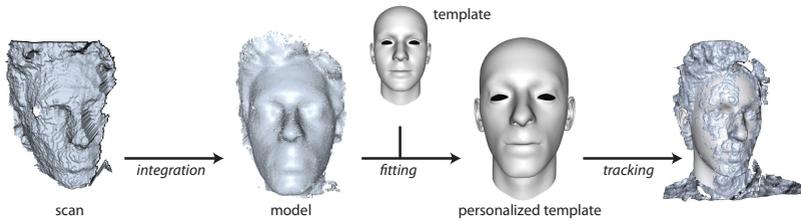


Figure 5.15: Automatic pose labeling: A user turns the head in front of the depth sensor, the scans are integrated into a point cloud model and a generic template is fit to it. The personalized template is used for accurate rigid tracking.

sequences of 20 different subjects (14 men and 6 women, 4 people with glasses) recorded while sitting about a meter away from the sensor. All subjects rotated their heads trying to span all possible ranges of yaw and pitch angles, but also some roll is present in the data.

To label the sequences with the position of the head and its orientation, we processed the data off-line with a state-of-the-art template-based head tracker [Weise *et al.* 2011]⁴, as illustrated in Figure 5.15. A generic template was deformed to match each person’s identity as follows. First, a sequence of scans of the users’ neutral face recorded from different viewpoints were registered and fused into one 3D point cloud as described by [Weise *et al.* 2009b]. Then, the 3D morphable model of [Paysan *et al.* 2009] was used, together with graph-based non-rigid ICP [Li *et al.* 2009], to adapt the generic face template to the point cloud. Each sequence was thus tracked with the subject’s template using ICP [Besl and McKay 1992], obtaining as output for each frame the 3D location of the head (and thus of the nose tip) and the rotation angles.

Using such automatic method to acquire the ground truth for our database allowed us to annotate over 15K frames in a matter of minutes. Moreover, we found that the mean translation and rotation errors were around 1 mm and 1 degree respectively. Please note that such personalized face model is only needed for labeling the training data: Our head pose

⁴Commercially available: <http://www.faceshift.com>

estimation system does not assume any initialization phase nor person-specific training, and works on a frame-by-frame basis.

The resulting Biwi Kinect Head Pose Database contains head rotations in the range of around $\pm 75^\circ$ for yaw, $\pm 60^\circ$ for pitch, and $\pm 50^\circ$ for roll. Faces are 90x110 pixels in size on average. Besides the depth data which we used for our algorithm, the corresponding RGB images are also available, as shown in Figure 5.16.

Experiments

Training patches must now be distinguished between positives (extracted from the head region) and negatives (belonging to other body parts). When we randomly extracted patches from the Biwi Kinect Head Pose Database, we labeled them as positive only if the Euclidean distance between the 3D point falling at the center of the patch and the closest point on the face model used for annotation was below 10 millimeters. In this way, negative patches were extracted not only from the torso and the arms, but also from the hair. Figure 5.17 shows this process.

In the following experiments, unless explicitly mentioned otherwise, all training and testing parameters are kept the same as in the previous evaluation done on high resolution scans. We only reduce the size of the patches to 80x80 because the heads are smaller in the Kinect images than in the 3D scans. Furthermore, we extract 20 negative patches per training image in addition to the 20 positive patches. For testing, patches ending in a leaf with $p(c|\mathcal{P}) < 1$ and $\text{tr}(\Sigma^1) \geq \text{max}_v$ are discarded. Given the much lower quality of the depth reconstruction, using the geometric normals as additional features does not bring any improvement to the estimation, therefore we only use the depth channel in this section. Because the database does not contain a uniform distribution of head poses, but has a sharp peak around the frontal face configuration, as can be noted from Figure 5.22, we bin the space of yaw and pitch angles and cap the number of images for each bin.

In Section 5.2.2, we described different ways to train forests capable of classifying depth patches into head or body and at the same time estimating the head pose from the positive patches. In order to compare the discussed training strategies (*interleaved*, *linear*, and *exponential*),

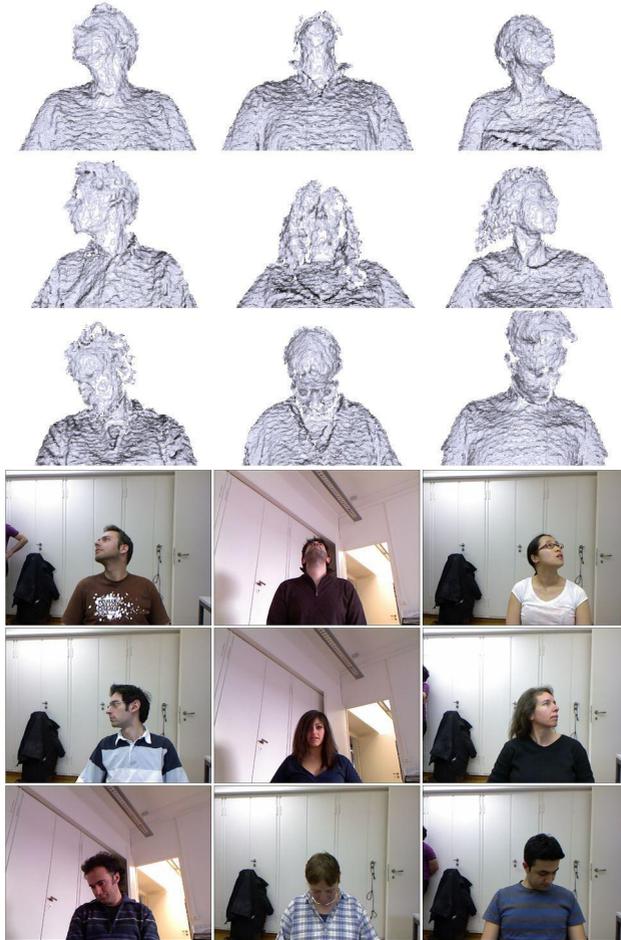


Figure 5.16: Example frames from the Biwi Kinect Head Pose Database. Both depth and RGB images are present in the dataset, annotated with head poses. In this paper, we only use the depth images for the head pose estimation algorithm.

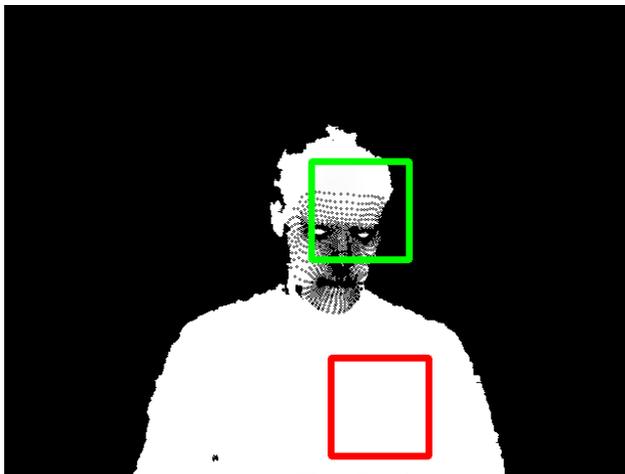


Figure 5.17: Training patches extracted from the annotated depth images of the Biwi Kinect Head Pose Database acquired with a Microsoft Kinect. The green box represents a positive patch, while the red one is an example of a negative patch. The dark dots on the face represent the model's vertices used to define the patch label: Only if the center of the patch falls near such vertices, the patch is considered as positive.

we divided the database into a testing and training set of respectively 2 (persons number 1 and 12) and 18 subjects.

Depending on the method used to combine the classification and regression measures, additional parameters might be needed. In the *interleaved* setting [Gall *et al.* 2011], each measure is chosen with uniform probability, except at the two deepest levels of the trees where the regression measure is always used. For the *linear* weighting approach (cf. Equation 5.7), we set α and t_p as suggested by [Okada 2009], namely to 1.0 and 0.8. For the *exponential* weighting function based on the tree depth (cf. Equation 5.8), we used λ equal to 2, 5, and 10. All comparisons were done with a forest of 20 trees and a stride of 10.

The success rate of the algorithm is shown in Figure 5.18(a), as the max_v parameter increases, *i.e.*, as more and more leaves are allowed to vote. Success means that the detected nose tip is within 20mm from the ground truth location, and that the angular error is below 15° . Fig-

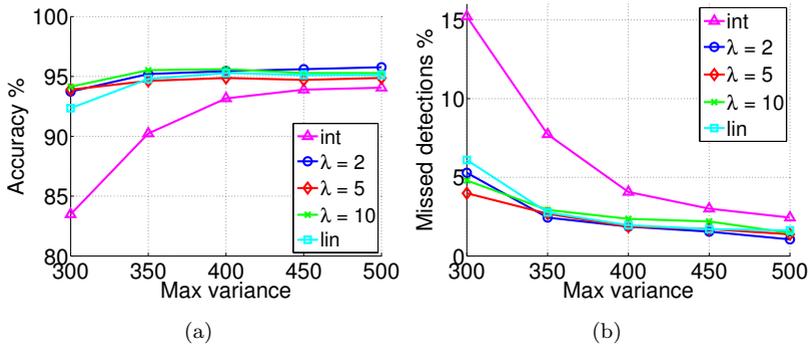


Figure 5.18: Accuracy (a) of the tested methods as a function of the maximum variance parameter, used to prune less informative leaves in the forest. Success is defined when the nose estimation error is below 20mm and the thresholds for the orientation estimation error is set to 15 degrees. The plots in (b) show the percentage of images for which the system did not return an estimate (false negatives), again as a function of the maximum variance. It can be noted that the evaluated methods perform rather similarly and the differences are small, except for the interleaved scenario, which consistently performs worse.

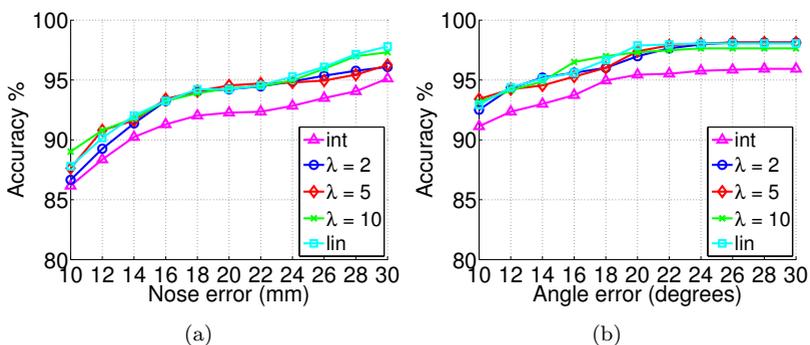


Figure 5.19: Accuracy for the nose tip estimation error (a), respectively the angle error (b) of the tested methods. The curves are plotted for different values of the threshold defining success.

Stride	5	10	15
Nose (mm)	12.2 ± 22.8	12.6 ± 23.4	13.4 ± 26.9
Dir.($^{\circ}$)	5.9 ± 8.1	6.1 ± 8.8	6.4 ± 9.4
Yaw ($^{\circ}$)	3.8 ± 6.5	4.0 ± 7.1	4.2 ± 7.8
Pitch ($^{\circ}$)	3.5 ± 5.8	3.6 ± 6.0	3.8 ± 6.4
Roll ($^{\circ}$)	5.4 ± 6.0	5.5 ± 6.2	5.5 ± 6.2
Missed (%)	6.6	6.5	6.5
Time (ms)	44.7	17.8	10.7

Table 5.2: Mean and standard deviation of the errors for the nose position and Euler angles estimation, together with rate of false negatives and average runtime, as functions of the stride. The values are computed by 4-fold, subject independent cross validation on the entire Biwi Kinect Head Pose Database.

ure 5.18(b) shows, again as a function of the maximum leaves' variance, the percentage of missed detections. In general, low values of the parameter max_v have a negative impact on the performance, as the number of votes left can become too small. However, reducing the maximum variance makes only the most certain votes pass, producing better estimates if there are many votes available, *e.g.*, when the face is covering a large part of the image; moreover, reducing max_v can also be used to speed up the estimation time. The parameter shows how well the different schemes minimize the classification and regression uncertainty. Because only the leaves with low uncertainties are used for voting, trees with a large percentage of leaves with a high uncertainty will yield a high missed detection rate, as shown in Figure 5.18(b). In this regards, all tested methods appear to behave similarly, except for the interleaved scenario, which consistently performs worse, indicating that the trees produced using such method had leaves with higher uncertainty. We also note that the exponential weighting scheme with $\lambda = 5$ returns the lowest number of missed detections.

The plots in Figures 5.19(a) and 5.19(b) show the success rate as function of a threshold on the nose error, respectively on the orientation error.

We note again the lower accuracy achieved by the interleaving scheme, while the other methods perform similarly.

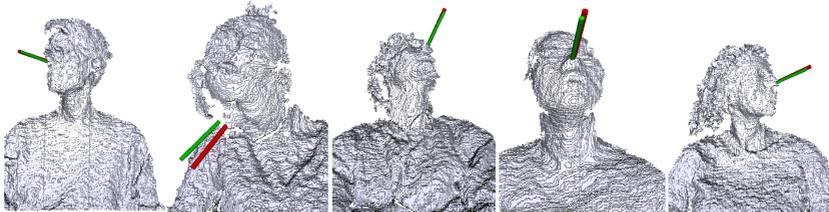


Figure 5.20: Examples of successfully estimated depth images out of our Kinect database. The green cylinder represents the estimated head pose, while the red one encodes the ground truth.

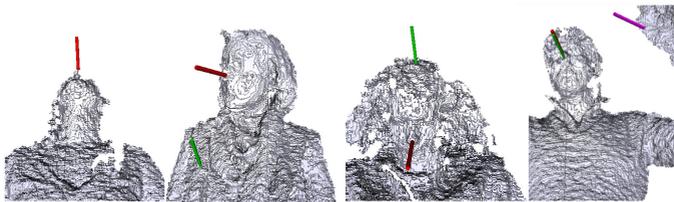


Figure 5.21: Some typical failure cases of the algorithm, showing a missed detection and two false positives. The estimated head pose is shown in green, the ground truth is in red. The magenta cylinder indicates a (wrong) second detection.

We performed a 4-fold, subject-independent cross-validation on the Biwi Kinect Head Pose Database, using an exponential weighting scheme with λ set to 5. All other parameters were kept as described earlier. The results are given in Table 5.2, where mean and standard deviation of the nose tip localization, face orientation estimation, yaw, pitch and roll errors are shown together with the percentage of missed detections and the average time necessary to process an image, depending on the stride parameter. It can be noted that the system performs beyond real time for strides greater than or equal to 10 (needing less than 20ms to process a frame on a 2.67GHz Intel Core i7 CPU, *i.e.*, running at over 50 frames per second), still maintaining a small number of missed detections and low errors. Some examples of successful estimations are

given in Figure 5.20, where the green cylinder encodes the estimated head pose, while the red one represents the ground truth.

Some typical failure cases are shown in Figure 5.21, with examples of missed detections, wrong detections, and a case of a false positive (the magenta cylinder on the hand). Apart from very large rotations, common issues of the current system include long hair covering part of the head, and distracting objects like hands or clothing. Adding more negatives samples to the training set (*e.g.*, of hands) would alleviate some of these problems.

Stride	5	10	15
Nose (mm)	19.7 ± 46.5	20.2 ± 47.3	21.7 ± 50.7
Dir.($^{\circ}$)	8.5 ± 12.9	8.7 ± 13.1	9.3 ± 14.0
Yaw ($^{\circ}$)	6.0 ± 11.5	6.2 ± 11.8	6.6 ± 12.6
Pitch ($^{\circ}$)	4.8 ± 7.1	4.9 ± 7.3	5.2 ± 7.7
Roll ($^{\circ}$)	5.8 ± 6.8	5.8 ± 6.8	6.0 ± 7.1
Missed (%)	9.3	9.2	8.7
Time (ms)	44.0	15.3	10.0

Table 5.3: Results of the cross-validation experiments, when synthetic data was used to extract positive training patches.

Figure 5.22 is the equivalent of Figure 5.10, *i.e.*, the results of the cross-validation (stride 10) are given as ratios of successfully estimated frames for each 15×15 degrees bin. Success is again declared for nose localization errors $\leq 20mm$ and angular errors $\leq 15^{\circ}$. The map is colored according to the number of images present in each bin. It can be noted how the central areas contain a lot more frames than the border ones, thus the necessity of binning the database before random sampling for training.

As a last experiment, we rendered depth images of the face templates which were used to annotate the database; see Figure 5.15. We simulated a Kinect by using the same intrinsics camera matrix. In this way, we created a dataset of synthetic depth images of heads, undergoing the

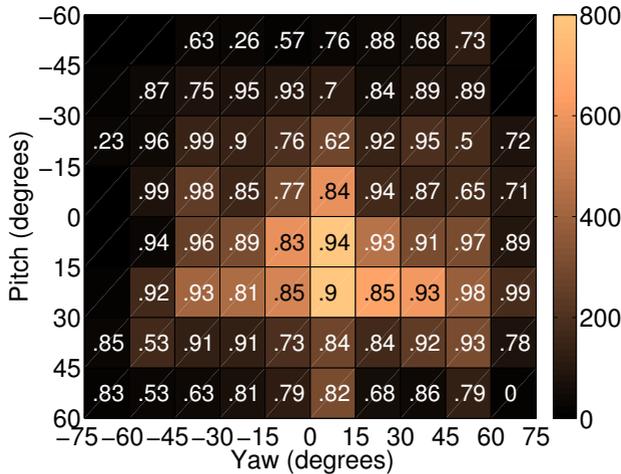


Figure 5.22: Equivalent of Figure 5.10: Normalized success rates of the estimation, (success means nose error $\leq 20\text{mm}$ and angular error $\leq 15^\circ$). The database was discretized in $15^\circ \times 15^\circ$ areas and the accuracy computed for each range of angles. The color encodes the number of images falling in each region, as explained by the side bar.

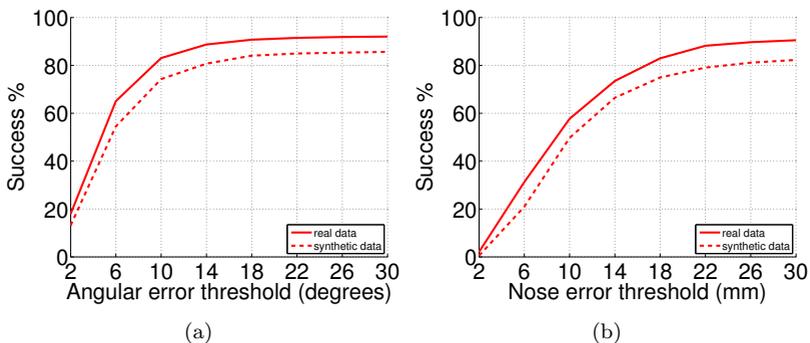


Figure 5.23: Percentage of correctly estimated images (4-fold cross validation) depending on the (a) head localization and (b) angular error thresholds. The continuous lines represent the performance when real data is used for training, while the dashed lines are the results of the forests trained on positive patches extracted from synthetically generated heads. The whole Kinect dataset is always used for testing.

same global movements as the original data. Also the identity of the templates are consistent with the recorded dataset. We thus extracted the positive patches from the synthetic data, while using the original depth data to sample negatives. Using the same settings as for Table 5.2, we achieved the results presented in Table 5.3. All errors are higher, in particular the ones related to the nose tip.

The plots in Figure 5.23 compare the accuracy of the system when trained on real data and when using the synthesized heads as positive samples. The continuous lines are the results obtained using real data, while the dashed lines represent the accuracy of the system when trained on synthetic positive samples (and tested on real data). Specifically, Figure 5.23(a) plots the success rate as a function of the orientation estimation error, while Figure 5.23(b) as a function of the nose error.

Using the synthetic heads decreased the performance, though not in a very incisive manner. The loss in performance can be explained by the incomplete head model, which does not include hair or anything below the neck as shown in Figure 5.15. The incompleteness of models for generating training data seems to be indeed a limitation of synthetic training data. Another source for the performance loss is the missing sensor noise in the synthetic data.

5.3.3 Facial features localization

Datasets

When extending the random forest framework for the purpose of facial features localization, once again a large dataset of annotated range images of faces is needed.

As a first dataset, we chose $B3D(AC)^2$, presented in Chapter 4. Depth and RGB images come together with a template of over 23K vertices, deformed to fit the specific expression. Thanks to such annotation, we could select a set of 14 facial features on the generic template and automatically extract their 3D locations from all frames in the dataset. In our facial features detection algorithm, we only use the depth images from the above database, *i.e.*, we do not rely on the RGB data.

As a second dataset, we used *BU3DFE* [Yin *et al.* 2006], which contains a larger number of subjects (100, 56 females and 44 males), and stronger facial deformations. Each subject performed the six basic expressions plus neutral in front of a 3D face scanner. Each of the six prototypic expressions (happiness, disgust, fear, angry, surprise and sadness) includes four levels of intensity, *i.e.*, there are 25 static 3D expression models for each subject, resulting in a total of 2500 faces. Because the dataset comes in form of 2.5 face models, we could render them into depth images, first without rotations, then with randomly varying the pitch, yaw, and roll angles, sampling uniformly between ± 20 degrees. All models come with manually annotated 83 facial features locations in 3D, from which we extracted the 14 fiducial which interested us: Eye, nose, and mouth corners, plus outer midpoints on the lips and the two extremes of the eyebrows.

Experiments

When building a forest for localizing facial features from range scans, we sample training patches both from the inside and the outside of the face region. A patch is considered as a positive training sample for facial feature k if the norm of the corresponding offset vector is below a threshold, *i.e.*, if $\|\theta^k\| \leq 0.2r$ where r is the radius of the average face. Since the definition of the class $c = k$ already localizes patches in a neighborhood of each feature, we use only the classification measure (5.6) for training. Using an additional regression measure did not change the performance in this setting.

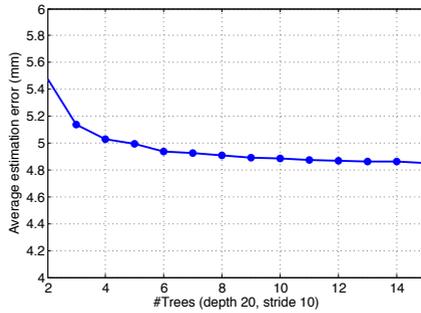
Since facial features depend more on local deformations of the face compared to the head pose, we use smaller patches of size 40x40 pixels. Since we have also more classes, we increased the depth of the trees to 20 and also the number of sampled patches for training. Each tree is built from 5000 randomly sampled images, each contributing with 50 patches, 30 extracted from within the face boundary (*i.e.*, the bounding box defined by the ground truth facial feature locations) and 20 from outside the face. During testing, each patch reaching a leaf votes for feature point k if $P(c = k | \mathcal{P}) \geq 0.5$, $\text{tr}(\Sigma^k) < \max_v^k$, and the confidence (5.11) is above a threshold. The threshold and the values \max_v^k for each facial feature point k are estimated by grid search over a validation set. In particu-

lar, we extract patches from 2000 randomly selected training images out of the $BA3D(AC)^2$ database. We only use the depth channel, without resorting to additional features like the geometric normals.

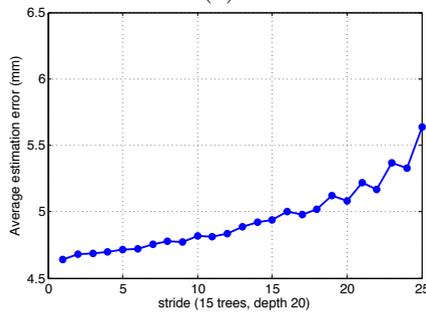
We experimentally evaluated the influence of the number of trees in the random forest, the stride, and the maximum depth of the trees. For these experiments, we trained on 12 of the subjects of the $B3D(AC)^2$ database and tested on the remaining two subjects, one man and one woman. As shown in Fig. 5.24, increasing the number of trees, letting them grow deeper, and reducing the stride, all have positive effects on the quality of the results. The plots show the mean Euclidean error, in millimeters, averaged over all the feature points and all the frames in the test set. For most of the configurations shown, the average error is below 5 millimeters. However, increased accuracy comes at the cost of a higher computation time. Fig. 5.25 shows the time in milliseconds needed to process a test image once loaded into memory (the values are averaged over 500 randomly selected frames), as a function of the number of trees, stride, and maximum depth of trees. As can be seen, for a stride of 10 pixels, we achieve real time performance, *i.e.*, frame rates above 25 fps, when loading up to 15 trees of depth 20. In all the following experiments, we thus use a forest of 15 trees, each with a maximum depth of 20 and set the stride to 10 pixels.

We further performed 5-fold, subject-independent cross validations on the $B3D(AC)^2$ database, and the $BU3DFE$ database rendered both in frontal pose and with random rotations added. Table 5.4 relates to the $B3D(AC)^2$ dataset, and shows mean and standard deviation of the errors in millimeters for all the analyzed facial features. Moreover, the success rates (for all feature points on the whole database) are given for two conservative thresholds of 10 and 5 millimeters. The outer brow corners are the points most often misplaced; this is not surprising, as the brows present limited variation in the depth channel. Tables 5.5 and 5.6 show the results of the equivalent 5-fold, subject-independent cross validation experiments on the $BU3DFE$ database in its frontal renderings, respectively on the same dataset with added rotations. We note that, for the $BU3DFE$ database, where the mouth deforms more, the lower lip midpoint is also sometimes wrongly estimated.

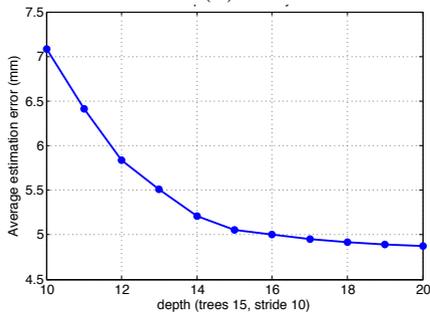
The plot in Fig. 5.26 shows the percentage of correctly estimated points for all the tested databases, as a function of the threshold defining suc-



(a)

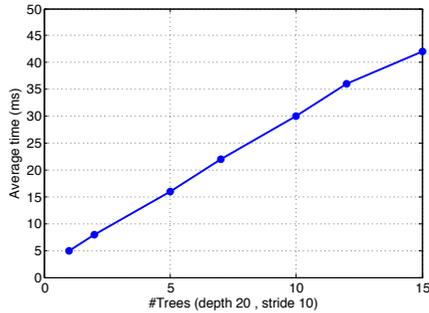


(b)

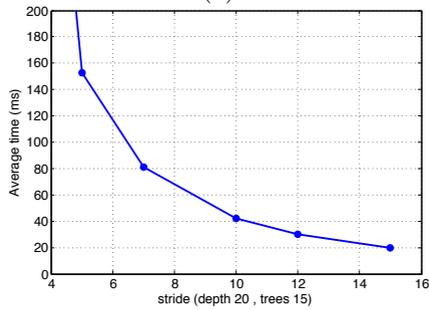


(c)

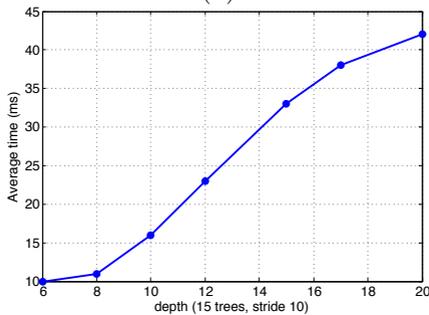
Figure 5.24: (a) Average Euclidean error for the localization of all fiducials in mm depending on the number of trees, for a stride of $t_0 = 10$ and maximum depth of 20. (b) Error depending on the stride, with 15 trees of depth 20. (c) Error depending on the trees' depth, with 15 trees and stride 10. For most configurations, the average error is below 5 mm.



(a)



(b)



(c)

Figure 5.25: Estimation time for all the 14 feature points, averaged over 500 randomly selected frames. As can be noted, the values are low and our system runs faster than 25 fps for most of the configurations. (a) Processing time depending on the number of trees, for a stride of 10. (b) Run time depending on the stride, with 15 trees. (c) Time to process a frame, depending on the maximum tree depth.

fiducial	succ. % (5/10mm)	mean \pm std
outEyeL	85.37/98.67	3.29 \pm 3.56
innEyeL	97.12/99.19	2.58 \pm 2.94
innEyeR	95.20/98.83	3.41 \pm 2.87
outEyeR	72.86/96.89	4.69 \pm 6.42
noseL	96.80/99.88	2.41 \pm 1.52
noseR	94.53/99.30	2.60 \pm 2.47
mouthL	88.88/98.97	3.04 \pm 2.15
mouthR	85.13/98.54	3.38 \pm 3.38
upLip	94.55/99.85	2.95 \pm 1.46
lowLip	86.17/98.34	3.38 \pm 2.61
outBrowL	68.31/95.56	4.50 \pm 3.66
innBrowL	93.95/98.39	2.86 \pm 3.85
innBrowR	92.50/97.83	3.34 \pm 4.16
outBrowR	77.01/94.66	4.99 \pm 7.05

Table 5.4: Summary of the performance of our method, applied to a 5-fold cross validation on the $B3D(AC)^2$ dataset, for each fiducial. Together with mean and standard deviation of the Euclidean errors, the success rates for conservative thresholds of 5, respectively 10 millimeters are shown.

cess. For the $B3D(AC)^2$ dataset, we localized the feature points with an error below or equal to 5 mm in 87.7% of the cases, which becomes 98.2% for a threshold of 10 mm. For the $BU3DFE$ database in its frontal renderings, we correctly localized 76.8% of the points for a 5 mm threshold and 96.9% for a 10 mm one; such accuracies are lower for the database with synthetically introduced rotations, namely 62.4% and 92.2%.

Some examples of successful detections of the 14 facial feature points on range images from the test datasets are shown in Figure 5.27 ($B3D(AC)^2$), and in Figure 5.28 ($BU3DFE$). Some failure examples, where not all fiducials were correctly localized, are shown in Figure 5.29. Most errors occur around the mouth regions due to the large deformations and the noisy reconstruction of the teeth and oral cavity.

As a last experiment, in order to test the performance with regard to partial occlusions and missing reconstructions, we tested our system on synthetically corrupted range images. First, we randomly selected parts

fiducial	succ. % (5/10mm)	mean \pm std
outEyeL	81.20/99.35	3.36 \pm 2.09
innEyeL	97.72/99.95	2.32 \pm 1.28
innEyeR	97.68/99.95	2.44 \pm 1.47
outEyeR	83.55/99.26	3.32 \pm 2.32
noseL	88.34/99.87	3.11 \pm 1.58
noseR	87.45/99.75	3.24 \pm 1.67
mouthL	69.38/95.53	4.46 \pm 3.25
mouthR	69.95/95.93	4.41 \pm 3.21
upLip	87.33/99.30	3.10 \pm 2.09
lowLip	75.76/95.41	4.52 \pm 5.39
outBrowL	49.49/88.10	5.90 \pm 3.64
innBrowL	71.29/98.45	4.42 \pm 2.56
innBrowR	68.77/97.68	4.59 \pm 2.74
outBrowR	47.86/88.46	6.10 \pm 3.93

Table 5.5: Summary of the performance of our method, applied to a 5-fold cross validation on the *BU3DFE* database, for each fiducial. Together with mean and standard deviation of the Euclidean errors, the success rates for conservative thresholds of 5, respectively 10 millimeters are shown.

of the depth images in the $B3D(AC)^2$ database and set them to zero, then, we rendered a hand model in front of the faces from the *BU3DFE* dataset, in order to simulate more realistic occlusions. In both cases, we trained on the original data and tested on the corrupted images, in a 5-fold cross-validation experiment.

Figure 5.30 shows the mean error, averaged over all the facial feature points, as a function of the amount of synthetically removed reconstructions on the $B3D(AC)^2$ corpus. The extent of missing data is measured as the percentage of the area covered by the face bounding box, *i.e.*, the smallest rectangle enclosing all projections on the depth image of the facial feature points ground truth locations, enlarged by the patch size (40 pixels) on both dimensions. The occluding patches are required to fall within the face bounding box and sample test faces are rendered over the curve to ease visualization. As can be seen from the plot, the proposed method is robust to such missing reconstructions: Even when

fiducial	succ. % (5/10mm)	mean \pm
outEyeL	66.01/95.93	4.66 ± 3.39
innEyeL	92.12/99.83	2.87 ± 1.71
innEyeR	91.43/99.83	2.94 ± 1.72
outEyeR	64.14/94.27	4.77 ± 4.69
noseL	81.03/99.55	3.48 ± 1.84
noseR	81.52/99.43	3.56 ± 1.92
mouthL	55.70/87.77	6.04 ± 5.44
mouthR	54.20/88.55	5.82 ± 4.77
upLip	73.04/98.05	4.05 ± 2.44
lowLip	49.61/89.52	6.45 ± 6.68
outBrowL	32.48/77.46	7.37 ± 4.22
innBrowL	51.23/92.89	5.42 ± 2.95
innBrowR	49.00/92.28	5.61 ± 3.08
outBrowR	32.19/75.39	7.71 ± 4.59

Table 5.6: Summary of the 5-fold cross validation on the BU3DFE database with rotations, for each fiducial.

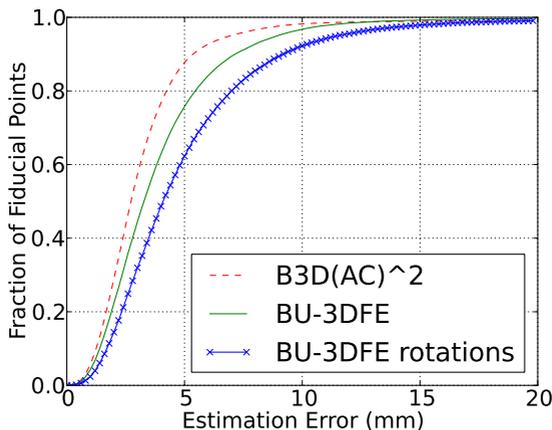


Figure 5.26: Accuracy of the algorithm (percentage of correctly estimated facial features) on all databases (5-fold cross validation), as the threshold defining success changes.

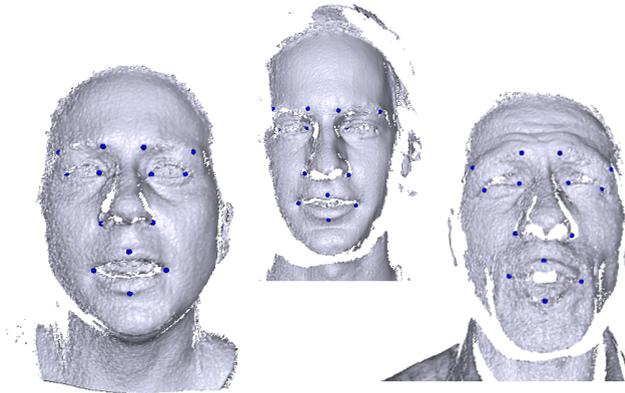


Figure 5.27: Successfully localized facial features localization on some test scans from the $B3D(AC)^2$ database.

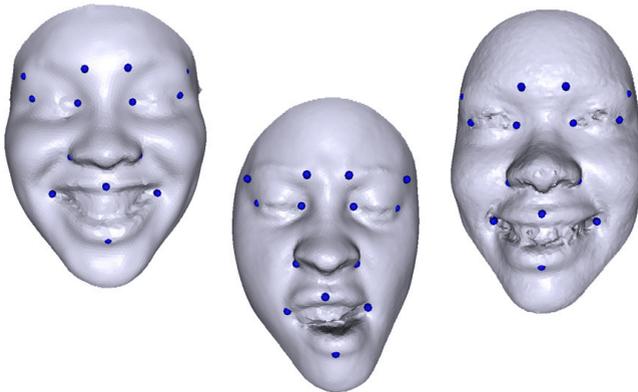


Figure 5.28: Example of successfully estimated depth images from the $BU3DFE$ dataset.

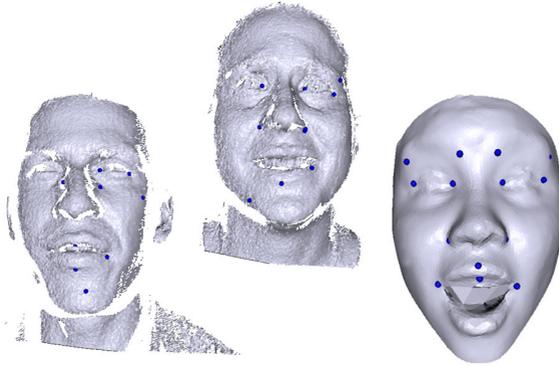


Figure 5.29: Examples failure cases for the facial feature detector. The mouth feature points and the brow's endpoints are the fiducial most often misplaced.

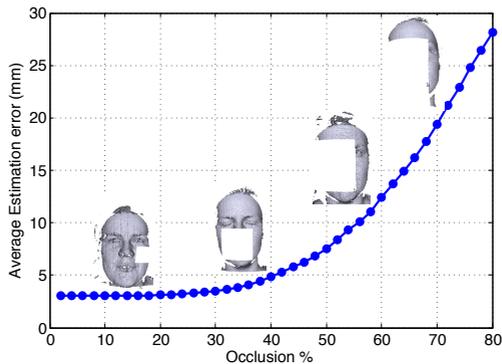


Figure 5.30: Mean errors (averaged over all the feature points) as a function of the amount of synthetically removed reconstruction from the $B3D(AC)^2$ database, measured as % of the bounding box enclosing the ground truth locations of the fiducials. Example image are overlaid on the plot, more examples are shown in Figure 5.32(a)

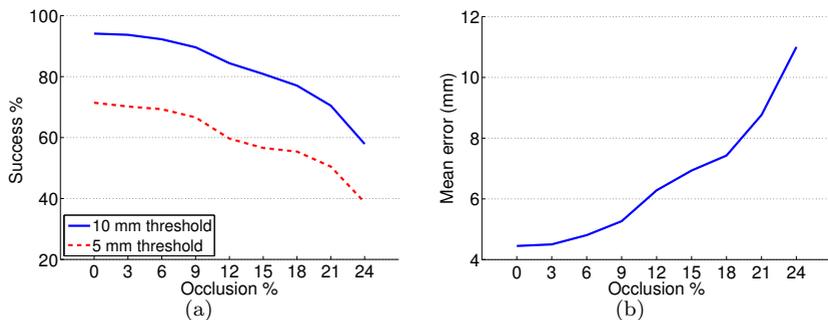


Figure 5.31: (a) Success rate, for a threshold of 10, respectively 5 mm, for the facial features localization task, plotted as functions of the percentage of face pixels occluded by the hand, in the renderings of the *BU3DFE* dataset. (b) Average errors, functions of the percentage of face pixels occluded by the hand. Examples are shown in Figure 5.32(b)

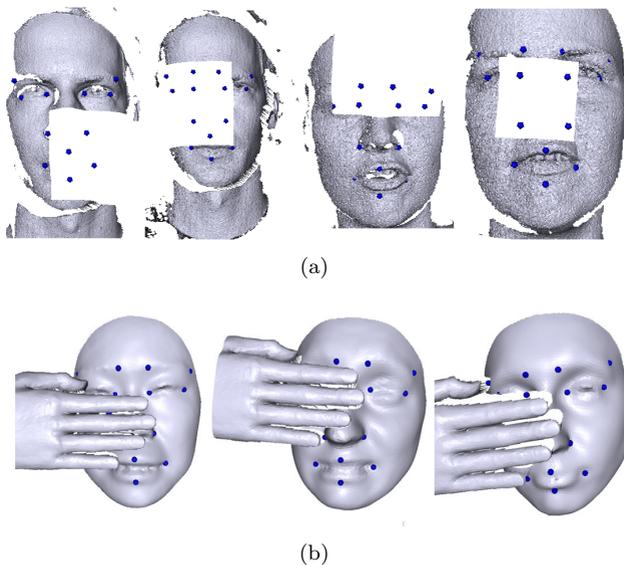


Figure 5.32: Some corrupted test images where our algorithm still manages to predict plausible locations of the feature points. (a) Missing reconstructions in the *B3D(AC)²* database. (b) Synthetically occluded images from the *BU3DFE* dataset.

50% of the data was missing, we still obtained an average error below 8 mm.

Figures 5.31(a) and 5.31(b) relate to the artificially occluded *BU3DFE* data. In particular, in Figure 5.31(a), the success rate (averaged over all feature points) is plotted against the percentage of occlusion. The blue, continuous line, relates to a threshold of 10 mm, while the red, dashed line correspond to a threshold of 5 mm. The amount of occlusion is calculated as the ratio of pixels in the face which are covered by the hand. Similarly, Figure 5.31(b) shows the average error in the localization, as the occlusion increases. As our system was trained only on positive patches coming from depth images of faces, this experiment proved more challenging than the previous one, and the errors grow faster as the hand occludes a higher percentage of the face surface.

Some examples of successful detections on corrupted images are shown in Figure 5.32(a), depicting the missing reconstructions in the *B3D(AC)*² database, and in Figure 5.32(b), with the hand-occluded renderings of the *BU3DFE* dataset.

In order to qualitatively evaluate the performance of our algorithm, we also tested it on new subjects, directly as they were scanned by the structured light scanner of [Weise *et al.* 2007]. We used a forest trained on the full *B3D(AC)*² database. We asked the subjects to perform different motions, also partly occluding their face with their hands or sunglasses. As shown in Figure 5.33, the results are robust to such occlusions. The video also shows how the algorithm is able to run in real time, at around 15 frames per second on a computer equipped with a 2GHz processor and 2GB of RAM, acquiring the range scans while estimating the 3D locations of the facial features.

5.4 Conclusions

We have proposed a fast and robust framework based on random forests for real time head movement analysis. Intuitive parameters like number of trees and sampling stride provide straight-forward tools for adapting the system to different levels of computing power availability. We described in details its application for head pose estimation using both



Figure 5.33: Qualitative results, on subjects not present in the training dataset, of the system running in real time.

high quality range scans and low resolution depth images, and for 3D facial features localization. Our method runs on a frame-to-frame basis and therefore does not suffer from the usual shortcomings of tracking approaches, lending itself as a valuable tool for (re-)initialization of such methods.

We have demonstrated the accuracy and robustness of the proposed method on challenging and realistic datasets which are available to the community. Moreover, for our experiments on real time head pose estimation from consumer depth cameras, we acquired and annotated a new database containing different subjects rotating their heads, recorded using a Microsoft Kinect, which we made available for download.

Our framework relies on the abundance of annotated training depth data. New and more realistic training databases are required, covering all the scenarios which should be expected at test time. In our future work, we intend to train on full upper body models instead of isolated faces in order to better handle hair and other non-face body parts. Synthesis of such databases is very challenging due to the need of generating different hair styles, facial expressions, and head-wears. On the other hand, acquiring and annotating real-life scenes, to be used for testing new algorithms, would probably prove even more challenging.

The use of depth data solves many of the inherent problems of standard images, however, is bounded by the availability of such sensors. Even though prices have recently dropped, the distribution of depth cameras is still limited compared to standard video recording devices and most have problems in outdoor scenarios. Our recent work [Dantone *et al.* 2012] shows how to join real time head pose estimation and facial features localization for 2D images of faces acquired “in the wild”.

6

Conclusions and Outlook

In this thesis, we have presented new tools for the automatic analysis of human behavior, with a clear focus on head and facial movements. We have attacked problems such as robust mouth localization for audio-visual speech recognition, automatic facial expression analysis, head pose estimation, and 3D facial features localization. Moreover, we have collected and annotated two new valuable datasets of affective multimodal speech and head pose estimation, which are made available to the research community.

6.1 Discussions

Here we summarize the contributions of the single chapters of thesis.

- Even though automatic speech recognition performance greatly improved in recent years [Schalkwyk *et al.* 2010], ambient noise still poses a problem in many application scenarios. The visual channel provides valuable additional cues in such cases, but at the price of having to localize the mouth area from a video stream of the speaker.

The method presented in Chapter 2 goes into this direction. Rather than relying on the detection of specific landmarks, we use random forests to map the appearance of small image patches into Hough votes for the mouth location.

Real time processing is achieved thanks to the removal of variations in scale and rotation based on the automatic detection of the irises.

Our experiments showed the goodness of the method. In particular, our automatically extracted mouth images proved competitive to those localized using manual intervention for the task of audiovisual speech recognition.

- In Chapter 3, we proposed a fully automatic system for classifying video sequences into one of the facial expressions of the six basic emotions.

We extended the Hough forest of [Gall *et al.* 2011], initially designed for human action recognition, to the harder task of recognizing the subtler movements building up facial expressions.

Our approach reached results which are comparable to the state of the art by using features separately encoding facial shape and motion, extracted from automatically normalized facial images thanks to the automatically tracked eyes' positions.

- The multimodal corpus presented in Chapter 4 is a valuable tool for the research community, not only for the analysis, but also synthesis of facial movement.

Being emotions crucial in human communication, we presented a method for the acquisition and automatic annotation of a rich multimodal database of emotional speech.

The Biwi 3D Audiovisual Corpus of Affective Communication is made available to the community, with its over 120K frames of high quality facial range scans. The faces are annotated by deforming a generic template to fit each frame: Such spatial and temporal correspondences across all sequences and speakers represent an important tool for further analysis and modeling of the face data. The audio channel is also provided, annotated with detailed phoneme segmentation, a phonological representation of the utterances, fundamental frequency, and signal intensity.

The emotional states were elicited through movie clips, which, far from being a substitute of naturalistic emotions, proved a good compromise when high-quality data are desired. The online survey which we set up to evaluate the affective content of the database confirmed the goodness our choice.

Our corpus stands out from all currently available datasets, which are either completely posed, limited to dynamic facial expressions without speech, or lacking 3D information.

- In Chapter 5, we presented an approach to head pose estimation and facial features detection from depth data.

Our algorithm is based on random forests and provides results competitive or superior to the state of the art without the need of special hardware or the visibility of specific facial features. Intuitive parameters like number of trees and sampling stride provide straightforward tools for adapting the system to different levels of computing power availability.

Real time performance and the capability of running on a frame-by-frame basis make our method valuable for many important applications, where the depth data can be key to overcome the many limitations of image-based approaches, like illumination changes or textureless facial regions. Source code and a database of annotated head poses were made available for download.

6.2 Future Work

Current automatic methods for the analysis of head and facial motions still don't stand a chance when compared to human performance. These represent important steps for many applications, especially related to human-computer interaction. Researchers in these fields face therefore many challenges in the future, together with great opportunities for advancing in the way we interact with computers.

In the following, we present possible future work to be built upon the proposed methods.

- Both automatic methods presented in Chapters 2 and 3 rely on the detection of the irises for the normalization of the facial images with respect to scale and orientation. This limits the application of the proposed algorithms to scenarios where the user is facing the camera and the eyes are visible. To limit the errors caused by temporary failure of the eye detections, *i.e.*, during blinking, a pair

of Kalman filters was used. However, more advanced techniques could be employed to better stabilize the results, *e.g.*, by tracking larger parts of the face by means of a template.

- In the facial expression recognition field, the recent trend is to recognize single muscles' activations, rather than classifying the whole face deformation into one of the six basic emotions. In fact, such prototypical expressions are rarely encountered in our daily lives. A natural extension to the work presented in Chapter 3 would thus be its application to the recognition of Action Units, as defined in the Facial Action Coding System.
- The multimodal corpus presented in Chapter 4 is already being used by several research groups around the world. Although the evaluation of the database indicates that its contents convey similar affective states to human observers as the eliciting video clips did, the used induction method is far from being a replacement of naturalism. Being the database targeted not only to the recognition of affective states, but also to the synthesis of believable emotional visual speech, the naturalness of the emotions had to be sacrificed in exchange for high quality data.

The online survey which evaluated the affective contents of the corpus was based on the original audio recordings and videos produced by rendering the tracked templates. This choice was meant to assess the quality of the processed data (brought into temporal and spatial correspondence), to be used for the training of systems aimed both at recognition and synthesis of emotional visual speech. However, the current lack of eyes, eyelids, and inner mouth in the template used to track the 3D data is a serious limitation. Being the original 3D recordings included, the corpus could serve as a common test bed for new and better face tracking algorithms.

- The framework presented in Chapter 5 is an important step forward in the fields of automatic head pose estimation and facial features localization from 3D data. The use of depth data, however, is bounded by the availability of such sensors: Even though prices have recently dropped, their distribution is still limited and most have problems in outdoor scenarios.

While our framework relies on the abundance of training data, the experiments have also shown the limitations of synthesizing such depth images using a model of the head alone. New and more realistic training databases are required, covering all the scenarios which should be expected at test time. Synthesis of such databases is very challenging due to the need of generating different hair styles, facial expressions, and head-wears. On the other hand, acquiring and annotating real-life scenes to be used for testing new algorithms would probably prove even more challenging.

Future extensions to the proposed head pose estimation system include training on full upper body models instead of isolated faces and the use of additional feature channels extracted from the RGB camera, available in most commercial devices like Kinect. Moreover, scaling the patches according to their 3D location would increase the range of operation along the z -axis.

The extension of the algorithm to facial features detection currently only works on high quality depth scans. The additional information coming from the RGB camera could prove decisive in extending the method to handle lower quality data produced by cheap sensors. Our recent publication [Dantone *et al.* 2012] shows how to join head pose estimation and facial features localization for 2D images of faces, acquired “in the wild”.

An eventual coupling of the proposed methods with tracking algorithms taking also temporal information into account would further push the state of the art in head and face motion analysis, paving the road to better applications in recognition and human computer interaction.

Bibliography

- [Aleksic and Katsaggelos 2006] P. S. Aleksic and A. K. Katsaggelos. Automatic facial expression recognition using facial animation parameters and multi-stream hmms. *Trans. on Information Forensics and Security*, 1(1):3–11, 2006. 3.1
- [Ambadar *et al.* 2009] Z. Ambadar, J. Cohn, and L. Reed. All smiles are not created equal: Morphology and timing of smiles perceived as amused, polite, and embarrassed/nervous. *Journal of Nonverbal Behavior*, 33(1):17–34, 2009. 3
- [Amberg and Vetter 2011] B. Amberg and T. Vetter. Optimal landmark detection using shape models and branch and bound slides. In *Proceedings of the International Conference on Computer Vision*, 2011. 5.1.2
- [Amit and Geman 1997] Y. Amit and D. Geman. Shape quantization and recognition with randomized trees. *Neural Computation*, 9(7):1545–1588, 1997. 2.1.1
- [Association and Corporate 1999] I. P. Association and C. A. I. Corporate. *Handbook of the International Phonetic Association : A Guide to the Use of the International Phonetic Alphabet*. Cambridge University Press, 1999. 2.5
- [Bakis 1976] R. Bakis. Continuous speech recognition via centisecond acoustic states. *The Journal of the Acoustical Society of America*, 59(1):97, 1976. 2.5
- [Balasubramanian *et al.* 2007] V. N. Balasubramanian, J. Ye, and S. Panchanathan. Biased manifold embedding: A framework for person-independent head pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007. 5.1.1

- [Ballard 1981] D. H. Ballard. Generalizing the hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981. 2.1.1
- [Banse and Scherer 1996] R. Banse and K. R. Scherer. Acoustic profiles in vocal emotion expression. *Journal of Personality and Social Psychology*, 70:614–636, 1996. 4.1
- [Bänziger and Scherer 2007] T. Bänziger and K. R. Scherer. Using actor portrayals to systematically study multimodal emotion expression: The gemep corpus. In *Proceedings of the International Conference on Affective Computing and Intelligent Interaction*, 2007. 4.1
- [Bartlett *et al.* 2005] M. S. Bartlett, G. Littlewort, M. Frank, C. Lainscsek, I. Fasel, and J. Movellan. Recognizing facial expression: Machine learning and application to spontaneous behavior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005. 3.1
- [Belhumeur *et al.* 2011] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 5.1.2
- [Beskow and Nordenberg 2005] J. Beskow and M. Nordenberg. Data-driven synthesis of expressive visual speech using an mpeg-4 talking head. In *Proceedings of the European Conference on Speech Communication and Technology*, 2005. 4.1
- [Besl and McKay 1992] P. Besl and N. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992. 5.3.2
- [Blanz and Vetter 1999] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *ACM International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 1999. 3.1, 5.1.1
- [Botsch and Sorkine 2008] M. Botsch and O. Sorkine. On linear variational surface deformation methods. *IEEE Transactions on Visualization and Computer Graphics*, 14(1):213–230, 2008. 4.2.3
- [Bourdev and Malik 2009] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *Proceedings of the International Conference on Computer Vision*, 2009. 2.1.1

- [Bradley *et al.* 1996] M. Bradley, B. Cuthbert, and P. Lang. Picture media and emotion: Effects of a sustained affective context. *Psychophysiology*, 33(6):662–670, 1996. 4
- [Bregler and Omohundro 1995] C. Bregler and S. Omohundro. Nonlinear manifold learning for visual speech recognition. In *Proceedings of the International Conference on Computer Vision*, 1995. 2.1
- [Breidt *et al.* 2011] M. Breidt, H. Buelthoff, and C. Curio. Robust semantic analysis by synthesis of 3d facial motion. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2011. 5
- [Breiman *et al.* 1984] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth and Brooks, 1984. 2.1.1
- [Breiman 2001] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001. 2.1.1, 2.4.1, 5.2
- [Breitenstein *et al.* 2008] M. D. Breitenstein, D. Kuettel, T. Weise, L. Van Gool, and H. Pfister. Real-time face pose estimation from single range images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008. 5, 5.1.1, 5.3.1, 5.9, 5.3.1, 5.1, 5.10, 5.3.1, 5.3.1
- [Breitenstein *et al.* 2009] M. D. Breitenstein, J. Jensen, C. Hoiland, T. B. Moeslund, and L. Van Gool. Head pose estimation from passive stereo images. In *Proceedings of the Scandinavian Conference on Image Analysis*, 2009. 5, 5.1.1
- [Buenaposada *et al.* 2008] J. M. Buenaposada, E. M. noz, and L. Baumela. Recognising facial expressions in video sequences. *Pattern Analysis & Applications*, 11(1):101–116, 2008. 3.1
- [Busso *et al.* 2008] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. Chang, S. Lee, and S. Narayanan. Iemocap: interactive emotional dyadic motion capture database. *Journal of Language Resources and Evaluation*, 42(4):335–359, 2008. 4.1
- [Cai *et al.* 2010] Q. Cai, D. Gallup, C. Zhang, and Z. Zhang. 3d deformable face tracking with a commodity depth camera. In *Proceedings of the European Conference on Computer Vision*, 2010. 5, 5.1.1

- [Cao *et al.* 2005] Y. Cao, W. C. Tien, P. Faloutsos, and F. H. Pighin. Expressive speech-driven facial animation. *ACM Transactions on Graphics*, 24(4):1283–1302, 2005. 4.1
- [Chang *et al.* 2006] K. I. Chang, K. W. Bowyer, and P. J. Flynn. Multiple nose region matching for 3d face recognition under varying facial expression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1695–1700, 2006. 5, 5.1.2
- [Chen *et al.* 2003] L. Chen, L. Zhang, Y. Hu, M. Li, and H. Zhang. Head pose estimation using fisher manifold learning. In *Analysis and Modeling of Faces and Gestures*, 2003. 5.1.1
- [Chen 2000] L. S.-H. Chen. *Joint processing of audio-visual information for the recognition of emotional expressions in human-computer interaction*. PhD thesis, University of Illinois at Urbana-Champaign, Champaign, IL, USA, 2000. AAI9971046. 4.1
- [Chua and Jarvis 1997] C. S. Chua and R. Jarvis. Point signatures: A new representation for 3d object recognition. *International Journal of Computer Vision*, 25(1):63–85, 1997. 5.1.2
- [Clark 1983] D. Clark. On the induction of depressed mood in the laboratory: Evaluation and comparison of the velten and musical procedures. *Advances in Behaviour Research and Therapy*, 5(1):27–49, 1983. 4
- [Cohen *et al.* 2003] I. Cohen, N. Sebe, A. Garg, L. S. Chen, and T. S. Huang. Facial expression recognition from video sequences: temporal and static modeling. *Computer Vision and Image Understanding*, 91(12):160 – 187, 2003. 3.1
- [Cohn 2006] J. F. Cohn. Foundations of human computing: facial expression and emotion. In *Proceedings of the International Conference on Multimodal Interfaces*, 2006. 3
- [Colbry *et al.* 2005] D. Colbry, G. Stockman, and A. Jain. Detection of anchor points for 3d face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005. 5.1.2
- [Cooke *et al.* 2006] M. Cooke, J. Barker, S. Cunningham, and X. Shao. An audio-visual corpus for speech perception and automatic speech recognition. *The Journal of the Acoustical Society of America*, 120(5):2421–2424, 2006. 2.1

- [Cootes *et al.* 2001] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):681–685, 2001. 3.1, 5.1.1, 5.1.2
- [Cootes *et al.* 2002] T. F. Cootes, G. V. Wheeler, K. N. Walker, and C. J. Taylor. View-based active appearance models. *Image and Vision Computing*, 20(9-10):657 – 664, 2002. 5.1.2
- [Cosker *et al.* 2011] D. Cosker, E. Krumhuber, and A. Hilton. A face valid 3d dynamic action unit database with applications to 3d dynamic morphable facial modelling. In *Proceedings of the IEEE International Conference on Computer Vision*, 2011. 4.1
- [Cowie and Cornelius 2003] R. Cowie and R. R. Cornelius. Describing the emotional states that are expressed in speech. *Speech Communication*, 40(1-2):5–32, 2003. 4, 4.1
- [Cowie *et al.* 2002] R. Cowie, E. Douglas-Cowie, N. Tsapatsoulis, G. Votsis, S. Kollias, W. Fellenz, and J. G. Taylor. Emotion recognition in human-computer interaction. *IEEE Signal Processing Magazine*, 18(1):32–80, 2002. 4
- [Cowie *et al.* 2005] R. Cowie, E. Douglas-Cowie, and C. Cox. Beyond emotion archetypes: Databases for emotion modelling using neural networks. *Neural Networks*, 18(4):371–388, 2005. 4.1
- [Craggs and Wood 2004] R. Craggs and M. M. Wood. A two dimensional annotation scheme for emotion in dialogue. In *AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications*, 2004. 4
- [Criminisi *et al.* 2010] A. Criminisi, J. Shotton, D. Robertson, and E. Konukoglu. Regression forests for efficient anatomy detection and localization in ct studies. In *Recognition techniques and applications in medical imaging*, 2010. 5.2.2
- [Criminisi *et al.* 2011] A. Criminisi, J. Shotton, and E. Konukoglu. Decision forests for classification, regression, density estimation, manifold learning and semi-supervised learning. Technical Report TR-2011-114, Microsoft Research, 2011. 2.1.1
- [Cristinacce and Cootes 2008] D. Cristinacce and T. Cootes. Automatic feature localisation with constrained local models. *Journal of Pattern Recognition*, 41(10):3054–3067, 2008. 5.1.2

- [Damasio 1995] A. R. Damasio. *Descartes' Error: Emotion, Reason, and the Human Brain*. Harper Perennial, 1 edition, 1995. 4
- [Dantone *et al.* 2012] M. Dantone, J. Gall, G. Fanelli, and L. Van Gool. Real-time facial feature detection using conditional regression forests. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012. 1.2, 5.4, 6.2
- [Darwin 1872] C. Darwin. *The Expression of the Emotions in Man and Animals*. John Murray, 1872. 3
- [De La Torre and Cohn 2011] F. De La Torre and J. F. Cohn. Facial expression analysis. In *Visual Analysis of Humans*, pages 377–409. Springer London, 2011. 3.1
- [Deng and Neumann 2007] Z. Deng and U. Neumann. *Data-Driven 3D Facial Animation*. Springer, 2007. 4
- [Dorai and Jain 1997] C. Dorai and A. K. Jain. Cosmos - a representation scheme for 3d free-form objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(10):1115–1130, 1997. 5.1.2
- [Dornaika and Davoine 2008] F. Dornaika and F. Davoine. Simultaneous facial action tracking and expression recognition in the presence of head motion. *International Journal of Computer Vision*, 76(3):257–281, 2008. 3.1
- [Douglas-Cowie *et al.* 2003] E. Douglas-Cowie, N. Campbell, R. Cowie, and P. Roach. Emotional speech: Towards a new generation of databases. *Speech Communication*, 40(1-2):33–60, 2003. 4.1
- [Douglas-Cowie *et al.* 2007] E. Douglas-Cowie, R. Cowie, I. Sneddon, C. Cox, O. Lowry, M. McRorie, J.-C. Martin, L. Devillers, S. Abrilian, A. Batliner, N. Amir, and K. Karpouzis. The humane database: Addressing the collection and annotation of naturalistic and induced emotional data. In *Proceedings of the International Conference on Affective Computing and Intelligent Interaction*, 2007. 4.1
- [Duda and Hart 1972] R. Duda and P. Hart. Use of the hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(1):11–15, 1972. 2.1.1
- [Edge *et al.* 2009] J. D. Edge, A. Hilton, and P. Jackson. Model-based synthesis of visual speech movements from 3d video. *EURASIP Journal on Audio Speech Music Processing*, 2009:4:2–4:2, 2009. 4

- [Ekman and Friesen 1971] P. Ekman and W. Friesen. Constants across cultures in the face and emotion. *Journal of Personality and Social Psychology*, 17(2):124–129, 1971. 3
- [Ekman and Friesen 1978] P. Ekman and W. Friesen. *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Consulting Psychologists Press, 1978. 3, 4.1
- [Ekman 1971] P. Ekman. Constants across cultures in the face and emotion. *Journal of Personality and Social Psychology*, 17(2):124–129, 1971. 4
- [Essa 1998] I. A. Essa. Coding, analysis, interpretation, and recognition of facial expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:757–763, 1998. 3.1
- [Everingham *et al.* 2006] M. Everingham, J. Sivic, and A. Zisserman. Hello! my name is... buffy - automatic naming of characters in tv video. In *Proceedings of the British Machine Vision Conference*, 2006. 5.1.2
- [Fanelli *et al.* 2009] G. Fanelli, J. Gall, and L. Van Gool. Hough transform-based mouth localization for audio-visual speech recognition. In *Proceedings of the British Machine Vision Conference*, 2009. 1.2
- [Fanelli *et al.* 2010a] G. Fanelli, A. Yao, P.-L. Noel, J. Gall, and L. Van Gool. Hough forest-based facial expression recognition from video sequences. In *Proceedings of the International Workshop on Sign, Gesture and Activity*, 2010. 1.2
- [Fanelli *et al.* 2010b] G. Fanelli, J. Gall, H. Romsdorfer, T. Weise, and L. Van Gool. 3d vision technology for capturing multimodal corpora: Chances and challenges. In *Proceedings of the International Workshop on Multimodal Corpora*, 2010. 1.2
- [Fanelli *et al.* 2010c] G. Fanelli, J. Gall, H. Romsdorfer, T. Weise, and L. Van Gool. A 3-d audio-visual corpus of affective communication. *IEEE Transactions on Multimedia*, 12(6):591 – 598, 2010. 1.2
- [Fanelli *et al.* 2011a] G. Fanelli, J. Gall, and L. Van Gool. Real time head pose estimation with random regression forests. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 1.2, 5, 5.2.2, 5.2.2

- [Fanelli *et al.* 2011b] G. Fanelli, T. Weise, J. Gall, and L. Van Gool. Real time head pose estimation from consumer depth cameras. In *Proceedings of the German Association for Pattern Recognition (DAGM)*, 2011. 1.2, 5, 5.2.2, 5.2.2
- [Fanelli *et al.* 2012a] G. Fanelli, M. Dantone, J. Gall, A. Fossati, and L. Van Gool. Random forests for real time 3d face analysis. *International Journal of Computer Vision*, 2012. in press. 1.2
- [Fanelli *et al.* 2012b] G. Fanelli, J. Gall, and L. Van Gool. Real time 3d head pose estimation: recent achievements and future challenges. In *Proceedings of the International Symposium on Communications, Control and Signal Processing*, 2012. 1.2
- [Fasel and Luetttin 2003] B. Fasel and J. Luetttin. Automatic facial expression analysis: a survey. *Pattern Recognition*, 36(1):259–275, 2003. 3, 3.1
- [Felzenszwalb and Huttenlocher 2005] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61(1):55–79, 2005. 5.1.2
- [Field 1987] D. Field. Relations between the statistics of natural images and the response properties of cortical cells. *Journal of the Optical Society of America*, 4(12):2379–2394, 1987. 3.3.1
- [Frigo 2006] S. Frigo. The relationship between acted and naturalistic emotional corpora. In *Proceedings of Workshop on Corpora for Research on Emotion and Affect*, 2006. 4
- [Fukushima 1980] K. Fukushima. Neocognitron: a self-organizing neural network model for mechanisms of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–202, 1980. 3.3.1
- [Galatas *et al.* 2011] G. Galatas, G. Potamianos, A. Papangelis, and F. Makedon. Audio visual speech recognition in noisy visual environments. In *Proceedings of the International Conference on Pervasive Technologies Related to Assistive Environments*, 2011. 2.1
- [Gall and Lempitsky 2009] J. Gall and V. Lempitsky. Class-specific hough forests for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009. 2, 2.1.1, 2.1.1, 3.2.1

- [Gall *et al.* 2011] J. Gall, A. Yao, N. Razavi, L. Van Gool, and V. Lempitsky. Hough forests for object detection, tracking, and action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011. 1.2, 2, 2.1.1, 2.1.1, 2.4, 2.4.1, 2.7, 3, 3.2, 3.2.1, 3.2.2, 3.5, 5.2.2, 5.3.2, 6.1
- [Girshick *et al.* 2011] R. Girshick, J. Shotton, P. Kohli, A. Criminisi, and A. Fitzgibbon. Efficient regression of general-activity human poses from depth images. In *Proceedings of the International Conference on Computer Vision*, 2011. 2.1.1
- [Gordan *et al.* 2002] M. Gordan, C. Kotropoulos, and I. Pitas. A support vector machine-based dynamic network for visual speech recognition applications. *EURASIP Journal on Advances in Signal Processing*, 2002(1):1248–1259, 2002. 2.1
- [Grabner *et al.* 2006] H. Grabner, M. Grabner, and H. Bischof. Real-time tracking via on-line boosting. In *Proceedings of the British Machine Vision Conference*, 2006. 2.2
- [Gray *et al.* 1996] M. S. Gray, J. R. Movellan, and T. J. Sejnowski. Dynamic features for visual speechreading: A systematic comparison. In *Proceedings of the International Conference on Neural Information Processing System*, 1996. 2.1
- [Grimm *et al.* 2008] M. Grimm, K. Kroschel, and S. Narayanan. The vera am mittag german audio-visual emotional speech database. In *Proceedings of the International Conference on Multimedia and Expo*, 2008. 4.1
- [Gross and Levenson 1995] J. Gross and R. Levenson. Emotion elicitation using films. *Cognition and Emotion*, 9(1):87–108, 1995. 4
- [Gross *et al.* 2005] R. Gross, I. Matthews, and S. Baker. Generic vs. person specific active appearance models. *Image and Vision Computing*, 23:1080 – 2093, 2005. 5.1.2
- [Gurban and Thiran 2009] M. Gurban and J.-P. Thiran. Information theoretic feature extraction for audio-visual speechrecognition. *IEEE Transactions on Signal Processing*, 57(12):4765–4776, 2009. 2.1, 2.6.3
- [Heckmann *et al.* 2001] M. Heckmann, F. Berthommier, and K. Kroschel. A hybrid ann/hmm audio-visual speech recognition system. In *Proceedings of the International Conference on Auditory-Visual Speech Processing*, 2001. 2.1

- [Izard 2009] C. E. Izard. Emotion theory and research: highlights, unanswered questions, and emerging issues. *Annual review of psychology*, 60(1):1 – 25, 2009. 4
- [Jaimes and Sebe 2007] A. Jaimes and N. Sebe. Multimodal human-computer interaction: A survey. *Computer Vision and Image Understanding*, 108(1-2):116–134, 2007. 4
- [Jesorsky *et al.* 2001] O. Jesorsky, K. J. Kirchberg, and R. Frischholz. Robust face detection using the hausdorff distance. In *Proceedings of the International Conference on Audio- and Video-Based Biometric Person Authentication*, 2001. 2.6
- [Jiang 2010] H. Jiang. Discriminative training of hmms for automatic speech recognition: A survey. *Computer Speech and Language*, 24(4):589 – 608, 2010. 2.1
- [Jones and Viola 2003] M. Jones and P. Viola. Fast multi-view face detection. Technical Report TR2003-096, Mitsubishi Electric Research Laboratories, 2003. 5.1.1
- [Ju *et al.* 2009] Q. Ju, S. O’keefe, and J. Austin. Binary neural network based 3d facial feature localization. In *Proceedings of the International Joint Conference on Neural Networks*, 2009. 5.1.2
- [Kakadiaris *et al.* 2007] I. A. Kakadiaris, G. Passalis, G. Toderici, M. N. Murtuza, Y. Lu, N. Karampatziakis, and T. Theoharis. Three-dimensional face recognition in the presence of facial expressions: An annotated deformable model approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4):640–649, 2007. 5.1.2
- [Kanade *et al.* 2000] T. Kanade, J. F. Cohn, and Y. Tian. Comprehensive database for facial expression analysis. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2000. 3.4
- [Kanwisher and Yovel 2006] N. Kanwisher and G. Yovel. The fusiform face area: a cortical region specialized for the perception of faces. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 361(1476):2109–2128, 2006. 1
- [Kaucic *et al.* 1996] R. Kaucic, B. Dalton, and A. Blake. Real-time lip tracking for audio-visual speech recognition applications. In *Proceedings of the European Conference on Computer Vision*, 1996. 2.1

- [Lehmann *et al.* 2011] A. Lehmann, B. Leibe, and L. Van Gool. Fast prism: Branch and bound hough transform for object class detection. *International Journal of Computer Vision*, 94(2):175–197, 2011. 2.1.1
- [Leibe *et al.* 2008] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision*, 77(1-3):259–289, 2008. 2.1.1, 2.4
- [Lepetit *et al.* 2005] V. Lepetit, P. Lagger, and P. Fua. Randomized trees for real-time keypoint recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005. 2.1.1
- [Li *et al.* 2009] H. Li, B. Adams, L. J. Guibas, and M. Pauly. Robust single-view geometry and motion reconstruction. *ACM Transactions on Graphics (Proceedings SIGGRAPH Asia 2009)*, 2009. 5.3.2
- [Li *et al.* 2012] K. Li, R. Xin, M. Wang, and H. Bai. Research of lip contour extraction in face recognition. *Advances in Electronic Engineering, Communication and Management*, 2:333–339, 2012. 2
- [Lichtenauer *et al.* 2005] J. Lichtenauer, E. Hendriks, and M. Reinders. Isophote properties as features for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005. 2.3
- [Lin *et al.* 2009] Z. Lin, Z. Jian, and L.S.Davis. Recognizing actions by shape-motion prototype trees. In *Proceedings of the International Conference on Computer Vision*, 2009. 3.1
- [Lipori 2010] G. Lipori. Manual annotations of facial fiducial points on the cohn kanade database, laiv laboratory, university of milan, 2010. <http://lipori.dsi.unimi.it/download/gt2.html>. 3.4
- [Littlewort *et al.* 2006] G. Littlewort, M. S. Bartlett, I. Fasel, J. Susskind, and J. Movellan. Dynamics of facial expression extracted automatically from video. *Image and Vision Computing*, 24(6):615 – 625, 2006. 3.1
- [Liu *et al.* 2010] X. Liu, Y. Cheung, M. Li, and H. Liu. A lip contour extraction method using localized active contour model with automatic parameter selection. In *Proceedings of the International Conference on Pattern Recognition*, pages 4332–4335, 2010. 2
- [Livescu *et al.* 2007] K. Livescu, O. Cetin, M. Hasegawa-Johnson, S. King, C. Bartels, N. Borges, A. Kantor, P. Lal, L. Yung, A. Bez-

- man, S. Dawson-Haggerty, B. Woods, J. Frankel, M. Magami-Doss, and K. Saenko. Articulatory feature-based methods for acoustic and audio-visual speech recognition: Summary from the 2006 jhu summer workshop. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, volume 4, pages IV-621 –IV-624, 2007. 2.1
- [Lu and Jain 2006] X. Lu and A. K. Jain. Automatic feature extraction for multiview 3d face recognition. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2006. 5, 5.1.1
- [Lucey *et al.* 2002] S. Lucey, Q. U. of Technology. School of Electrical, and E. S. Engineering. *Audio-visual speech processing*. PhD thesis, Queensland University of Technology, Brisbane, 2002. 2.1
- [Lucey *et al.* 2010] P. Lucey, J. Cohn, T. Kanade, J. Saragih, Z. Ambaradar, and I. Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *Proceedings of the International Workshop CVPR for Human Communicative Behaviour Analysis*, 2010. 3.4
- [Luetttin and Thacker 1997] J. Luetttin and N. A. Thacker. Speechreading using probabilistic models. *Computer Vision and Image Understanding*, 65(2):163–178, 1997. 2, 2.1
- [Maji and Malik 2009] S. Maji and J. Malik. Object detection using a max-margin hough transform. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1038–1045. IEEE, 2009. 2.1.1
- [Martin *et al.* 2009] J. C. Martin, G. Caridakis, L. Devillers, K. Karpouzis, and S. Abrilian. Manual annotation and automatic image processing of multimodal emotional behaviors: validating the annotation of tv interviews. *Personal Ubiquitous Computing*, 13(1):69–76, 2009. 4.1
- [Martins and Batista 2008] P. Martins and J. Batista. Accurate single view model-based head pose estimation. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2008. 5.1.1
- [Matthews and Baker 2003] I. Matthews and S. Baker. Active appearance models revisited. *International Journal of Computer Vision*, 60:135–164, 2003. 3.1, 5.1.2

- [Matthews *et al.* 1998] I. Matthews, J. A. Bangham, R. Harvey, and S. Cox. A comparison of active shape model and scale decomposition based features for visual speech recognition. In *Proceedings of the European Conference on Computer Vision*, 1998. 2.1
- [McGurk and MacDonald 1976] H. McGurk and J. MacDonald. Hearing lips and seeing voices. *Nature*, 264:746–748, 1976. 2
- [Mehrabian 1968] A. Mehrabian. Communication without words. *Psychology Today*, 2(9):52–55, 1968. 4
- [Mehryar *et al.* 2010] S. Mehryar, K. Martin, K. Plataniotis, and S. Stergiopoulos. Automatic landmark detection for 3d face image processing. In *Proceedings of the Congress on Evolutionary Computation*, 2010. 5.1.2
- [Mian *et al.* 2006] A. Mian, M. Bennamoun, and R. Owens. Automatic 3d face detection, normalization and recognition. In *Proceedings of the International Symposium on 3D Data Processing, Visualization, and Transmission*, 2006. 5.1.1
- [Morency *et al.* 2003] L.-P. Morency, P. Sundberg, and T. Darrell. Pose estimation using 3d view-based eigenspaces. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2003. 5.1.1
- [Morency *et al.* 2008] L.-P. Morency, J. Whitehill, and J. R. Movellan. Generalized adaptive view-based appearance model: Integrated framework for monocular head pose estimation. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2008. 5.1.1
- [Morlec *et al.* 2001] Y. Morlec, G. Bailly, and V. Aubergé. Generating prosodic attitudes in french: Data, model and evaluation. *Speech Communication*, 33(4):357–371, 2001. 4, 4.1
- [Mpiperis *et al.* 2008] I. Mpiperis, S. Malassiotis, and M. Strintzis. Bilinear models for 3-d face and facial expression recognition. *IEEE Transactions on Information Forensics and Security*, 3(3):498–511, 2008. 5.1.2
- [Mueller *et al.* 2005] P. Mueller, G. A. Kalberer, M. Proesmans, and L. Van Gool. Realistic speech animation based on observed 3d face dynamics. *IEE Proceedings Vision, Image & Signal Processing*, 152:491–500, 2005. 4

- [Murphy-Chutorian and Trivedi 2009] E. Murphy-Chutorian and M. Trivedi. Head pose estimation in computer vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(4):607–626, 2009. 5, 5.1.1
- [Nair and Cavallaro 2009] P. Nair and A. Cavallaro. 3-d face detection, landmark localization, and registration using a point distribution model. *IEEE Transactions on Multimedia*, 11(4):611–623, 2009. 5.1.2
- [Okada 2009] R. Okada. Discriminative generalized hough transform for object detection. In *Proceedings of the International Conference on Computer Vision*, 2009. 2.1.1, 2.1.1, 5.2.2, 5.3.2
- [Ommer and Malik 2009] B. Ommer and J. Malik. Multi-scale object detection by clustering lines. In *Proceedings of the International Conference on Computer Vision*, 2009. 2.1.1
- [Opelt *et al.* 2008] A. Opelt, A. Pinz, and A. Zisserman. Learning an alphabet of shape and appearance for multi-class object detection. *International Journal of Computer Vision*, 80(1):16–44, 2008. 2.1.1
- [Osadchy *et al.* 2005] M. Osadchy, M. L. Miller, and Y. LeCun. Synergistic face detection and pose estimation with energy-based models. In *Neural Information Processing Systems*, 2005. 5.1.1
- [Pachoud *et al.* 2008] S. Pachoud, S. Gong, and A. Cavallaro. Macrocuboids based probabilistic matching for lip-reading digits. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008. 2.1
- [Pandzic and Forchheimer 2002] I. Pandzic and R. Forchheimer. Mpeg-4 facial animation. *The Standard, Implementation and Applications (John Wiley & Sons, LTD, 2002)*, 2002. 3
- [Pantic *et al.* 2005] M. Pantic, M. Valstar, R. Rademaker, and L. Maat. Web-based database for facial expression analysis. In *Proceedings of the International Conference on Multimedia and Expo*, 2005. 3.4
- [Pantic 2009] M. Pantic. Machine analysis of facial behaviour: naturalistic and dynamic behaviour. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 364(1535):3505–3513, 2009. 3.1, 3.1
- [Papageorgiou *et al.* 1998] C. Papageorgiou, M. Oren, and T. Poggio. A general framework for object detection. In *Proceedings of the International Conference on Computer Vision*, 1998. 5.2.2

- [Patterson *et al.* 2002] E. K. Patterson, S. Gurbuz, Z. Tufekci, and J. Gowdy. Moving-talker, speaker-independent feature study, and baseline results using the cuave multimodal speech corpus. *EURASIP Journal on Advances in Signal Processing*, 2002(1):1189–1201, 2002. 2, 2.6, 2.6.3
- [Paysan *et al.* 2009] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter. A 3d face model for pose and illumination invariant face recognition. In *IEEE International Conference on Advanced Video and Signal Based Surveillance*, 2009. 5.2.2, 5.3.1, 2, 5.3.2
- [Petajan 1984] E. D. Petajan. *Automatic lipreading to enhance speech recognition (speech reading)*. PhD thesis, University of Illinois at Urbana-Champaign, Champaign, IL, USA, 1984. 2, 2.1
- [Picard 2000] R. W. Picard. Toward computers that recognize and respond to user emotion. *IBM Systems Journal*, 39(3-4):705–719, 2000. 4
- [Plutchik 2001] R. Plutchik. The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. *American Scientist*, 89(4):344–350, 2001. 4
- [Potamianos and Scanlon 2005] G. Potamianos and P. Scanlon. Exploiting lower face symmetry in appearance-based automatic speechreading. In *Audio-Visual Speech Processing*, pages 79–84, 2005. 2.5
- [Potamianos *et al.* 1998] G. Potamianos, H. P. Graf, and E. Cosatto. An image transform approach for hmm based automatic lipreading. In *Proceedings of the International Conference on Image Processing*, pages 173–177, 1998. 2
- [Potamianos *et al.* 2004] G. Potamianos, C. Neti, J. Luetttin, and I. Matthews. *Issues in Visual and Audio-Visual Speech Processing*, chapter Audio-Visual Automatic Speech Recognition: An Overview. MIT Press, 2004. 2, 2.1, 2.5
- [Rabiner and Juang 1993] L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition*. Prentice Hall, 1993. 2.1
- [Rabiner 1990] L. R. Rabiner. Readings in speech recognition. chapter A tutorial on hidden Markov models and selected applications in speech recognition, pages 267–296. Morgan Kaufmann Publishers Inc., 1990. 2.5

- [Ramnath *et al.* 2008] K. Ramnath, S. Koterba, J. Xiao, C. Hu, I. Matthews, S. Baker, J. Cohn, and T. Kanade. Multi-view aam fitting and construction. *International Journal of Computer Vision*, 76:183–204, 2008. 5.1.1
- [Reddy *et al.* 2009] K. K. Reddy, J. Liu, and M. Shah. Incremental action recognition using feature-tree. In *Proceedings of the International Conference on Computer Vision*, 2009. 3.1
- [Romsdorfer and Pfister 2005] H. Romsdorfer and B. Pfister. Phonetic labeling and segmentation of mixed-lingual prosody databases. In *Proceedings of Interspeech*, 2005. 4.2.4
- [Romsdorfer and Pfister 2007] H. Romsdorfer and B. Pfister. Text analysis and language identification for polyglot text-to-speech synthesis. *Speech Communication*, 49(9):697–724, 2007. 4.2.4
- [Romsdorfer 2004] H. Romsdorfer. An approach to an improved segmentation of speech signals for the training of statistical prosody models. Technical report, ETH Zurich, 2004. 4.2.4
- [Romsdorfer 2009] H. Romsdorfer. *Polyglot Text-to-Speech Synthesis. Text Analysis and Prosody Control*. PhD thesis, No. 18210, Computer Engineering and Networks Laboratory, ETH Zurich (TIK-Schriftenreihe Nr. 101), 2009. 4.2.4, 4.2.4
- [Russell 1980] J. Russell. A circumplex model of affect. *Journal of Personality and Social Psychology*, 39(6):1161, 1980. 4
- [Savran *et al.* 2008] A. Savran, N. Alyüz, H. Dibeklioglu, O. Çeliktutan, B. Gökberk, B. Sankur, and L. Akarun. Bosphorus database for 3d face analysis. In *Biometrics and Identity Management*, 2008. 4.1
- [Schalkwyk *et al.* 2010] J. Schalkwyk, D. Beeferman, F. Beaufays, B. Byrne, C. Chelba, M. Cohen, M. Kamvar, and B. Strope. Google search by voice: A case study. *Visions of Speech: Exploring New Voice Apps in Mobile Environments, Call Centers and Clinics*, 2010. 2, 2.1, 6.1
- [Scherer *et al.* 1998] K. Scherer, T. Johnstone, and T. Bänziger. Automatic verification of emotionally stressed speakers: The problem of individual differences. In *International Workshop on Speech and Computer*, 1998. 4

- [Schindler and Van Gool 2008] K. Schindler and L. Van Gool. Action Snippets: How many frames does human action recognition require? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 3.3.1
- [Schröder 2008] M. Schröder. Expressive speech synthesis: Past, present, and possible futures. In *Affective Information Processing*. Springer, 2008. 4.2.4
- [Sebe *et al.* 2006] N. Sebe, I. Cohen, T. Gevers, and T. S. Huang. Emotion recognition based on joint visual and audio cues. In *Proceedings of the International Conference on Pattern Recognition*, 2006. 4
- [Sebe *et al.* 2007] N. Sebe, M. Lew, Y. Sun, I. Cohen, T. Gevers, and T. Huang. Authentic facial expression analysis. *Image and Vision Computing*, 25(12):1856–1863, 2007. 3.1, 4
- [Sebe 2009] N. Sebe. Multimodal interfaces: Challenges and perspectives. *Journal of Ambient Intelligence and Smart Environments*, 1(1):23–30, 2009. 4
- [Seemann *et al.* 2004] E. Seemann, K. Nickel, and R. Stiefelhagen. Head pose estimation using stereo vision for human-robot interaction. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2004. 5.1.1
- [Segundo *et al.* 2010] M. Segundo, L. Silva, O. R. P. Bellon, and C. Queirolo. Automatic face segmentation and facial landmark detection in range images. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 40(5), 2010. 5.1.2
- [Shan *et al.* 2009] C. Shan, S. Gong, and P. McOwan. Facial expression recognition based on Local Binary Patterns: A comprehensive study. *Image and Vision Computing*, 27(6):803–816, 2009. 3.1
- [Shang and Chan 2009] L. Shang and K.-P. Chan. Nonparametric discriminant hmm and application to facial expression recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009. 3.1
- [Sharp 2008] T. Sharp. Implementing Decision Trees and Forests on a GPU. In *Proceedings of the European Conference on Computer Vision*, 2008. 2.1.1

- [Shotton *et al.* 2008] J. Shotton, M. Johnson, and R. Cipolla. Semantic textron forests for image categorization and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008. 2.1.1
- [Shotton *et al.* 2011] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 2.1.1, 5.3.2
- [Soleymani *et al.* 2012] M. Soleymani, J. Lichtenauer, T. Pun, and M. Pantic. A multi-modal database for affect recognition and implicit tagging. *IEEE Transactions on Affective Computing*, 3(1):42–55, 2012. 4.1
- [Storer *et al.* 2009] M. Storer, M. Urschler, and H. Bischof. 3d-mam: 3d morphable appearance model for efficient fine head pose estimation from still images. In *Proceedings of the International Workshop on Subspace Methods*, 2009. 5.1.1
- [Stratou *et al.* 2011] G. Stratou, A. Ghosh, P. Debevec, and L.-P. Morency. Effect of illumination on automatic expression recognition: A novel 3d relightable facial database. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2011. 4.1
- [Summerfield 1992] Q. Summerfield. Lipreading and audio-visual speech perception. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 335(1273):71–78, 1992. 2.7
- [Sun and Yin 2008] Y. Sun and L. Yin. Automatic pose estimation of 3d facial models. In *Proceedings of the International Conference on Pattern Recognition*, 2008. 5, 5.1.2
- [Sun *et al.* 2011] X. Sun, J. Lichtenauer, M. F. Valstar, A. Nijholt, and M. Pantic. A multimodal database for mimicry analysis. In *Proceedings of the International Conference on Affective Computing and Intelligent Interaction*, 2011. 4.1
- [Suwa *et al.* 1978] M. Suwa, N. Sugie, and K. Fujimora. A preliminary note on pattern recognition of human emotional expression. In *Proceedings of the International Joint Conference on Pattern Recognition*, 1978. 3.1

- [Synvo 2012] Synvo. <http://www.synvo.com>, 2012. 4.2.4
- [Tian *et al.* 2011] Y. Tian, T. Kanade, and J. F. Cohn. Facial expression recognition. In S. Z. Li and A. K. Jain, editors, *Handbook of Face Recognition*, pages 487–519. Springer London, 2011. 3.1
- [Tsao *et al.* 2006] D. Y. Tsao, W. A. Freiwald, R. B. Tootell, and M. S. Livingstone. A cortical region consisting entirely of face-selective cells. *Science*, 311(5761):670–674, 2006. 1
- [Türk 2001] U. Türk. The technical processing in smartkom data collection: a case study. Technical report, LMU Munich, 2001. 4.1
- [Valenti and Gevers 2011] R. Valenti and T. Gevers. Accurate eye center location through invariant isocentric patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):1, 2011. 2, 2.3
- [Valstar and Pantic 2010] M. Valstar and M. Pantic. Induced disgust, happiness and surprise: an addition to the mmi facial expression database. In *Proceedings of the International Workshop on EMOTION*, 2010. 3.4
- [Valstar *et al.* 2007] M. Valstar, H. Gunes, and M. Pantic. How to distinguish posed from spontaneous smiles using geometric features. In *Proceedings of the International Conference on Multimodal Interaction*, 2007. 4
- [Valstar *et al.* 2010] M. Valstar, B. Martinez, X. Binefa, and M. Pantic. Facial point detection using boosted regression and graph models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010. 2, 2.6, 2.6.2, 2.6, 2.6.2, 2.6.2, 5.1.2
- [Valstar *et al.* 2011] M. Valstar, B. Jiang, M. Mehu, M. Pantic, and K. Scherer. The first facial expression recognition and analysis challenge. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2011. 3.1
- [Vatahska *et al.* 2007] T. Vatahska, M. Bennewitz, and S. Behnke. Feature-based head pose estimation from images. In *Proceedings of the International Conference on Humanoid Robots*, 2007. 5.1.1
- [Velten 1968] E. Velten. A laboratory task for induction of mood states. *Behaviour research and therapy*, 6:473–482, 1968. 4

- [Viola and Jones 2004] P. Viola and M. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, 2004. 2.2
- [Vukadinovic and Pantic 2005] D. Vukadinovic and M. Pantic. Fully automatic facial feature point detection using gabor feature based boosted classifiers. In *Proceedings of the International Conference on Systems, Man and Cybernetics*, 2005. 2.1, 2, 2.6, 2.6.2, 2.6, 2.6.2, 2.6.2
- [Wampler *et al.* 2007] K. Wampler, D. Sasaki, L. Zhang, and Z. Popovic. Dynamic, expressive speech animation from a single mesh. In *Symposium on Computer Animation*, 2007. 4.1
- [Wang *et al.* 2002] Y. Wang, C. Chua, and Y. Ho. Facial feature detection and face recognition from 2d and 3d images. *Pattern Recognition Letters*, 10(23):1191–1202, 2002. 5.1.2
- [Weise *et al.* 2007] T. Weise, B. Leibe, and L. Van Gool. Fast 3d scanning with automatic motion compensation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007. 4.2, 5, 5.1.1, 5.3.1, 5.12, 5.3.1, 5.3.3
- [Weise *et al.* 2009a] T. Weise, H. Li, L. Van Gool, and M. Pauly. Face/off: Live facial puppetry. In *Symposium on Computer Animation*, 2009. 4.2.3, 5, 5.1.2
- [Weise *et al.* 2009b] T. Weise, T. Wismer, B. Leibe, and L. Van Gool. In-hand scanning with online loop closure. In *3-D Digital Imaging and Modeling*, 2009. 4.2.3, 5.3.2
- [Weise *et al.* 2011] T. Weise, S. Bouaziz, H. Li, and M. Pauly. Realtime performance-based facial animation. *ACM International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 2011. 5, 5.1.2, 5.3.2
- [Welch and Bishop 2001] G. Welch and G. Bishop. An introduction to the kalman filter. *Design*, 7(1):1–16, 2001. 2.3
- [Westermann *et al.* 1996] R. Westermann, K. Spies, G. Stahl, and F. W. Hesse. Relative effectiveness and validity of mood induction procedures: a meta-analysis. *European Journal of Social Psychology*, 26(4):557–580, 1996. 4

- [Whissell 1972] C. M. Whissell. *The dictionary of affect in language*. R. Plutchik and H. Kellerman, Reading, MA, 1972. 4
- [Whitehill and Movellan 2008] J. Whitehill and J. R. Movellan. A discriminative approach to frame-by-frame head pose tracking. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2008. 5.1.1
- [Wu *et al.* 2010] T. Wu, M. Bartlett, and J. Movellan. Facial expression recognition using gabor motion energy filters. In *Proceedings of the International Workshop on Human Communicative Behavior Analysis*, 2010. 3.1
- [Yao *et al.* 2010] A. Yao, J. Gall, and L. Van Gool. A hough transform-based voting framework for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010. 2.1.1, 3, 3.1, 3.2, 3.2.1, 3.3, 3.3.1, 3.14, 3.4
- [Yeasin *et al.* 2006] M. Yeasin, B. Bulot, and R. Sharma. Recognition of facial expressions and measurement of levels of interest from video. *Transactions on Multimedia*, 8(3):500 – 508, 2006. 3.1
- [Yin *et al.* 2004] P. Yin, I. Essa, and J. M. Rehg. Asymmetrically boosted hmm for speech reading. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2004. 2.1
- [Yin *et al.* 2006] L. Yin, X. Wei, Y. Sun, J. Wang, and M. J. Rosato. A 3d facial expression database for facial behavior research. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2006. 4.1, 5.3.3
- [Yin *et al.* 2008] L. Yin, X. Chen, Y. Sun, T. Worm, and M. Reale. A high-resolution 3d dynamic facial expression database. In *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, 2008. 4.1
- [Young *et al.* 1999] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland. *The HTK Book*. Entropic Ltd., Cambridge, 1999. 2.1, 2.5, 2.6.3, 1
- [Yu and Moon 2008] T.-H. Yu and Y.-S. Moon. A novel genetic algorithm for 3d facial landmark localization. In *Biometrics: Theory, Applications and Systems*, 2008. 5.1.2

- [Zara *et al.* 2007] A. Zara, V. Maffiolo, J.-C. Martin, and L. Devillers. Collection and annotation of a corpus of human-human multimodal interactions: Emotion and others anthropomorphic characteristics. In *Affective Computing and Intelligent Interaction*, 2007. 4.1
- [Zeng *et al.* 2007] Z. Zeng, J. Tu, M. Liu, T. S. Huang, B. Pianfetti, D. Roth, and S. E. Levinson. Audio-visual affect recognition. *IEEE Transactions on Multimedia*, 9(2):424–428, 2007. 4
- [Zeng *et al.* 2009] Z. Zeng, M. Pantic, G. I. Roisman, and T. S. Huang. A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(1):39–58, 2009. 3.1, 4, 4.1
- [Zhang *et al.* 2004] L. Zhang, N. Snavely, B. Curless, and S. M. Seitz. Spacetime faces: High-resolution capture for modeling and animation. In *ACM Annual Conference on Computer Graphics*, 2004. 4
- [Zhao and Pietikäinen 2009] G. Zhao and M. Pietikäinen. Boosted multi-resolution spatiotemporal descriptors for facial expression recognition. *Pattern Recognition Letters*, 30(12):1117–1127, 2009. 3.1
- [Zhao *et al.* 2011] X. Zhao, E. Dellandréa, L. Chen, and I. Kakadiaris. Accurate landmarking of three-dimensional facial data in the presence of facial expressions and occlusions using a three-dimensional statistical facial feature model. *IEEE Transactions on Systems, Man, and Cybernetics*, 41(5):1417–1428, 2011. 5.1.2