PATREC 4257
26 October 2007 Disk Used
ARTICLE IN PRESS
No. of Pages 11, Model 5+

Available online at www.sciencedirect.com
ScienceDirect
Pattern Recognition Letters xxx (2007) xxx–xxx
Pattern Recognition Letters
www.elsevier.com/locate/patrec

Real-time facial feature localization by combining space displacement neural networks

Shehzad Muhammad Hanif *, Lionel Prevost, Rachid Belaroussi, Maurice Milgram

Université Pierre and Marie Curie-Paris 6, Groupe Perception et Réseaux Connexionnistes BC 252, 4 Place Jussieu, 75252 Paris Cedex 5, France

Abstract

We present in this paper a new facial feature localizer. It uses a kind of auto-associative neural network trained to localize specific facial features (like eyes and mouth corners) in orientation-free face-images (i.e. images where faces are rotated in-plane and out-of-plane). To increase localization accuracy, two extensions are presented. The first one uses space displacement neural networks instead of classical, fully-connected networks. The second one combines several specialized networks trained to deal with each face orientation. A gating network is then used for combination. Finally, a two stage localizer is presented, which increases speed. Thorough evaluation is performed; including sensitivity to identity, noise and occlusions. The mean localization error (estimated on more than 4000 test images) is about 15% and the system can perform 40 images/s.

Keywords: Face analysis; Facial feature localization; Space displacement neural network; Gating network

1. Introduction

Localization and tracking of face and facial features 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. Although these problems are usually simple tasks for the human visual system, they have proven to be difficult for machine vision. Due to changes in orientation, lightning or expression, face and facial features can have quite different appearances. In this paper, we focus on facial feature localization as the key step in feature-based face image compression or head pose estimation. Many face recognition systems are based on facial features, such as eyes, nose and mouth, and their spatial relationship. Chellappa et al. (1995) called this the constituted approach. Chin and head boundary extraction has also been addressed by Xiao and Yan (2004). Many feature detection methods have been developed in the last decade, but a wide majority concentrates on eye detection. In fact, eyes are known as the most important salient feature and one of the easiest to detect (nose appearance changes with face pose and mouth aspect with facial expression).

In this paper, we address the problem of facial features (eyes and mouth corners) localization in orientation-free (also called multi-view) face-images. There can be three kinds of head rotations: in-plane (left–right head leaning), out-of-plane (up–down nodding) and profile view. The localization problem is far more complex than in the frontal face issue. As we already developed in our lab a face localizer (Belaroussi et al., 2006), we assume that face has been already roughly localized in a cluttered image.

The paper is organized as follows. The following section is devoted to a brief overview of state-of-art methods. In Section 3, we describe the database we used for experiments. Section 4 focuses on the localization algorithm. It is a kind of auto-associative neural network trained to output a feature map, in which intensity sorted local maxima

* Corresponding author. Tel.: +33 1 4427 9673; fax: +33 1 4427 4438.
E-mail address: shehzad.muhammad@lisif.jussieu.fr (S.M. Hanif).

0167-8655/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2007.09.016

correspond to facial feature position. To increase localization accuracy, two extensions are presented. The first one uses space displacement neural networks instead of classical, fully-connected networks. The second one combines several specialized networks trained to deal with each face orientation. A gating network is then used for combination. Section 5 is devoted to experimental results. Conclusions and prospects are presented in Section 6.

2. Overview of existing methods

Existing methods can be divided into several categories. A first classification is based on the acquisition device: active infrared-based approaches (Zhu and Ji, 2005) or passive image-based approaches. Another one depends on the processed images: pre-focused images where rough feature regions have already been located or cluttered images where face detection is preceded before feature detection. A third category is based on the detection algorithm: low-level image-based approaches or high-level statistical appearance-based approaches. In order to get the best of both worlds, many algorithms combine these approaches. We present, in detail, some of these methods.

Image-based approaches use one or several low-level detectors to find specific properties (such as edge, color, and symmetry). Initial algorithms (like Xie et al., 1994) were based on edge images, while a good edge image is hard to get under uncontrolled lightning when the eye contrast is low. Toennies et al. (2002) applied Generalized Hough Transform to detect and track eyes. Feng and Yuen (2001) use three cues to detect eyes: the intensity (eye intensity is relatively low), the estimated direction of the line joining the eye centers and the result of the convolution of the image by an eye variance filter. This process generates a list of candidate eye pairs which are further validated.

Statistical appearance-based approaches can be divided into static and dynamic (active) methods. Moghaddam and Pentland (1997) applied local principal component analysis in feature images to describe them in a low-dimensional space (eigenfeatures space). Duffner and Garcia (2005) use a Convolutional Neural Network to perform facial feature detection. Viola and Jones' state-of-art face detector (2001) based on a cascade of boosted classifier has been applied to feature detection by Cristinacce and Cootes (2003). The method uses simple Haar wavelets to find optimal templates and the AdaBoost algorithm to train the detector. They demonstrate that the performance of these local detectors can be significantly improved by adding global shape constraints. Peng et al. (2005) use more discriminant features instead of Haar wavelets to improve eye detection accuracy in a similar AdaBoost process. Active methods are also widely used. Yuille et al. (1992) propose to use deformable templates to locate human eyes. They design an eye model (parameterized template) and define an energy function depending on the image texture. The eye position is found by minimizing the function through a recursive process. Recently, Active Appearance Model (Cootes et al., 1998) has also been used to predict facial feature locations, by attempting to match a face model to an unseen face through adaptation of the model shape and texture parameters. These methods are very promising but time consuming and significantly influenced by noise, occlusions, and lightening.

3. Database

We have used two database in our experimentation – LISIF database and ECU database. The LISIF database contains images of 37 individuals with various ages, genders, and ethnicities. Images were taken under controlled lightning. For each person, we took 36 images with several facial orientations (in-plane and out-of-plane), expressions, and "accessories" like beard or glasses (Fig. 9). The original resolution is 100×100 pixels. In order to increase the number of data, we computed the mirroring image. This procedure results in a 2750 example dataset. The ECU database contains more than 3500 images of different persons with complex background and images are taken under different light conditions. Using ground truth data (rectangular face region), we extracted face-images and after mirroring we obtained a dataset of more than 7000 face-images.

We clicked manually four facial features, left eye (1st feature), right eye (2nd feature), left mouth corner (3rd feature) and right mouth corner (4th feature) to create one feature map F for each face image. This feature map had the size of the face image and its pixels have the following value (where x_{oi} and y_{oi} denote the true feature coordinates):

–At the feature location: $F(x_{oi}, y_{oi}) = +1$
–Anywhere else: $F(i, j) = -1$

To normalize input images (Fig. 1), we performed histogram equalization. To normalize feature maps, we convolved these images with a 3×3 gaussian filter, which results in smoothing feature maps. Several sub-sampling were tested to reduce the data dimension and, thus the number of parameters to be trained.

Facial feature are not randomly organized (except in Picasso's paintings perhaps). So, we can get anthropomorphic information about their spatial organization by analyzing feature coordinates (x_{oi}, y_{oi}). Assuming the feature coordinates joint density distribution is gaussian, we can evaluate its parameters: mean (8 parameters) and covariance matrix (36 parameters) by using Maximum Likelihood estimator. Assuming this density is monovariate, this estimation can be done on the whole dataset and leads to orientation-free parameters. To take into account the face orientation, we assume that feature density distribution can be modeled by a mixture of gaussians, one for each face orientation. In this latter case, we estimate parameters on a given cluster. To perform self-supervised

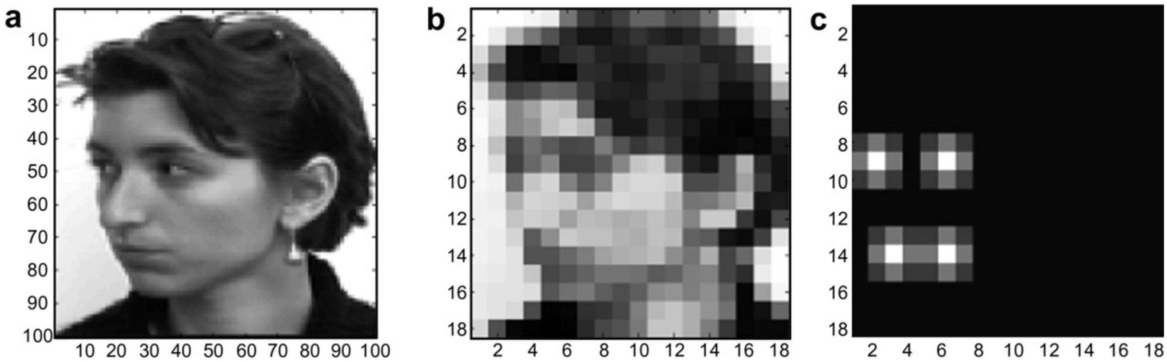

Fig. 1. Normalization process: original image (a), sub-sampled input image (b), sub-sampled and smoothed feature map (c).

orientation clustering, we assumed that there exists a unique relationship between 2D facial feature location and 3D face pose. So, knowing the facial feature localization allowed predicting the face orientation. The Expectation–Maximization algorithm is applied to get clusters with K-Means initialization and 1000 training epochs. We applied this procedure considering up to six face orientations. As can be seen for five clusters (Fig. 2), the clustering had roughly separated the whole database in subsets, each one corresponding to a certain orientation.

4. Neural localizer

4.1. Hybrid auto-associative network

It is a completely connected two-layered perceptron. The input and output layers have the same size as the desired output is equal to the input. So, the network is trained to reconstruct an output identical to its input. It implements a specialized compression as its hidden layer has much less units than input or output does. Kramer (1991) and Hsieh (2001) shown that this compression is quite similar to non-linear principal component analysis. This network was successfully used for data compression by DeMers and Cottrell (1993), handwritten character recognition by Schwenk and Milgram (1995), and face detection by Belaroussi et al. (2006) and Féraud et al. (2002). In this latter application, the network is used to model the "face-class" and trained to reconstruct face-images. Here, we do not want to reconstruct a specific pattern class (the "face-class" for example) but to localize specific features within these patterns (eyes and/or mouth corners in the face case). In other words, we want to associate an image of face (input) with a facial feature map (output). So, we used the normalized feature maps as desired output described in Section 3. The network is trained using the back-propagation algorithm with adaptive momentum. The cost function is the mean squared error between network output and desired output (Fig. 3). Once trained, the network is able to localize facial feature on unknown test images. The feature positions can directly be inferred by simply searching the local maxima in the output image and back-projected onto the original image (Fig. 4). The first four local maxima (representing eye centers and mouth corners), sorted by the intensity values, are used. Let (x_{di}, y_{di}) be the coordinates of these detected features.

4.2. Space displacement neural network

Convolutional Neural Networks – also called Space Displacement Neural Networks (SDNN) in image analysis – are slightly different networks. Instead of being fully-connected like classical MLP's, their first layer(s) have local receptive fields. Each hidden cell is just connected to a small part of the input image and connections have their own independent weight(s). The concept of local receptive

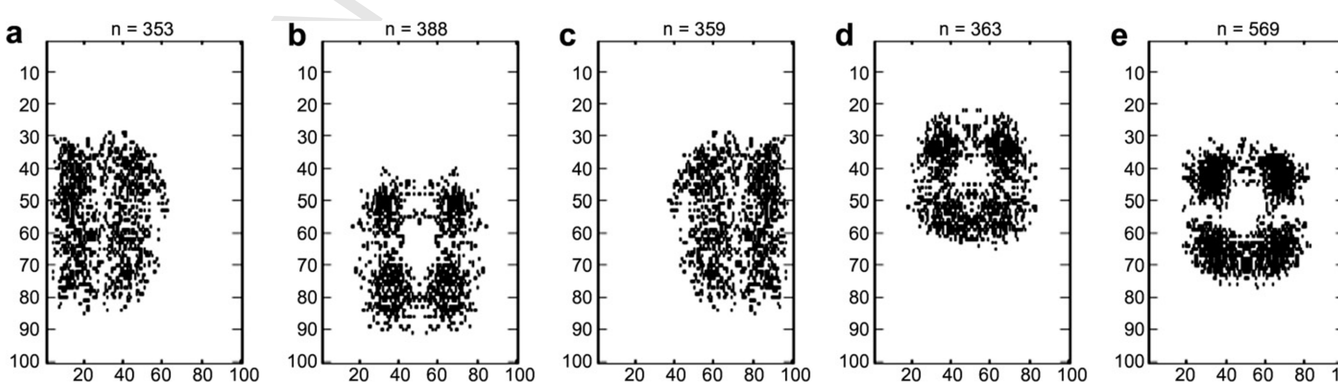

Fig. 2. Facial feature position for five clusters: left-sided (a), downward (b), right-sided (c), upward (d), and frontal (e).

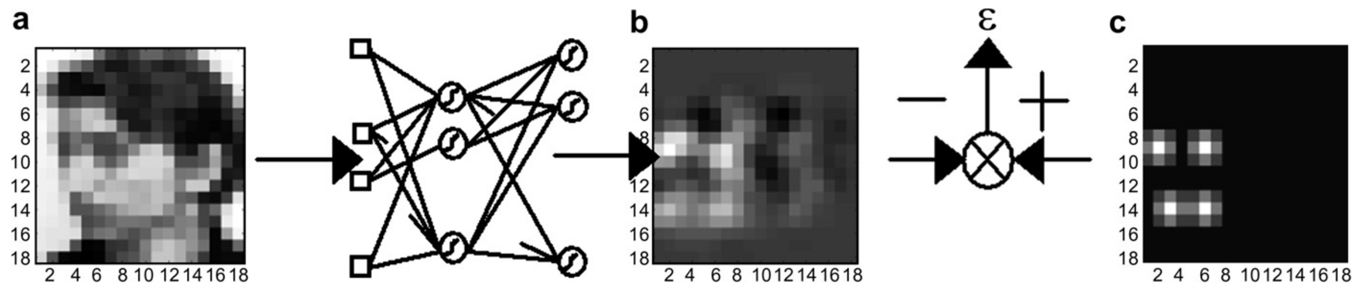

Fig. 3. Training process: input image (a) feeds the network. The mean squared error ε between network output (b) and feature map (c) is used as cost function.

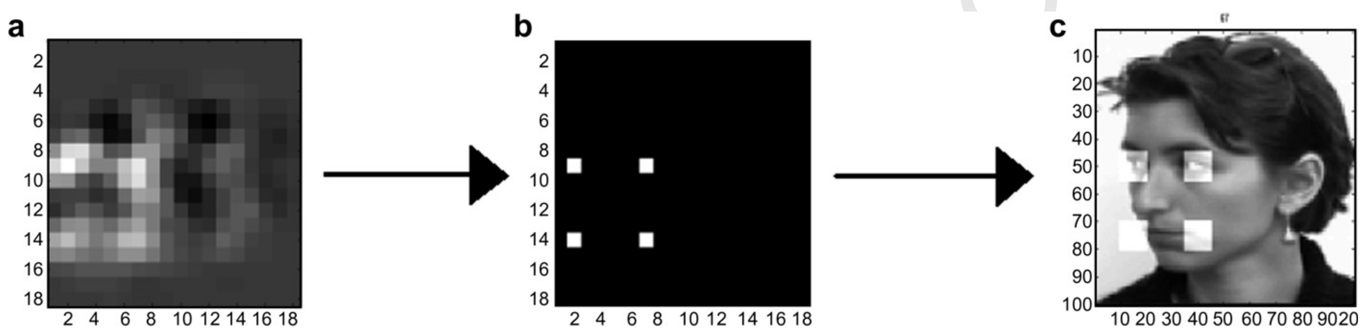

Fig. 4. Decision process: network produces the output image (a) where local maxima are detected (b) and back-projected onto the original image (c).

field is inspired by perceptive psychology (Hubel and Wiesel, 1962). Convolutional neural networks have been applied by LeCun et al. (1998) to handwritten character recognition, by Garcia and Delakis (2004) to face detection and by Duffner and Garcia (2005) to facial feature detection. The proposed network architecture is shown in Fig. 5. The number of neurons in the first hidden layer depends on the choice of the size of the sub-window (X and Y) and the overlapping (defined as dX and dY) between two adjacent sub-windows. The purpose of the second layer is to compress the features information extracted by the first layer (feature extraction layer). The third layer extracts higher order features and transforms the compressed extracted features to desired output map. The second and third (output) layers are fully-connected layers. Non-linear sigmoïdal units are used for hidden and output layers neurons. Hidden layers in neural networks are responsible for constructing higher order fea-

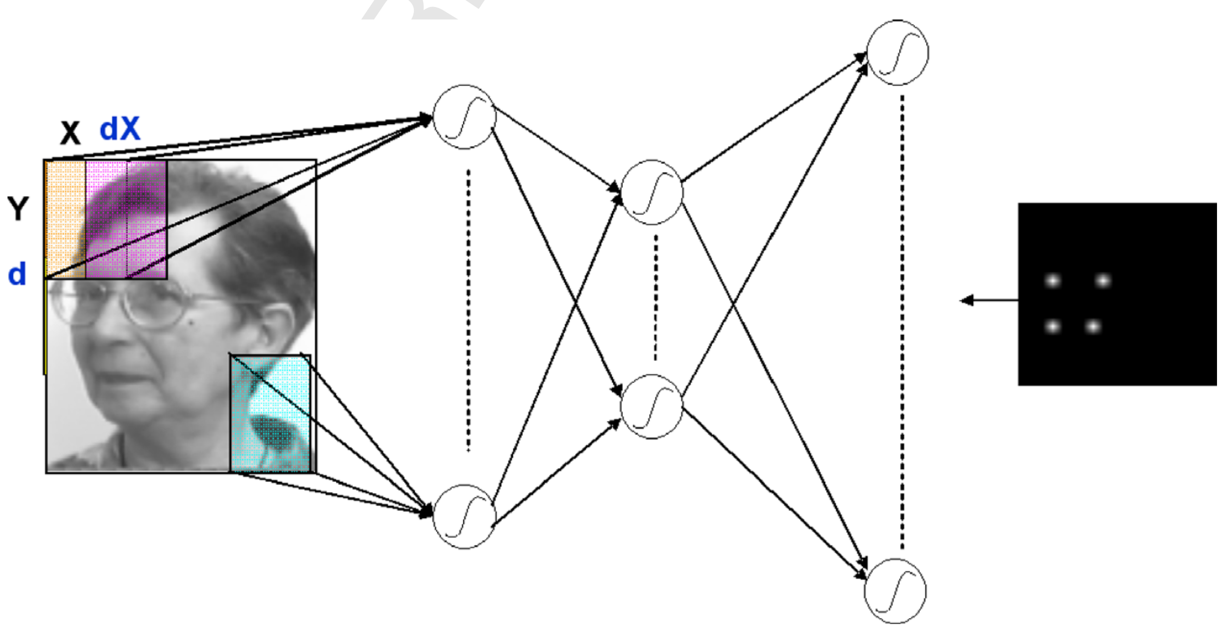

Fig. 5. Space displacement neural network.

Please cite this article in press as: Hanif, S.M. et al., Real-time facial feature localization by combining space ..., Pattern Recognition Lett. (2007), doi:10.1016/j.patrec.2007.09.016

tures, so more hidden layers can be added to increase the network representation capability as done by Garcia et al. [5]. However, increasing the hidden layers will also increase the computational requirement for the training of neural network. The proposed architecture is a compromise between these two constraints, i.e. optimal number of hidden layers and computational complexity.

Important parameters of this network which are considered during training are window size (X and Y), overlapping ($\mathrm{d}X$ and $\mathrm{d}Y$), number of hidden cells (N_{H}), and number of epochs (N). A small overlapping between two adjacent sub-windows allows interpreting the overlapped region by two different feature extractors. Thus, overlapping helps in construction of useful features from input image.

4.3. Multiple localizers and gating network

To improve the localizer accuracy, we decided to use several localizers; each one specialized on a given orientation. The clustering procedure described in Section 2 could separate the initial dataset into several subsets corresponding to a given face pose. Given N the number of considered orientations, the corresponding multiple localizer consists in N networks. So, for an input image, we have now N output images and N localization hypotheses corresponding to the first four intensity sorted local maxima of each output image (Fig. 6).

We employ a Gating Network to combine these hypotheses. The Gating Network is a part of an ensemble network as shown in Fig. 7. Ensemble networks are powerful tools specially when facing complex problems. Network ensembles are made up of a linear combination of several networks that have been trained using the same data, although the actual sample used by each network to learn can be different. Each network within the ensemble has a potentially different weight in the output of the ensemble. Perrone and Cooper (1993) have shown that generally, the network ensemble has a generalization error smaller than that obtained with a single network and also that the variance of the ensemble is smaller than that of a single network. The output of an ensemble Y is

$$Y = \Sigma g(i)y(i)$$

where $y(i)$ is the output of ith network in ensemble when a face-image is presented to it and $g(i)$ is the coefficient or weight associated to the ith network. The sub-sampled version of the same face-image is also presented to gating network.

In general, during the training of ensemble network, the experts learn their own task and gating network generates associated weights. But in our case, experts and gating network are trained individually. Experts have already been attributed the task. Each expert is specialized on a given face orientation. Once experts have been trained, the gating network is trained. The desired output of gating network is the associated weight $g(i)$ of each network. These weights

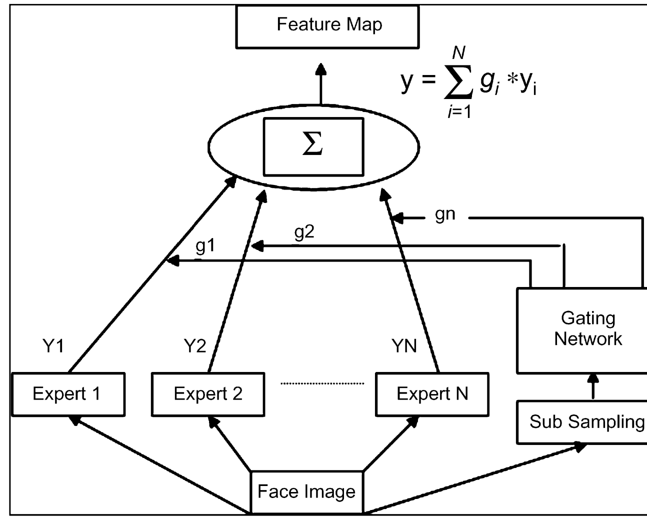

Fig. 7. Gating network.

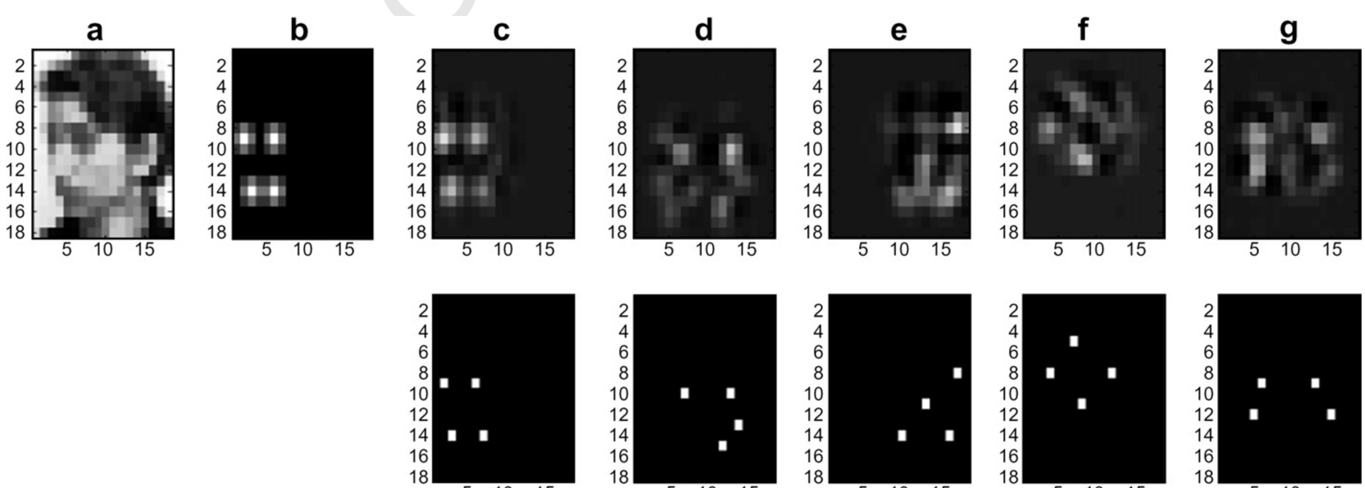

Fig. 6. Multiple localizers: Input image (a), target image (b), output image for the five networks and localization hypothesis (c–g).

are computed with Generalized Ensemble Method (GEM) using the output of each expert. Perrone and Cooper (1993) have shown that the mean square error of GEM estimator is always less than or equal to mean square error of naïve estimator. Moreover, as far as, the error of different experts in ensemble is correlated, GEM provides the best estimate of the target function (Y) in mean square sense. For a detail discussion, please see the authors work.

As stated earlier that generalized ensemble method chooses such weights $g(i)$ that minimize the mean square error with respect to target function. It computes the misfit function, i.e. deviation of expert output from target and then using the symmetric correlation matrix, the desired weights/coefficients $g(i)$ are calculated.

In our case, the purpose of gating network is not to extract features from face-image, a sub-sampled image is used for training. The gating network is a classical, fully-connected two-layered perceptron. The network input is a 20×20 face-image and its outputs are N (five for example) coefficients.

4.4. A cascade system for real-time facial feature localization

The performance of multiple localizer is better than single one but due to increased number of neural networks, its processing speed decreases. In Section 5, we can see that single network can process 110 images/s while multiple localizer can process on 11 images/s. However, in order to perform the localization task in real-time, the system must process at least 30 images/s while keeping the good performance, i.e. low localization error.

To accomplish this task, we propose a cascade system that is able to meet real-time constraints while keeping good performance. In our cascade system, the single and multiple localizers are intelligently combined. Single localizer acts as level 1 detector. If its hypothesis is rejected, multiple localizer is activated to perform the detection of facial features in face image. This acts as level 2 detector. If level 2 detector fails, the face-image is rejected and next face-image is processed.

The rejection or acceptance of a certain localizer's hypothesis (a feature map) is based on a validation step which is able to differentiate between a poor localization (false detections) or good localization (correct detections). As human face has a particular geometry so knowing the location of some facial features, any false detection can easily be detected and rejected. This validation step is based on the computation of six mutual distances between the four facial features (eyes centers and mouth corners). In this method, each distance is modeled by a univariate Gaussian distribution. The parameters (μ, σ) of these distributions are estimated on the reference database. A set of features is considered as valid if the six mutual distances lie within ($\mu \pm 2\sigma$) of the estimated distribution.

At level 3 of cascade a "local feature analyzer and corrector" is employed at the end to improve accuracy and to do detail analysis of facial features. It consists of four small neural networks which try to give accurate position of a certain detected feature and also analyze the detected region in more detail, e.g. eye region analyzer and corrector can make a detail map of eye and eyebrows and also accurately locate eye center as desired. In our experimentation, four such level 3 detectors are employed to correct the detection made by precedent detectors.

5. Experimental results

5.1. Performance measure

The performance of a certain localizer is evaluated in terms of normalized localization error (le). The normalized localization error is defined as the mean Euclidean distance between the detected feature position and the true feature position normalized with respect to the inter-ocular distance D_{eyes} (Euclidean distance between left and right eyes).

$$\text{le}(j) = \frac{1}{4^* D_{\text{eyes}}} \sum_{i=1}^{4} \sqrt{(x_{oi} - x_{di})^2 + (y_{oi} - y_{di})^2}$$

where (x_{oi}, y_{oi}) is the ith feature location in desired feature map and (x_{di}, y_{di}) is the ith feature location detected in network output.

The mean normalized error is computed on all the test images. All the localization speeds are given on Intel Pentium Centrino 1.6 GHz using Matlab.

5.2. LISIF database

To evaluate the localizer accuracy, we applied the leave-many-out method for all the following experiments (except the identity test). We divided the whole dataset into two sets: training set (three-fourth) and test (one-fourth). We dispatched peoples in both training and test sets with slightly different orientations. For the identity test, we applied the leave-one-out strategy: we tested networks on images of one individual and used all the others to train.

5.2.1. Hybrid auto-associative Network

We studied thoroughly this fully-connected architecture in (Prevost et al., 2006). We just present here the main conclusions.

Single localizer. First, we trained a single neural network to localize facial feature on the whole database and perform orientation-free localization. In the first experiment, we tested the localizer sensitivity to feature number and position. We trained several localizers. The first one consisted of four single feature localizers; each one specialized on one facial feature. It results in four localization errors for the left eye (LE), the right eye (RE), the left mouth corner (LC) and the right one (RC). The second localizer used two double feature localizers and each localizer deals with a couple of features: left and right eyes (LRE) and mouth corners (LRC). The last one was a quadruple feature localizer (LREC). Table 1 summarizes results in term of mean

Please cite this article in press as: Hanif, S.M. et al., Real-time facial feature localization by combining space ..., Pattern Recognition Lett. (2007), doi:10.1016/j.patrec.2007.09.016

Table 1
Mean normalized error of the single, double and quadruple feature localizers on the test set

Localizer	Mean normalized error
LE	0.12
RE	0.12
LC	0.15
RC	0.15
LRE	0.11
LRC	0.16
LREC	0.14

normalized error for the best network we found after optimization: 20×20 input and output cells corresponding to the total number of pixels in image and feature map, 60 hidden cells and 10,000 training epochs.

These results are very interesting. The mean normalized error is lower for the eyes than for the mouth corners as these latter are more sensitive to facial expression. Secondly, the mean error does not change when the number of feature to localize increases. Owing to these conclusions, we decide to use the quadruple localizer for further experiments. We also tested its sensitivity to person identity. The results are shown in Table 2. Some useful statistics: mean, median, standard deviation, etc. have been calculated from leave-one-out test. It has been observed that mean localization error approximately doubles when a person is not present during training. Person identity problem depends on various factors which include race, personal features (a kid has different features than an aged person), skin color, presence of "accessories" like beard or glasses, etc. However, using a huge database containing various age groups having different colors and race etc will solve this problem.

Multiple localizers. To improve the localizer accuracy, we decided to use several localizers; each one specialized on a given orientation. The clustering procedure described in Section 2 could separate the initial dataset into several subsets corresponding to a given face pose. Given N the number of considered orientations, the corresponding multiple localizers consist in N networks. So, for an input image, we have now N output images and N localization hypotheses corresponding to the first 4 intensity sorted local maxima of each output image. To compare the accuracy of the multiple localizers, we compute the normalized error for each hypothesis and apply the WTA (Winner Takes All) criterion to select the best one. We have considered up to $N = 6$ orientations.

Table 2
Sensitivity to person identity

Statistics	Mean normalized error (network trained with all persons present)	Mean normalized error (network trained using leave-one-out method for identity test)
Maximum	0.53	0.82
Minimum	0.05	0.10
Mean	0.16	0.28
Standard deviation	0.10	0.16
Median	0.12	0.24

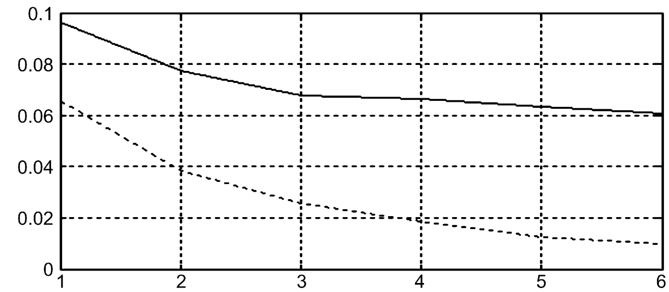

Fig. 8. Mean normalized error on the training set (dotted) and the test set (solid) versus number of orientations considered.

As can be seen (Fig. 8) the mean normalized error decreases continuously on both training and test sets when N increases. Such results are quite logical: as the number of specialized networks increases, the range of face orientations each network has to deal with decreases. The association process between face image and feature map becomes easier and the normalized error decreases.

5.2.2. Space displacement neural network

Single localizer. Eight different realizations of this architecture, based on choice of values of window size, overlapping and number of neurons in second hidden layer, are trained and evaluated on the test dataset (720 images). A total of 10,000 iterations are used to train each network realization. The best score (minimum localization error in mean sense) is obtained from network realization R4 which contains 144 neurons (window size is 6×6 and overlapping is 2×2) in first hidden layer and 50 neurons in second hidden layer. The mean error obtained on reference database is 0.86 and on test database is 0.14. Comparison of first four realizations (R1–R4) is shown in Fig. 9. The window

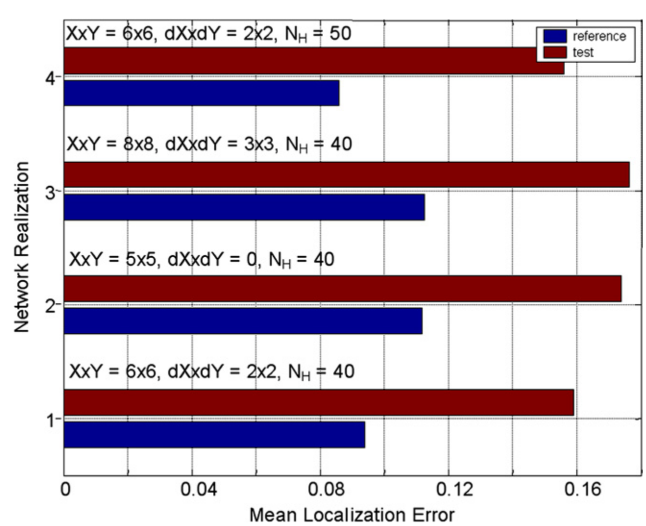

Fig. 9. Mean localization error for different network realizations.

Please cite this article in press as: Hanif, S.M. et al., Real-time facial feature localization by combining space ..., Pattern Recognition Lett. (2007), doi:10.1016/j.patrec.2007.09.016

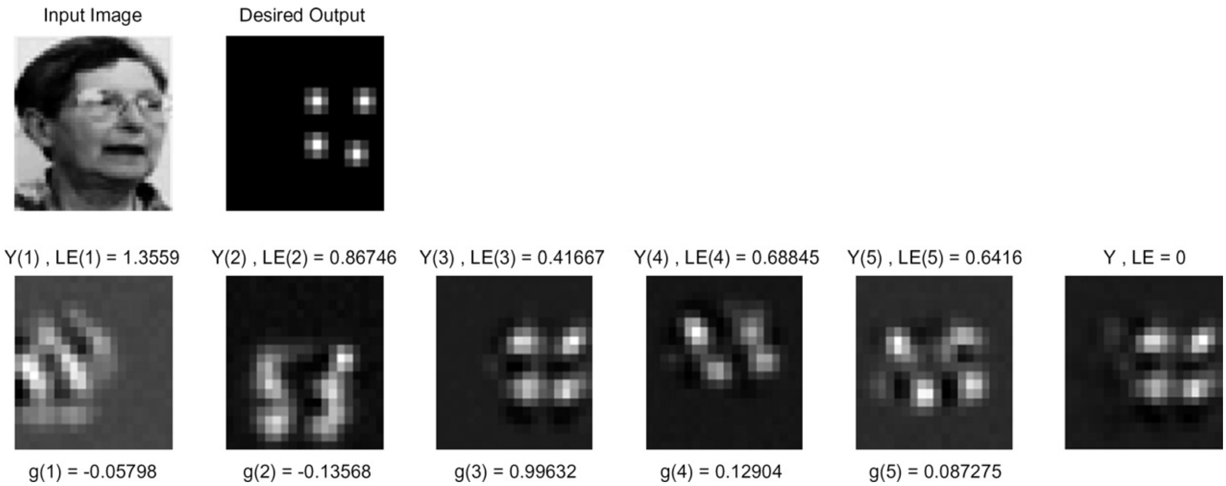

Fig. 10. *Multiple localizer*: $Y(i)$, $g(i)$, LE(i) for $i = 1, \ldots, 5$ are expert's outputs, the associated weights and localization error, respectively. Y is the output of the ensemble network and LE is the associated localization error. We can see that the localization error of ensemble network is less than that of experts.

sizes, overlapping and number of neurons in hidden layers are different for each realization.

Network realizations R1 and R4 have the same first layer but R4 contains more neurons in second hidden layer than R1 and generalize a little better. We will use R4 in the following experiments and summarize now its performance. The mean error is 0.08 on the training set and 0.15 on the test set, 60% examples have a localization error lower than 0.1 and the system can perform 110 images/s.

Multiple localizers. We trained five single-orientation specialized SDNN on each image subset. We also trained a gating network with 80 hidden cells. Fig. 10 shows the combination process of multiple localizer on one example. Each expert produces its hypothesis $Y(i)$ and gating network generates associated weights $g(i)$, we can see that the localization error of each expert is greater than that of the final output (Y). Note that the expert # 3 has been weighted heavily because of its closeness to desired output.

Fig. 11 shows localization error distribution. The mean error is 0.05 on the training set and 0.12 on the test set, 65% of the examples have approximately 0.1 localization error. The overall multi-network system with gating network can perform 11 images/s.

5.2.3. Sensitivity to noise and occlusion

Finally, we wanted to evaluate the multiple SDNN localizer robustness against noise and occlusions.

Noise test. First, we synthesized images by adding white gaussian noise on all the images in test database. The signal to noise ratio varied from 0 dB to 20 dB. As can be seen (Fig. 12), the system is quite insensitive to gaussian noise due to its convolutional filters.

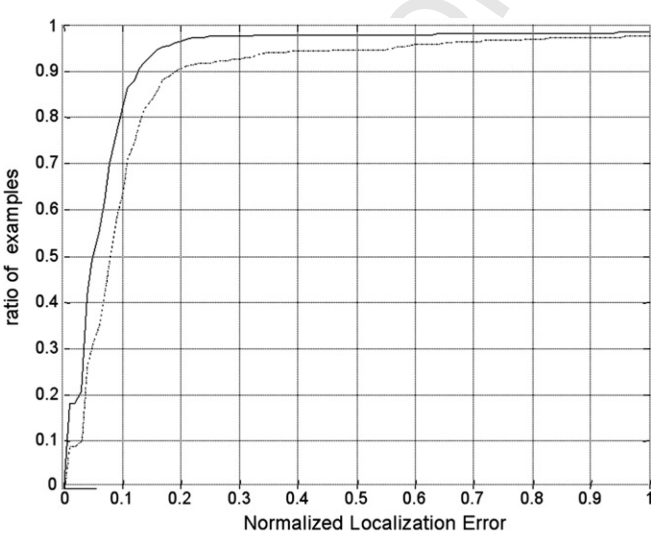

Fig. 11. Mean normalized error on the training set (solid) and the test set (dotted).

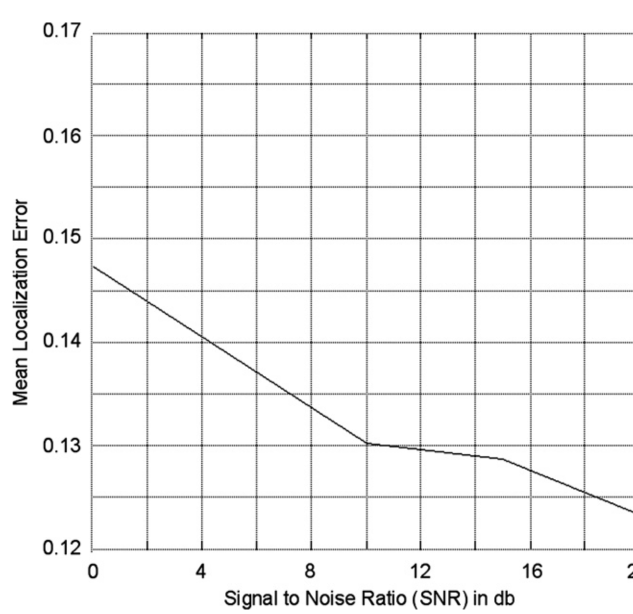

Fig. 12. SDNN multiple localizer: Noise Test.

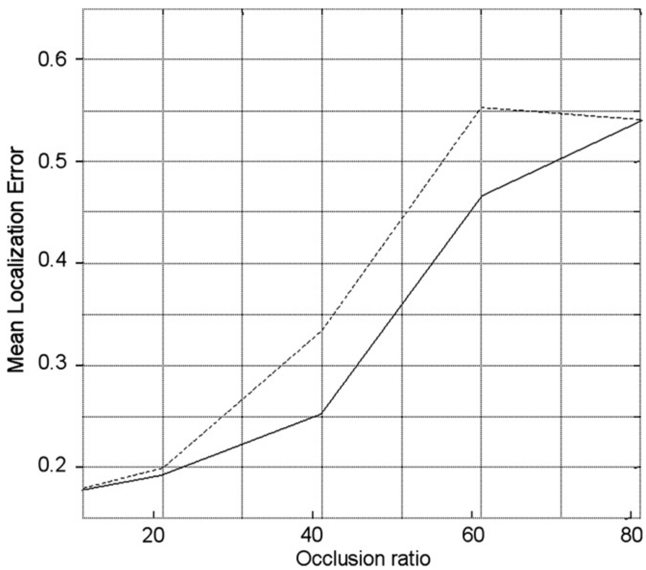

Fig. 13. SDNN multiple localizer: Occlusion test: localization error versus occlusion ratio, for bottom to top (solid) and for top to bottom (dotted).

Occlusion test. In real life, face occlusions are quite common. So we need to test the localizer on occluded images. For this test, synthetic images are formed by masking 10–80% of the face region. Images are occluded in two different manners: from Bottom to Top (mouth is occluded first) and from Top to Bottom (Eyes are occluded first). The localization error is directly proportional to occlusion percentage (Fig. 13). There is a small change in localization error when images are 10–20% occluded and it increases rapidly as occlusion percentage reaches 40%. An occlusion of 40% hides the mouth/eyes region in the face and thus hinders the network to extract the corresponding features. Neural networks are known to generate a mean output when outlier occurs. In this case, occlusion can be considered as an outlier thus, the network outputs the mean. Comparing the two occlusion tests discussed above, we can note that localization error is more sensitive to occlusion from top than from bottom. An occlusion of 40% results in a localization error of 33% when occluded from top while localization error is 28% when occluded from bottom. Thus in general we can say that detection of eyes is more stable than that of mouth as mouth undergo different expression and occlusion makes it nearly impossible to correctly detect the mouth corners.

5.3. Cascade system

Finally, the cascade system of Section 4.4 is implemented using single, multiple localizers, validation step and local feature analyzer and corrector. In first experiment, the simple cascade system i.e. single, multiple localizers and validation step is formed. This combination gives 12.1% mean localization error on the test database. The system rejection rate is 3% i.e. when level 1 and 2 detectors fails to localize the features and validation procedure declares a poor localization. The missed detection rate, i.e. validation procedure declares a poor localization when it is not true, is only 0.6%.

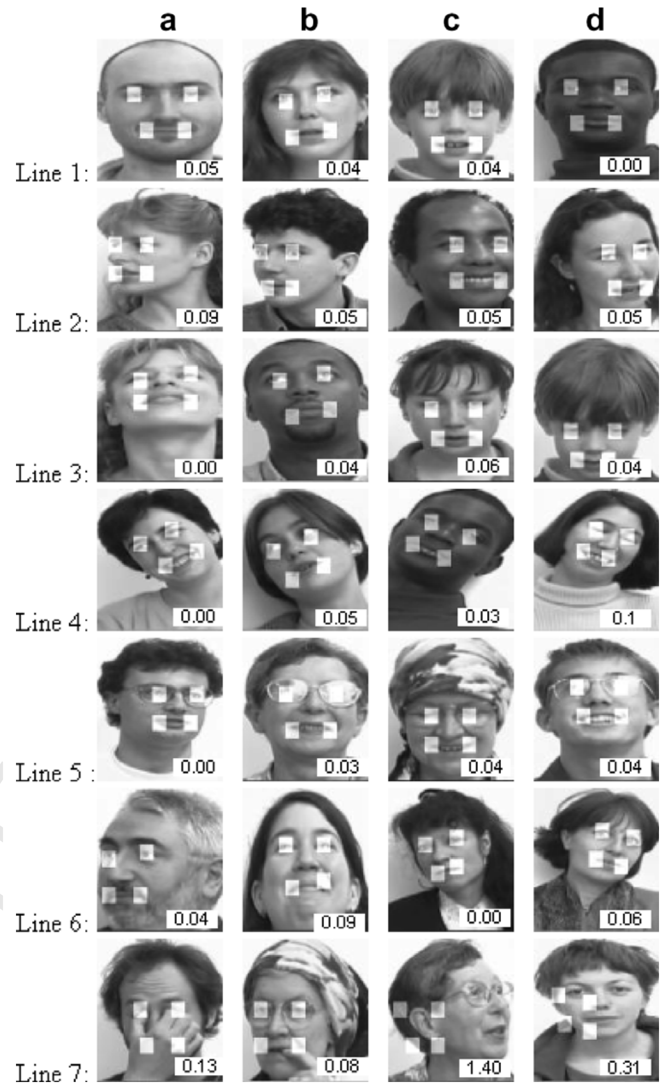

Fig. 14. Localization results on some test images of LISIF database. The normalized error is indicated bellow.

In second experiment, local feature analyzer and corrector is employed as level 3 detector along with single, multiple localizers, validation step. The input to local feature analyzer and corrector is an image 9×9 around a certain facial feature and the desired output is an image 9×9 containing the exact feature location as one bright point. This configuration gives a mean localization error of 11.9% on test database and can perform 40 images/s. The contribution made by "local feature analyzer and correctors" has a little effect (a gain of 0.2%) on localization error. However, the "cascade system with local feature analyzer and corrector" performs better than single localizer that gives 15.6% mean localization error.

The information combination outperforms the single localizer. We can summarize the cascade results on the test set as follows: 65% examples have localization error lower

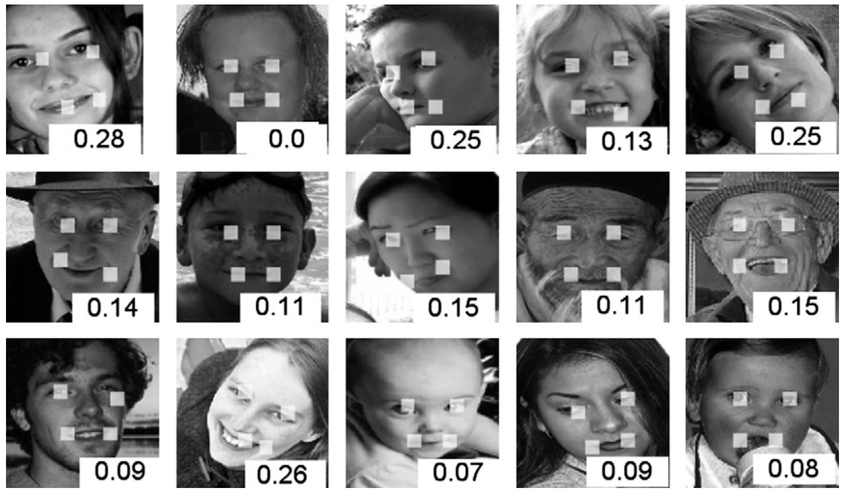

Fig. 15. Localization results on some test images of ECU database. The normalized error is indicated bellow.

than 0.1. For single network, only 60% examples had a localization error lower than 0.1%. Finally, we present some localization results on test images (Fig. 14): frontal faces (1st line), left-sided and right-sided faces (2nd line), upward and downward faces (3rd line), and tilted faces (4th line). Examples of localizer sensitivities to glasses (5th line), scale (6th line) and partial occlusions (7th line) are shown. The association procedure makes the system less sensitive to partial occlusions and noise: e.g. if one feature is not visible, its position is inferred by the positions of other visible features. Two localization errors are presented (7th line). Note that, in both cases, an accurate localization hypothesis was found but the combination method failed to select it.

5.4. ECU database

Opposite to LISIF database, this database contains general scenarios for facial feature localization. In particular, the number of persons (except some "starts" personalities) in this database is nearly equal to the number of images. As stated earlier, we extracted the face-images using ground truth (rectangle around face) and using mirror images, we constructed a face database of more than 7000 images with different resolutions. As this database does not contain various orientations, we decided to use simple localizer (Hybrid auto-associative network and SDNN), i.e. one network for all orientations. We divided the database into two equal halves, each containing more than 3500 face-images. One half serves as a training database and other half is used for evaluation (test database). The best network, obtained after rigorous experimentation, for hybrid auto-associative network has (900 inputs, 60 hidden cells and 400 outputs) while for SDNN has ($X = 6$, $Y = 6$, $\mathrm{d}X = 2$, $\mathrm{d}Y = 2$, $N_{\mathrm{H}} = 50$, 400 outputs). Both networks were trained for 10,000 epochs. The mean localization error obtained with hybrid auto-associative network is 0.10 on training database and 0.15 on test database, while we obtained 0.11 on training database and 0.14 on test database using SDNN. With these results, we can see that generalization of SDNN is better than auto-associative network when number of examples in training database increases. Moreover, the problem of sensitivity to persons identity has vanished. Some localization results are shown in Fig. 15.

6. Conclusions

We have presented a novel algorithm for the detection of facial features in a pre-focused face image. It is based on a particular neural network trained to associate a feature map with a face image. We studied thoroughly the single, orientation-free localizer and show that its accuracy increases with the number of features to detect. We proposed an alternate method where several specialized networks were trained to deal with specific face pose. The best localization hypothesis is then formed by combining all the network outputs with a gating network. This multiple localizer is more accurate than the orientation-free localizer: the mean normalized error decreases from 15.6% to 11.9% and the system performs more than 40 images/s.

We have shown that with a large training database, the system is less susceptible to person identity and its generalization increases. Currently, we are working on actual system with three step: face detection localization (Belaroussi et al., 2006), facial feature localization, detail facial feature analysis (local localizers).

7. Uncited references

Bishop (1995), Fumera and Roli (2005). Q3

References

Belaroussi, R., Prevost, L., Milgram, M., 2006. Algorithm fusion for face localization. J. Adv. Inform. Fusion 1 (1), 50–64.

Bishop, C.M., 1995. Neural Networks for Pattern Recognition. Oxford University Press.

Please cite this article in press as: Hanif, S.M. et al., Real-time facial feature localization by combining space ..., Pattern Recognition Lett. (2007), doi:10.1016/j.patrec.2007.09.016

Chellappa, R., Wilson, C.L., Sirohey, S., 1995. Human and machine recognition of faces: A survey. Proc. IEEE 83 (5), 705–740.

Cootes, T.F., Edwards, G.J., Taylor, J.C., 1998. Active appearance models. Eur. Conf. Computer Vision, 484–498.

Cristinacce, D., Cootes, T., 2003. Facial feature detection using AdaBoost and shape constraints. Brit. Mach. Vision Conf., 231–240.

DeMers, D., Cottrell, G., 1993. Non-linear dimensionality reduction. Neural Inform. Process. Systems 5, 580–587.

Duffner, S., Garcia, C., 2005. A Connexionist approach for robust and precise facial feature detection in complex scenes. IEEE Internat. Symp. Image Signal Process. Anal., 316–321.

Feng, G.C., Yuen, P.C., 2001. Multi-cues eye detection on gray intensity image. Pattern Recognit. 34 (5), 1033–1046.

Féraud, R., Bernier, O., Viallet, J., Collobert, M., 2002. A fast and accurate face detector based on neural networks. IEEE Trans. Pattern Anal. Machine Intell. 23 (1), 42–53.

Fumera, G., Roli, F., 2005. A theoretical and experimental analysis of linear combiners for multiple classifier systems. IEEE Trans. Pattern Anal. Machine Intell. 27 (6), 942–956.

Garcia, C., Delakis, M., 2004. Convolutional face finder: A neural architecture for fast and robust face detection. IEEE Trans. Pattern Anal. Machine Intell. 26 (11), 1408–1423.

Hsieh, W.W., 2001. Nonlinear principal component analysis by neural networks. Tellus 53A, 599–615.

Hubel, D., Wiesel, T., 1962. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. J. Psychol. 160, 106–154.

Kramer, M.A., 1991. Non-linear principal component analysis using auto-associative neural networks. Amer. Inst. Chem. Eng. J. 37 (2), 233–243.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient based learning applied to document recognition. Proc. IEEE 86 (11), 2278–2324.

Moghaddam, B., Pentland, A., 1997. Probabilistic visual learning for object representation. IEEE Trans. Pattern Anal. Machine Intell. 19 (7), 696–710.

Peng, K., Liming Chen, S., Ruan Kukharev, G., 2005. A robust and efficient algorithm for eye detection on gray intensity face. In: Proc. 3rd Internat. Conf. on Adv. in Pattern Recognition, pp. 302–308.

Perrone, M.P., Cooper, L.N., 1993. When network disagrees: Ensemble methods for hybrid neural network. In: Neural Networks for Speech and Image Processing. Chapman & Hall, pp. 126–142.

Prevost, L., Belaroussi, R., Milgram, M., 2006. Multiple neural networks for facial feature localization in orientation-free face images. In: IEEE-IAPR Workshop on Artificial Neural Networks, pp. 188–197.

Schwenk, H., Milgram, M., 1995. Transformation invariant auto-association with application to handwritten character recognition. Neural Inform. Process. Systems 7, 991–998.

Toennies, K.D., Behrens, F., Aurnhammer, M., 2002. Feasibility of hough-transform-based iris localisation for real-time-application. Internat. Conf. Pattern Recognition, 1053–1056.

Viola, P., Jones, M., 2001. Rapid object detection using a boosted cascade of simple features. Internat. Conf. Computer Vision Pattern Recognition (1), 511–518.

Xiao, Y., Yan, H., 2004. Face Boundary Extraction. Digital Images Computing: Theory and Application, pp. 947–956. Q1

Xie, X., Sudhakar, R., Zhuna, H., 1994. On improving eye feature extraction using deformable templates. Pattern Recognition 27, 791–799.

Yuille, A., Hallinan, P., Cohen, D., 1992. Feature extraction from faces using deformable templates. Internat. J. Computer Vision 8 (2), 99–111.

Zhu, Z., Ji, Q., 2005. Robust real-time eye detection and tracking under variable lighting conditions and various face orientations. Computer Vision Image Understanding 98 (1), 124–154.