A Connexionist Approach for Robust and Precise Facial Feature Detection in Complex Scenes

Stefan Duffner and Christophe Garcia
France Telecom Research & Development
4, rue du Clos Courtel
35512 Cesson-Sévigné, France
{stefan.duffner, christophe.garcia}@francetelecom.com

Abstract

We present a technique for robustly and automatically detect a set of user-selected facial features in images, like the eye pupils, the tip of the nose, the mouth centre, etc. Based on a specific architecture of heterogeneous neural layers, the proposed system automatically synthesises simple problem-specific feature extractors and classifiers from a training set of faces with annotated facial features. After training, the facial feature detection system acts like a pipeline of simple filters that treats the raw input face image as a whole and builds global facial feature maps, where facial feature positions can easily be retrieved by a simple search for global maxima. We experimentally show that our method is very robust to lighting and pose variations as well as noise and partial occlusions.

1. Introduction

Automatic facial feature detection is becoming a very important task in applications such as model-based video coding, facial image animation, face recognition, facial emotion recognition, visual speech understanding, and intelligent human-computer interaction.

Numerous approaches for facial feature detection have been proposed in the last decade. Most of them use independent facial feature detectors. These detectors generally rely on hand-designed filters that aim at segmenting visual features using image properties such as edges, intensity, colour, motion, or generalised measures [16, 15]. Other approaches are based on statistical template matching or MLP-based classifiers where several correlation templates are used to detect potential facial features (eigenfeatures [11]). The detected visual features are then selected using a global concept of face through constellation analysis using face geometry constraints [7, 9]. Active Appearance Models [3] (AAMs) have also been recently used to predict facial feature locations, by attempting to match a face model to an unseen face through adaptation of the parameters of a linear model which combines shape and texture. Compared to most previous approaches, AAMs have the advantage of embedding learnt geometrical (shape) constraints during facial feature detection, but they rely on an unstable optimisation procedure which depends on hundreds of parameters encoding shape and texture variations. The main drawback of these approaches is that the performance of independent feature detection or linear face model matching is significantly influenced by noise, occlusions, and especially changes in illumination conditions.

In this paper, we propose a novel neural-based facial feature detection scheme that is designed to precisely locate facial features in faces of variable size and appearance, rotated up to ± 30 degrees in image plane and turned up to ± 60 degrees, in complex real world images. The proposed system processes face images automatically extracted by a face detector [6], i.e. faces that are not perfectly centred and undergo slight scale and pose variations. It consists of several neural network components forming a pipeline of image transformations. As all components are sequentially connected, the system can be trained by simply presenting input image and desired output, i.e. true feature positions. Global constraints encoding the face model are automatically learnt and implicitly used in the detection process.

The remainder of the paper is organised as follows. In section 2, we describe the architecture of the proposed facial feature detector. In sections 3 and 4, we explain the way we train and apply the facial feature detector. In section 5, we assess the efficiency of our approach by analysing its precision and its sensitivity with respect to noise and level of occlusion. Some experimental results obtained on different international data sets are also presented to demonstrate the effectiveness and robustness of the proposed approach. Finally, conclusions are drawn in section 6.

2. Architecture of the facial feature detector

The proposed neural architecture is a specific type of neural network consisting of six layers where the first layer is the input layer, the three following layers are convolutional layers and the last two layers are standard feed-forward neuron layers. The aim of the system is to learn

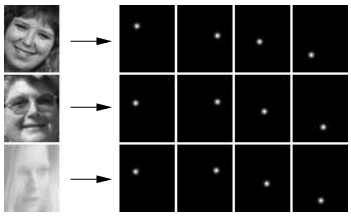

Figure 1. Some input images and desired output feature maps (right and left eye, nose tip and mouth centre).

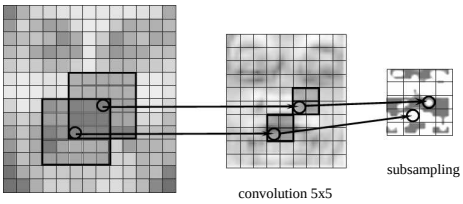

Figure 3. Example of a 5x5 convolution map followed by a 2x2 subsampling map

how to transform a raw input face image into desired output feature maps where facial features are highlighted (see Fig.1). Fig.2 gives an overview of the architecture.

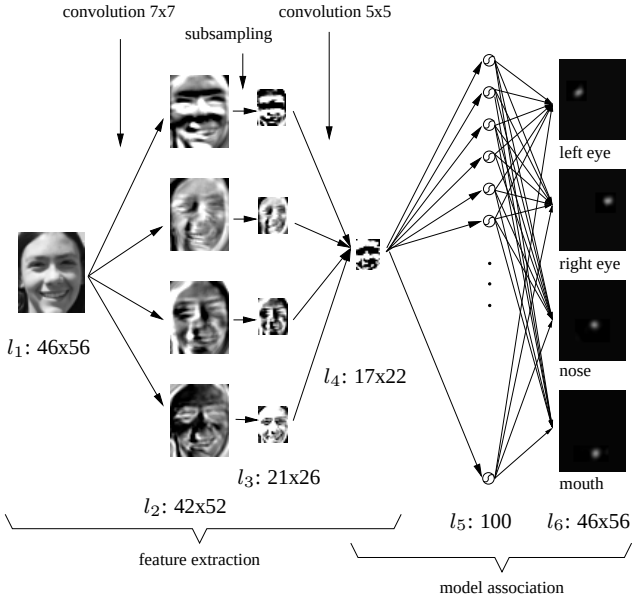

Figure 2. Architecture of the facial feature detector

The retina l_1 receives a cropped face image of 46x56 pixels, containing grey values normalised between -1 and $+1$. No further pre-processing like contrast enhancement, noise reduction or any other kind of filtering is performed.

The second layer l_2 consists of four so-called feature maps which are all connected to the input map as follows: each unit of a feature map receives its input from a set of neighbouring units of the input map (retina) as shown in Fig.3. This set of neighbouring units is often referred to as a *local receptive field*, a concept which is inspired by Hubel and Wiesel's discovery of locally-sensitive, orientation-selective neurons in the cat visual system [8]. Such local connections have been used many times in neural models of visual learning [5, 10, 12]. They allow extracting elementary visual features such as oriented edges, endpoints or corners which are then combined by subsequent layers in order to detect higher-order features. Clearly, the position of particular visual features can vary considerably in the input image because of distortions or shifts. Additionally, an elementary feature detector can be useful in several parts of the image. For this reason, each unit of a feature map shares its weights with all other units of the same feature map so that each map has a fixed feature detector. Thus, each feature map y_{2i} of layer l_2 is obtained by convolving the input map y_1 with a trainable kernel w_{2i}:

$$y_{2i}(x,y) = \sum_{(u,v)\in K} w_{2i}(u,v)\, y_1(x+u, y+v) \; + b_{2i} \; ,$$

where $K = \{(u,v) \,|\, 0 \le u < s_x \text{ and } 0 \le v < s_y\}$ and $b_{2i} \in \mathbb{R}$ is a trainable *bias* which compensates for lighting variations in the input. In our system, the four feature maps of the second layer perform each a different 7x7 convolution ($s_x = s_y = 7$). Note that the size of the obtained convolutional maps in l_2 is smaller to avoid border effects in the convolution.

The third layer l_3 subsamples its input feature maps into maps of reduced dimension by locally averaging neighbouring units. In fact, it performs a convolution of the preceding feature maps y_{2j} with a 2x2 kernel with identical weights w_{3j} (Fig.3). A trainable bias b_{3j} is added and, unlike the second layer, a sigmoid activation function $\Phi(x) = \arctan(x)$ is applied:

$$y_{3j}(x,y) = \Phi\Big(w_{3j} \sum_{(u,v)\in\{0,1\}^2} y_{2j}(2x+u, 2y+v) + b_{3j}\Big).$$

The goal of this layer is to make the system less sensitive to small shifts, distortions and variations in scale and rotation of the input at the cost of some precision.

Layer l_4 is another convolutional layer and consists of only one feature map. Basically, it works in the same way as the second layer but performs 5x5 instead of a 7x7 convolutions. Furthermore, it combines the convolution results of the four preceding subsampling maps into one feature map; it extracts higher-level features by fusing the results of the low-level feature detectors.

While the previous layers act principally as feature extraction layers, layers l_5 and l_6 transform the local information into a more global model. Layer l_5 is composed of a reduced number of neurons fully connected to layer l_4 and is dedicated to learn models (or constellations) of features and to activate the targeted positions in the output feature maps. This part of the network was inspired by *auto-associative* neural networks which are trained to reproduce an input (pattern) by means of a hidden layer containing much less

Figure 4. Virtual images created by applying various geometric transformations

neurons than the input dimension. It has been shown that auto-associative neural networks effectively perform a dimensionality reduction equivalent to the one produced by a Principal Component Analysis (PCA). In our case, we do not want to *reproduce* the input but instead to *associate* the output of the feature map in layer l_4 with the desired output of layer l_6. In that way, we only allow the activation of certain constellations of features in layer l_6. This global processing step makes the system less sensitive to partial occlusions and noise, e.g. if one eye is not visible, its position is inferred by the positions of other visible local features by activating the most likely constellation.

Layer l_5 contains 100 neurons and the output layer l_6 is composed of four feature maps, one for each feature that is to be detected. These maps have the same dimensions as the image at the input layer (46x56) and are fully connected to the preceding neurons. Sigmoid activation functions are used for both layers.

3. Training the facial feature detector

The training data set we used consists of extracted faces from the following face databases: FERET [13] (744 images), PIE [14] (1,216 images), the Yale face database [2] (165 images), BioID [4] (1,521 images), the Stirling face database [1] (185 images) as well as some face images downloaded from the internet (167 images). In total, it comprises 2,972 training and 1,026 validation face images, centred and normalised in scale. In order to make the system more robust to translation, rotation and scale, we created virtual samples of the extracted images by applying small translations (-2 and $+2$ pixels), rotation (from -20 to $+20$ degrees) and scaling (by a factor of 0.9 and 1.1). Figure 4 shows one of the training images and the respective transformed images. This procedure results in 56,468 training and 19,494 validation examples in total. The respective desired output maps are supposed to contain the value $+1$ at the feature positions and -1 everywhere else. However, in order to improve convergence, we assume that output values decrease smoothly in the neighbourhood of the feature position, thus the desired output maps are created using 2-dimensional Gaussian functions centred at the feature position and normalised between -1 and $+1$. For a particular feature map o having its desired feature at position (μ_x, μ_y), the desired function is as follows:

$$o(x,y) = 2e^{-\frac{1}{2}\left(\frac{(x-\mu_x)^2}{\sigma_x^2}+\frac{(y-\mu_y)^2}{\sigma_y^2}\right)} - 1$$

In our experiments, we set the variances $\sigma_x^2 = \sigma_y^2 = 2$.

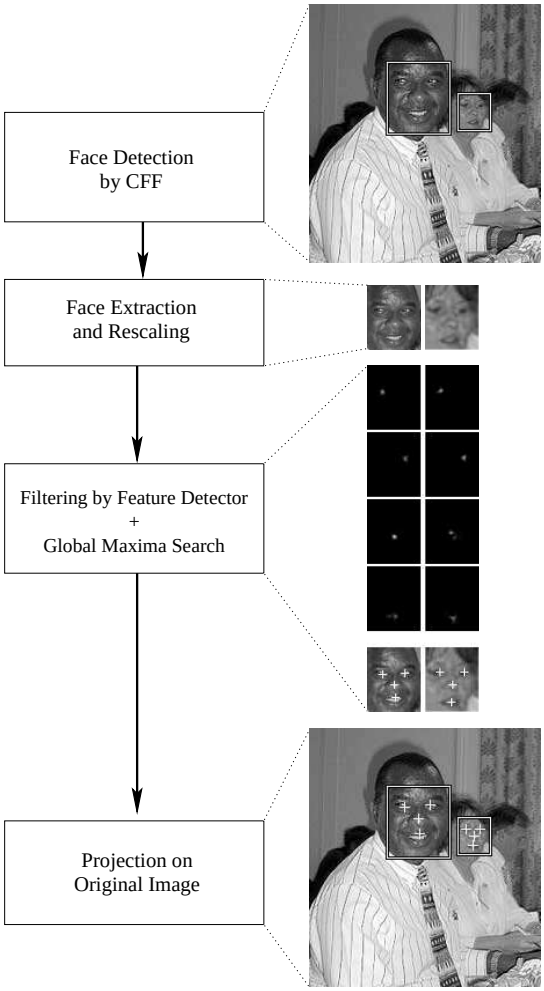

Figure 5. Principal steps of facial feature detection

The training phase was performed using the backpropagation algorithm which has been adapted in order to account for weight sharing in the convolution layers (l_2 and l_4). Additionally, a momentum term was used in the neuron layers (l_5 and l_6). At each iteration, every face image of the training set is presented to the system and the weights are updated accordingly (stochastic training). Classically, in order to avoid overfitting, after each training iteration, a validation phase is performed using the validation set. A minimal error on the validation set is supposed to give the best generalisation and the corresponding weight configuration is stored. We tried two alternative error criteria:

- the mean-squared error (MSE) between the values of the output maps and the respective values of the desired output maps, i.e. the error is calculated neuron by neuron,
- the mean-squared Euclidian distance between the four output features and the four respective desired output features.

In our experiments, we noticed that the latter leads to slightly better results.

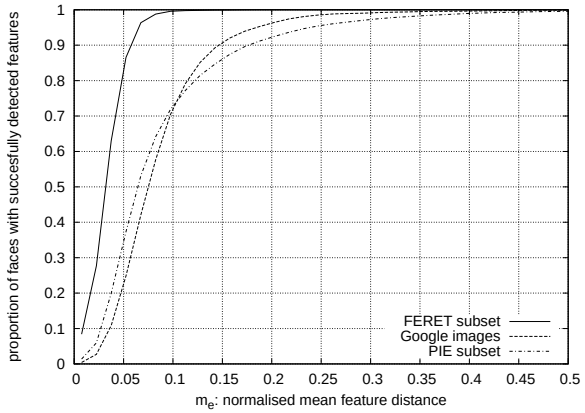

Figure 6. Detection rate of the four features versus m_e

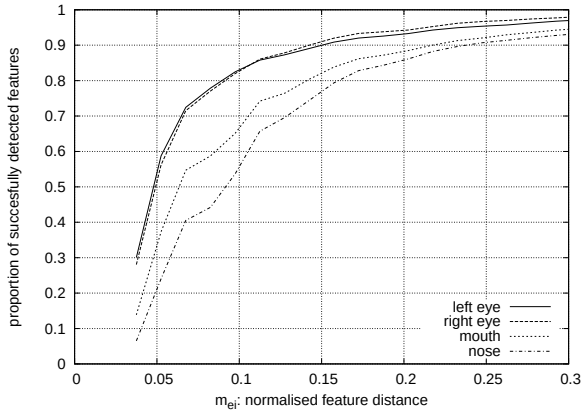

Figure 7. Detection rate of each facial feature versus m_{ei}

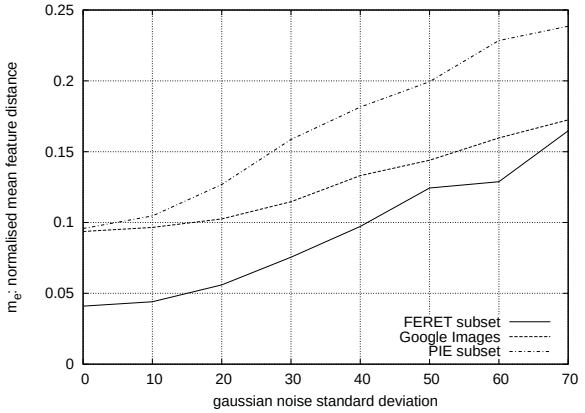

Figure 8. Sensitivity analysis: Gaussian noise

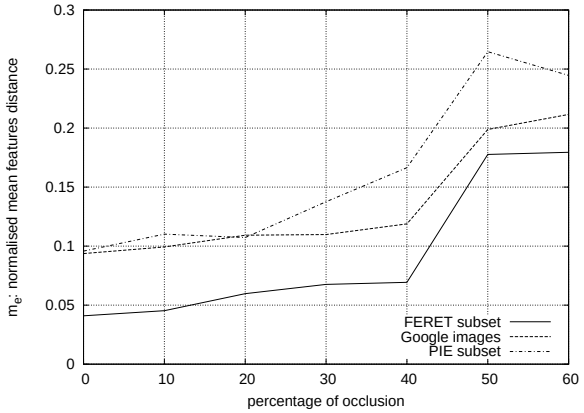

Figure 9. Sensitivity analysis: occlusion

4. Searching for facial features

Figure 5 illustrates the principal phases of the feature detection system. As mentioned before, we made use of the "Convolutional Face Finder" by C. Garcia and M. Delakis [6] to detect faces and find the respective face bounding boxes in the input images. The extracted faces are resized to the retina size and passed to the trained feature detector. The feature positions in the resized face image can directly be inferred by simply searching the maxima in the four output maps.

As the face bounding boxes may be imprecise, the last steps are repeated for slightly translated and scaled face image regions. In our experiments, we achieved good results with translations by $-4, -2, 0, +2, +4$ pixels and scale factors of $0.9, 1.0$ and 1.1. Then, for each face image region, the sum of the maxima of each output map is taken as a confidence measure and the solution having the maximal sum is adopted. Finally, the feature positions are backprojected onto the original image.

5. Experimental Results

In order to measure the performance of the proposed facial feature detector, we created several test sets with annotated face images that are not contained in the training or in the validation set. They were extracted from PIE (1,226 images), FERET (1058 images) and from images from the internet (384 images). As for the training and validation sets, the test sets were augmented by small transformations of the original images, i.e. translation, rotation and scaling, leading to three sets: *PIE subset* (23,294 images), *FERET subset* (20,102 images) and *Google images* (7,296 images).

The test images were presented to the facial feature detector and, for each face image, the mean Euclidian distance m_e between the four detected feature positions and the true feature positions, normalised with respect to the inter-ocular distance d_{eyes}, was calculated:

$$m_e = \frac{1}{4} \sum_{i=1}^{4} m_{ei} \ ,$$

with

$$m_{ei} = \frac{1}{d_{eyes}} \sqrt{(x_{oi} - x_{di})^2 + (y_{oi} - y_{di})^2} \ ,$$

where (x_{oi}, y_{oi}) is the output position and (x_{di}, y_{di}) is the true position of feature i.

Fig.6 shows the proportion of faces with successfully detected features varying the allowed m_e. The FERET test set clearly gave the best detection results because there are

Figure 10. Some results of combined face and facial feature detection

practically no pose and lighting variations as opposed to the PIE test set. The Google test set additionally contains images of low quality, with noise, extreme lighting variations and partial occlusions.

Fig.7 shows, for the PIE subset, the proportion of successfully detected features for each of the four features separately while varying m_{ei}. Obviously, the detection of the eyes is more precise than the detection of the tip of the nose and the mouth. Clearly, this is due to the fact that the local appearance of the eyes varies less under different poses and lighting conditions. The detection results of the tip of the nose are the least reliable. This seems plausible because the PIE test set shows considerable variations in pose and lighting and thus considerable variation in the appearance of the nose. The mouth is also subject to strong variations due to facial expressions (e.g. smile, open/closed mouth).

Further, we conducted two experiments showing the robustness of the facial feature detector with respect to noise and occlusion. In the first experiment, we added Gaussian noise with varying standard deviation σ to the normalised face images. Fig.8 shows the mean feature error m_e with σ varying from 0 to 70. We can notice that the proposed feature detector is very robust to noise as the error m_e remains rather low while adding a considerable amount of noise. For the worst of the three test sets (PIE subset) m_e stays below 0.2 for $\sigma = 50$.

The second experiment consists in occluding a certain percentage of the face images by a black zone in the lower part. Fig.9 shows the mean feature error m_e with an occlusion from 0 to 60%. For occlusions smaller than 50%, the only invisible feature, in most of the cases, is the mouth and the error m_e remains almost constant. Larger occlusions cover both mouth and nose, which explains the abrupt increase of m_e in all of the three test sets.

Finally, we tested the performance of the whole feature detection system as described in section 4. Fig.10 shows some results obtained on various images. The images of this test set contain neither training nor validation face images used for the training of the facial feature detector. Some of them are of rather low quality or show faces in difficult poses and under difficult lighting conditions and with partial occlusions (sunglasses, bottle etc.).

6. Conclusion

We have presented a novel method for the detection of facial features in face images based on a specific type of neural network. The proposed architecture closely connects local and global transformations and allows a straightforward training by simply presenting raw input face images and desired facial feature positions. The trained system has proven to be very robust with respect to noise and partial occlusions as well as to variations in lighting and pose. We further conducted experiments combining a face detector with the proposed facial feature detector and we obtained robust results.

As future extensions, several such facial feature detectors may be combined in a hierarchical way, in order to allow more precise feature detection and/or the detection of finer facial features, like eye or mouth corners. This means that each feature position detected by the proposed facial feature detector could be passed to a finer (more specialised) feature detector which focuses on a smaller region around a specific feature.

Acknowledgement This research was supported by the European Commission under contract FP6-001765 aceMedia.

References

[1] The psychological image collection at stirling university (pics). *http://pics.psych.stir.ac.uk*.

[2] P. Belhumeur, H. J.P., and D. Kriegmann. Eigenfaces vs fisherfaces: Recognition using class specific linear projection. *IEEE Trans. on Pattern Anal. and Machine Intelligence*, 17(7):711–720, 1997.

[3] D. Cristinacce and T. Cootes. A comparison of shape constrained facial feature detectors. In *Proc. of the 6th Int. Conf. on Automatic Face and Gesture Recognition*, pages 375–380, Seoul, Korea, 2004.

[4] T. B. F. Database. *http://www.humanscan.com/support/downloads/facedb.php*.

[5] K. Fukushima. Cognitron: A self-organizing multilayered neural network. *Biological Cybernetics*, 20:121–136, 1975.

[6] C. Garcia and M. Delakis. Convolutional face finder: A neural architecture for fast and robust face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11):1408 – 1423, November 2004.

[7] C. Garcia, G. Simantiris, and G. Tziritas. A feature-based face detector using wavelet frames. In *Intern. Workshop on Very Low Bitrate Video Coding*, pages 71–76, Athens, 2001.

[8] D. Hubel and T. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *Journal of Physiology*, 160:106–154, 1962.

[9] S. Jeng, H. Yao, C. Han, M. Chern, and Y. Liu. Facial feature detection using geometrical face model: An efficient approach. *Pattern Recognition*, 31(3):273–282, 1998.

[10] Y. LeCun. Generalization and network design strategies. In R. Pfeifer, Z. Schreter, F. Fogelman, and L. Steels, editors, *Connectionism in Perspective*, Zurich, 1989.

[11] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(7):696–710, 1997.

[12] M. C. Mozer. *The perception of multiple objects: a connectionist approach*. MIT Press, Cambridge, MA, USA, 1991.

[13] P. Philips, H. Wechsler, J. Huang, and P. Rauss. The feret database and evaluation procedure for face recognition algorithms. *Image and Vision Computing*, 16(5):295–306, 1998.

[14] T. Sim, S. Baker, and M. Bsat. The cmu pose, illumination and expression (pie) database. Technical Report CMU-RI-TR-01-02, The Robotics Institute, CMU, January 2001.

[15] J.-G. Wang and E. Sung. Frontal-view face detection and facial features extraction using color and morphological operations. *Pattern Recogn. Lett.*, 20(10):1053–1068, 1999.

[16] M.-H. Yang, D. J. Kriegman, and N. Ahuja. Detecting faces in images: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(1):34–58, 2002.