Neural Network Based Complex Visual Information Processing: Face Detection and Recognition

Vaclav Zacek, Eva Volna and Jaroslav Zacek

University of Ostrava, 30 Dubna 22, 701 03 Ostrava, Czech Republic

Abstract. This paper focuses on the issue of detecting and recognizing faces. The work is divided into three main categories. The first part is about detection of faces in constrained conditions. The second part focuses on creation of a different recognition approach. The third one is about the test with robotic devices. However mobile devices (such as robots, small CCD cameras or cheaper cell phones) have many limitations i.e. images quality or very limited computing performance. With respect to limitations the system manages two substantial parts. The first one is responsible for detecting a face in an image. The second one is responsible for calculating the information featured in a face image and recognition of that information. The system is able to process faces in real-time with minimal computation performance and to use minimal space for storing its data. The proposed system was tested on a face database. We have used a FDDB benchmark for an exact comparison.

1 Face Detection and Face Recognition Methods

In the early development of face detection [3], *geometric facial features* such as eyes, nose, mouth, and chins were explicitly used. Properties of the features and relations among them (e.g. areas, distances, angles) were used as descriptors for face recognition. *Statistical learning methods* are the mainstream approaches that have been used in building face recognition systems. Effective features are learned from training data and involve prior knowledge about faces. The appearance-based approaches, such as Principal Component Analysis (PCA) [11] or Linear Discriminant Analysis (LDA) [2], have significantly advanced the face recognition process. These approaches operate directly on an image-based representation (i.e. fields of pixel intensities). It extracts features in a subspace derived from training images. The most successful approach, so far, to handle the non-convex face segmentation works with *local appearance-based features*. These features are extracted using appropriate image filters. An advantage lies in a distribution of face images through local feature space, which is less affected by changes in facial appearance. Early works in this area include local features analysis (LFA) [10] and Gabor wavelet-based features [14]. Current methods are based on local binary pattern (LBP) [1]. There are many variants to the basic approaches like: Ordinal Features [8], Scale-Invariant Feature Transform (SIFT) [9], and Histogram of Oriented Gradients (HOG) [4]. While these features are general purpose, face specific local filters are learned from images [13].

Zacek V., Volná E. and Zacek J..
Neural Network Based Complex Visual Information Processing: Face Detection and Recognition.
DOI: 10.5220/0005126800530060
In *Proceedings of the International Workshop on Artificial Neural Networks and Intelligent Information Processing* (ANNIIP-2014), pages 53-60
ISBN: 978-989-758-041-3

The most famous early example of a face recognition system is due to Kohonen [7], who demonstrated the strength of a simple neural net that was able to perform face recognition for aligned and normalized face images. A face itself can be represented by a lot smaller number of eigenvectors. Nowadays *eigenfaces* are based on the work of Sirovich and Kirby [6] and use principal of component analysis. They start by creation of a feature space from a training set of all faces. Using the created feature space, the algorithm calculates additional information in the form of weights. Each face will have its weights, which are projected to the feature space for reconstruction of the face. The *eigenfaces* method expects that under the best-idealized conditions the variations between faces lie in a linear subspace. This means that classes are linearly separable. The reason behind the "upgrade" of *eigenfaces* approach is that it does not use class specific linear methods for dimension reduction. An example of a class specific method is Fisher's Linear Discriminant (FLD) (*fisherfaces*) [1]. Another approach, Local Binary Patterns (LBP), uses the operator for description of the area surrounding the pixel. If the simplest algorithm is implemented it considerate the surrounding of 3x3 only. When the LBP code for an image is calculated, the edges are ignored for the lack of information. To gain more information, overlapping of the operators is used to obtain high and low frequency information about the neighbors. This means that the required space for storing the basic information about neighboring pixels would be too demanding. We have to use the data reduction to minimize the space requirements. Therefore we introduce uniform patterns.

2 Proposed Face Recognition System

All systems described above have very good results by applying them to the image captured in standard conditions. However mobile devices (such as robots, small CCD cameras or cheaper cell phones) have many limitations i.e. images quality or very limited computing performance. The proposed system should be able to detect and recognize faces in the image despite the limitations. At first, we have to define a limitation conditions for the group of mobile devices [16]:

- The image quality of the input device,
- Real-time detection,
- Processing speed requirements,
- Memory requirements.

With respect to limitations defined above the system has to state two substantial parts.The first part is responsible for detecting a face in an image. The second part is responsible for calculating the information featured in the fac┼ *e image and recognition of that information. The system itself is proposed to be light-weighted. This means that the system should be able to process faces in real-time with minimal computation performance and to use minimal space for storing its data.

2.1 Face Detection

For detection of face, the Haar cascades were chosen. The speed of processing is the

most important part of the proposed system and the Haar cascades are adequate fast because of the ability to make a parallel processing. The speed of the process has been verified in many publications before, i.e. [17]. The Haar Cascades works just like a convolutional kernel mask. To explain the Haar Cascades method, we have to introduce integral images. It simplifies calculation of the sum of pixels. All possible sizes and locations of each kernel are used to calculate plenty of features. Each feature is a single value obtained by subtracting the sum of pixels under a white rectangle from the sum of pixels under a black rectangle. These integral images can be computed in a very fast way (1)

$$ii(x,y) = \sum_{x' \le x, y' \le y} i(x', y') \tag{1}$$

where *ii(x,y)* is the integral image and *i(x',y')* is the original image [13]. Using the following pair of recurrent equations (2):

$$\begin{aligned} s(x,y) &= s(x,y-1) + i(x,y) \\ ii(x,y) &= ii(x-1,y) + s(x,y) \end{aligned} \tag{2}$$

where *s(x,y)* is the cumulative row sum, *s(x,-1) = 0*, and *ii(-1,y) =0*. Because of the form of equations, the integral image can be computed in one pass over the original image. This means that the integral image is a subarea of the original image. We use these areas to calculate the differences inside the integral images with respect to the learnt cascades before. The best threshold is found for each feature, which will classify the faces to positive and negative. But obviously, there will be errors or misclassifications. We select the features with the minimum error rate, which means the features that make better classification of the face and non-face images. The final classifier is a weighted sum of weak classifiers. It is called weak because they cannot classify the image alone, but together they form a strong classifier almost comparable to strong classifiers. In terms of weak classifiers, it is impossible to describe the window with only one classifier. The process of choosing the classifiers could be described as follows (3):

$$h_j(x) = \begin{cases} 1 & \text{if } p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where weak classifier $h_j(x)$ consists of feature f_j, threshold θ_j and parity p_j that indicates the direction of inequality sign. This process takes a large amount of time, because there are a large number of features to try. After the learning process is finished, these functions can be effectively used to obtain the desired shapes from image in the fastest way. This is done by the parallelization of the search task.

The principle of the proposed face detection using Haar Cascades will consists of two phases. The goal of the *first* phase is conversion of the image to the grayscale. Detection does not depend on the color in the image. This means that detection is possible even with black and white pictures. In the *second* phase, the system tries to localize all faces in the image with the Haar Cascades. The used cascades have been taught to recognize faces from another set of faces before. These areas with detected faces will be later on used for feature extraction [15, 16].

2.2 Face Recognition

When the system successfully detects a face in the image, it should be able to work with it. The parameters for the face recognition must be chosen carefully to ensure that important information about the face is preserved. The amount of the information should not limit us in real-time processing. When the recognition algorithm is learned properly, it should be able to recognize the face across different input conditions (e.g. slightly rotated head, different illumination conditions, and even other face accessories). We needed to choose parameters, which are fast enough to process but not redundant (they need to be orthogonal). This condition means that these parameters could not be obtained from another parameter by linear combination. In our solution we assume at least three parameters in the geometrical representation in space, which are orthogonal. The number of the used parameters for recognition is not limited to only three. This three-tuple vector will be used as an input vector for the neural network with dimension of three [16].

Fig. 1. Face layout features (distances) that are invariance thanks to their size.

We can also state an invariant parameter - face color (because of the Haar Cascades preprocessing). Then the initially infinite possibilities of face are limited to thirty-six categories (Von Luschan's chromatic scale). First, we localize specific areas (as mouth and eyes) on face and create color medians. The reasons for excluding these areas are that the mouth and eyes color are different from the skin color. Thanks to the exclusion, the obtained median is more accurate to the actual skin color of the person. The median for the face is used as one of the inputs for the neural network. Other parameters, which have been considered for its invariance thanks to size, are the face layout features. Figure 1 shows the distances considered for the size features that are the following.

- A / B gives the face specific width to height comparison. These numbers will differ between specific subtypes, which could be found when we divide these shapes.
- C / (E + F) means the ratio between the distance of centers of both eyes and the sum of distance from left border of the detected area to the left eye center (E) and the distance from right border of the detected area to the center of the right eye (F).
- C / 2 means a ratio between the distance of the line which connects the eyes and center of the mouth area and the height of the detected area for face.
- D / B. The center of the line, which connects the eyes centers, is chosen as a start point of the distance D. This feature is invariant of the size when it is divided by height of the detected area.

With the knowledge about the basic neurons possibilities, the multilayer neural network has been chosen for the experimental part [16]. Having taken the number of features into consideration, the basic input dimension for the recognition was determined to be of size six. Each one of the input neurons represents specific information about the face as follows:

- x1 – face color median
- x2 – eye one color median
- x3 – eye two color median
- x4 – mouth color median
- x5 – comparison of width of the face to its weight
- x6 – relative distance of the eyes

The number of hidden units has been set to sixteen. This number may look quite large but it has shown the best result in the testing and thanks to this number the network is able to remember a larger number of faces. The activation function for the hidden layer is the sigmoid function. The output dimension is going to bet set to one output neuron with the linear activation function. Additional information, which needs to be processed, is the output of the network. Even if the output from the network is linear, we cannot be sure that we are getting the exact face. That is why the output will be further processed with each of the known faces to be sure if the difference between the output of the network and the face is in the acceptable margin.

2.3 Technical Implementation of the Proposed Approach

Because this solution is aimed to use in real-time environment and with the bad camera conditions the test on perform some tests and proof of concept on the robots. These tests were realized on two types of robots. The first one was NAO robot from Aldebaran robotics. The second one was Wifibot M from Networked Robotics. NAO (in the latest version) has a capturing device with maximal resolution of 1280x960 pixels with 30 fps. This camera provides images with a relatively good quality.

When we did a real-time processing with the NAO we were able to process every tenth frame from the video stream. This slow processing was caused by the size of the image. When the image was downscaled to a smaller resolution it was faster to obtain required face with features. If the features would not be extracted from the face region, it would mean that every fifth frame could be extracted from the stream. This information is interesting for the face tracking in the image. When the face recognition is not the main focus, it could be used to obtain accurate face location in the image.

Wifibot M from Networked robotics has been equipped with camera, which has got maximal resolution 768x576 with 25 fps. Quality of pictures from this camera has not been very good. The whole image was a lot darker than expected.

Behavior of the solution with the robots is the same as the one expected from the first part of the tests. This means that if we want the solution to perform well we need to know the quality of the captured images and do the preprocessing accordingly to the quality. In case of too dark images and a lot of small noise it is need to add additional lighting to the scene or apply specific image filters. This change will make the detection possible even with the bad quality cameras.

3 Experimental Results

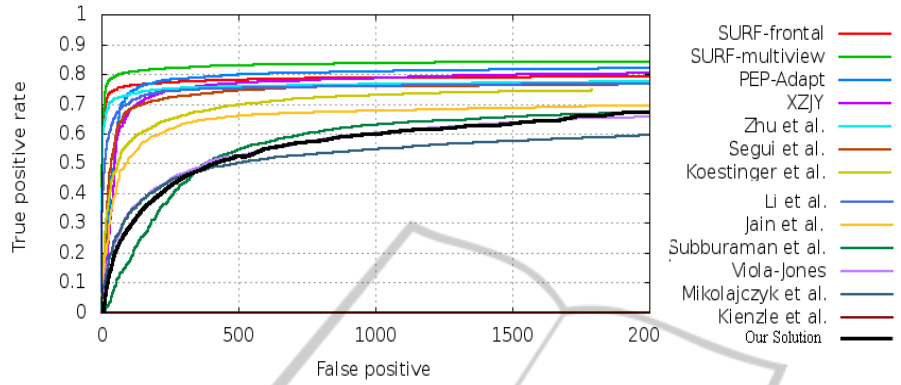

Fig. 2. A comparison for the detection part of the proposed system (the black line) with different algorithms on Face Detection Data Set and Benchmark (FDDB).

The proposed system was tested on a face database. Most such systems required faces to be normalized, which is not the case of the system that was proposed in [16]. This means that the two parts of the system will be tested if both of them can handle faces rotated in a certain degree. At first, the detection parameters needed to be set to detect faces and to detect the features in face. Comparison for the detection part will be realized with help of the Face Detection Data Set and Benchmark (FDDB) [12]. To compare the second part of the system, the study of recognition algorithms has been used [5]. Figure 2 shows results of the comparison for the detection part (our solution represents the black line). When the results should be interpreted it should be taken that the results meant that the face itself was recognized but two or more detection windows have covered it. Figure 3 shows results of the comparison for the recognition part (our solution represents the orange line).

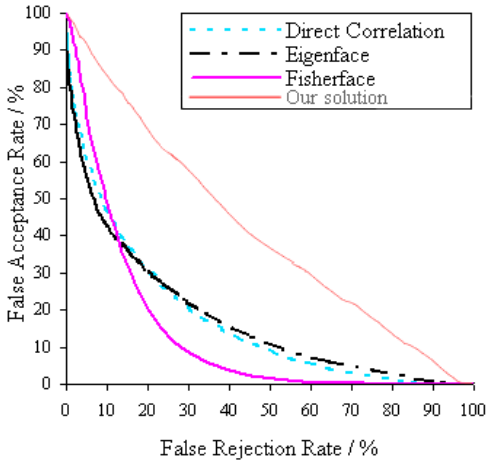

Fig. 3. A comparison for the recognition part of the proposed system (the orange line) with different algorithms on Face Detection Data Set and Benchmark (FDDB).

4 Conclusion

This paper summarizes state of the art face recognition methods and proposes a new method to face recognition on mobile devices. First part of the paper introduces conventional methods to detect and recognition face and also states a main condition for our experiment. Second part introduces a new algorithm to detect a face from poor image quality inputs. The algorithm is based on Haar Cascades and is optimized to work with minimal memory and computation consumption. An integral part of the new approach is also the set of key parameters as an input vector into neural network. Last part of the paper is focused to compare the state of the art algorithms with our proposed solution. We used a FDDB benchmark for an exact comparison. Proposed algorithm can be compared to Viola-Jones and delivers the same positive rate, which is a significant success, because the FDDB benchmark contains only good quality pictures with high resolution. That means our chosen parameter does not affect the positive rate and our solution is able to work also with images in poor quality.

The detection part of the algorithm shows better results with the noised images and the images with problematic light distribution when compared to the original solution with predefined parameters.

The result for our solution of the face recognition shows that the detection itself is still not very robust to be used without additional work, because used parameters are not strong enough to be representative for the face. In the future we would like to insert additional invariant information into the recognition mechanism to make a detection mechanism more precise (like EigenFaces and FisherFaces).

Acknowledgement

This work was supported by the University of Ostrava grant no. SGS16/PrF/2014 and grant no. SGS17/PrF/2014. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

References

1. Ahonen, T., Hadid, A., Pietikainen, M.: Face recognition with local binary patterns. In: Proceedings of the European Conference on Computer Vision, pp. 469–481. Prague, (2004)
2. Belhumeur, P. N., Hespanha, J. P., Kriegman, D. J.: Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. IEEE Trans. Pattern Anal. Mach. Intell. 19(7), 711–720 (1997)
3. Brunelli, R., Poggio, T.: Face recognition: Features versus templates. IEEE Trans. Pattern Anal. Mach. Intell. 15(10), 1042–1052 (1993)
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 886–893 (2005)
5. Heseltine, T., Pears, N., Austin, J a Chen, Z. Face Recognition: A Comparison of Appearance-Based Approaches., no 1. (2003)

6. Kirby, M., Sirovich, L. Application of the Karhunen - Loeve procedure for the characterization of human faces,' IEEE Pattern Analysis and Machine Intelligence, vol. 12, no. 1, pp. 103-108, (1990)
7. Kohonen, T. Self-organization and AssociativeMemory,Springer-Verlag, Berlin, 1989.
8. Liao, S., Lei, Z., Zhu, X., Sun, Z., Li, S.Z., Tan, T.: Face recognition using ordinal features. In: Proceedings of IAPR International Conference on Biometrics, pp. 40–46 (2006)
9. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proceedings of IEEE International Conference on Computer Vision, p. 1150, Los Alamitos, CA (1999)
10. Penev, P., Atick, J.: Local feature analysis: A general statistical theory for object representation. Neural Syst. 7(3), 477–500 (1996)
11. Turk,M.A., Pentland, A.P.: Eigenfaces for recognition. J. Cogn. Neurosci. 3(1), 71–86 (1991)
12. Vidit, Jain and Erik Learned-Miller.FDDB: A Benchmarkfor Face Detection in UnconstrainedSettings.Technical Report UM-CS-2010-009, Dept. ofComputer Science, University of Massachusetts, Amherst. (2010)
13. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, p. 511 (2001)
14. Wiskott, L., Fellous, J., Kruger, N., von der Malsburg, C.: Face recognition by elastic bunch graph matching. IEEE Trans. Pattern Anal. Mach. Intell. 19(7), 775–779 (1997)
15. Žáček, V., Žáček, J., Volná, E.: Face extraction from image with weak cascade classifier. In Proceeding of CSOC14 - 3rd Computer Science On-line Conference. Advances in Intelligent Systems and Computing volume 285, pp 495-505. Springer Int. Publishing (2014)
16 Žáček, V. Soft-Computing Based Complex Visual Information Processing. Diploma Thesis. University of Ostrava,Ostrava (2014)
17 Savchenko A.V., Khokhlova Ya.I.: About Neural-Network Algorithms Application in Viseme Classification Problem with Face Video in Audiovisual Speech Recognition Systems, Optical Memory and Neural Networks, 23(1), 34-42 (2014)