

A Symmetry Based Face Detection Technique

Sriparna Saha, *Student Member, IEEE* and Sanghamitra Bandyopadhyay, *Senior Member, IEEE*
Machine Intelligence Unit, Indian Statistical Institute, Kolkata, India-700108
Email: {sriparna_r, sanghami}@isical.ac.in

Abstract—In this paper a symmetry based face detection technique is proposed. Here symmetry of the human face is used to quickly locate the human faces from an image. At first sobel edge detection operator is applied to get the edge image from the original image. In order to measure total number of symmetrical points present in an image window, a newly proposed point symmetry based distance is used. Kd-tree based search is used to reduce the complexity of computing point symmetry based distance. The window having the maximum number of symmetrical pairs is considered having the candidate face region. The effectiveness of the proposed method is demonstrated in detecting faces from six different images.

Index Terms—Face detection, symmetry property, point symmetry based distance, Kd-tree

I. INTRODUCTION

In recent years the problem of human face recognition has attracted considerable attention [1] [2]. A comprehensive overview of this problem can be found in [3][4]. The applications of face recognition systems widely range from secure access control, financial transactions to many others. The first important step of fully automatic human face recognition is human face detection. Face detection determines the locations and sizes of faces in an input image. Human faces represent one of the most common patterns in our vision. They are easily located in cluttered scenes by infants and adults alike. However automatic human face detection by computers is a very challenging task because face patterns can have significantly variable image appearances. For example, human faces vary from genders, ages, hairstyles and races etc. In addition, the variations of scales, shapes and poses of faces in images also hinder the success of automatic face detection systems.

One of the earliest works in face detection was reported by Sakai et al. [5]. They define the model of a human face in terms of several subtemplates corresponding to face contour, eye, nose and mouth. An edge map extracted from the input image is matched against the subtemplates with possible variations in the position and size. The location of a face is determined from the scores of match. The merit of template matching techniques is that are very simple. However, they prove to be inadequate because wide variations exist in human faces. Some face detection systems adopt image-invariance schemes that assume that there are certain spatial image relationships common and possibly unique to all face patterns [6]. Another approach to face detection is based on deformable templates that use parameterized curves and surfaces to model the nonrigid features of interests e.g., eyes,

mouth of faces [7]. The template then interacts dynamically with the input image, by altering its parameter values to minimize deformation "stress" in the feature. Recently, the use of neural networks or other mechanisms in face detection has been studied by many researchers [8]. Training a neural network or a classifier for the face detection task is challenging because of the difficulty in characterizing prototypical "nonface images". Practically any image containing no face can serve as a nonface example. Since the space of nonface image is much larger than the space of face images how to collect a "representative" set of nonfaces so as to force the network or the classifier to learn the precise boundary between face and nonface image is a very demanding problem.

In this paper an efficient method to locate human faces in a complex background is proposed. Here we have used the symmetry property of human faces to quickly locate the candidate faces. A newly proposed point symmetry based distance [9] is used for this purpose. For reducing the complexity of computing the PS-distance, Kd-tree [10] based nearest neighbor search is also used.

II. A NEW DEFINITION OF THE POINT SYMMETRY DISTANCE

In this section, a new PS distance [9], $d_{ps}(\bar{x}, \bar{c})$, associated with point \bar{x} with respect to a center \bar{c} is described. As shown in [9], $d_{ps}(\bar{x}, \bar{c})$ is able to overcome some serious limitations of an earlier PS distance [11]. Let a point be \bar{x} . The symmetrical (reflected) point of \bar{x} with respect to a particular centre \bar{c} is $2 \times \bar{c} - \bar{x}$. Let us denote this by \bar{x}^* . Let $knear$ unique nearest neighbors of \bar{x}^* be at Euclidean distances of d_i , $i = 1, 2, \dots knear$. Then

$$d_{ps}(\bar{x}, \bar{c}) = d_{sym}(\bar{x}, \bar{c}) \times d_e(\bar{x}, \bar{c}), \quad (1)$$

$$= \frac{\sum_{i=1}^{knear} d_i}{knear} \times d_e(\bar{x}, \bar{c}), \quad (2)$$

where $d_e(\bar{x}, \bar{c})$ is the Euclidean distance between the point \bar{x} and \bar{c} . It can be seen from Equation 2 that $knear$ cannot be chosen equal to 1, since if \bar{x}^* exists in the data set then $d_{ps}(\bar{x}, \bar{c}) = 0$ and hence there will be no impact of the Euclidean distance. On the contrary, large values of $knear$ may not be suitable because it may overestimate the amount of symmetry of a point with respect to a particular cluster center. Here $knear$ is chosen equal to 2.

Note that $d_{ps}(\bar{x}, \bar{c})$, which is a non-metric, is a way of measuring the amount of symmetry between a point and a cluster center, rather than the distance like any Minkowski distance.

The benefit of using several neighbors instead of just one in Equation 2 is as follows.

- 1) Here since the average distance between \bar{x}^* and its $knear$ unique nearest neighbors have been taken, this term will never be equal to 0, and the effect of $d_e(\bar{x}, \bar{c})$, the Euclidean distance, will always be considered. Note that if only the nearest neighbor of \bar{x}^* is considered and this happens to coincide with \bar{x}^* , then this term will be 0, making the distance insensitive to $d_e(\bar{x}, \bar{c})$. This in turn would indicate that if a point is marginally more symmetrical to a far off cluster than to a very close one, it would be assigned to the farthest cluster. This often leads to undesirable results as demonstrated in [9].
- 2) Considering the $knear$ nearest neighbors in the computation of d_{ps} makes the PS-distance more robust and noise resistant. From an intuitive point of view, if this term is less, then the likelihood that \bar{x} is symmetrical with respect to \bar{c} increases. This is not the case when only the first nearest neighbor is considered which could mislead the method in noisy situations.

It is evident that the symmetrical distance computation is very time consuming because it involves the computation of the nearest neighbors. Computation of $d_{ps}(\bar{x}_i, \bar{c})$ is of complexity $O(N)$. Hence for N points the complexity of computing total symmetrical distance with respect to a particular cluster center is $O(N^2)$. In order to reduce the computational complexity, an approximate nearest neighbor search using the Kd-tree approach is adopted in this article.

A. Kd-tree Based Nearest Neighbor Computation

A K-dimensional tree, or Kd-tree is a space-partitioning data structure for organizing points in a K -dimensional space. ANN(Approximate Nearest Neighbor) is a library written in C++ [12], which uses Kd-tree for both exact and approximate nearest neighbor searching in arbitrarily high dimensions. In this article ANN is used to find exact d_i s, where $i = 1, \dots, knear$, in Equation 2 efficiently. Note that ANN assumes that distances are measured using any class of distance functions called Minkowski metrics. Here d_i s are the Euclidean distances between \bar{x}^* (the reflected point of \bar{x} with respect to \bar{c}) and $knear$ unique nearest neighbors of it. The Kd-tree structure can be constructed in $O(N \log N)$ time and takes $O(N)$ space. Friedman et al. [13] reported $O(\log N)$ expected time complexity for finding the nearest neighbor using Kd-based data structure. Hence for N points, the complexity of computing the symmetrical distances of N points with respect to a particular center is $O(N \times \log(N))$.

III. HUMAN FACE DETECTION ALGORITHM

Our human face detection algorithm uses the symmetry property of human faces to quickly locate candidate face from some background. Here we assume that the human face can be approximated by an ellipse, therefore, our goal is to quickly locate ellipsoidal-shaped objects in an image. In fact it is not a trivial task since the sizes and orientations of the

objects of interest may vary a lot. In order to efficiently solve the problem the following five step procedure is adopted.

- 1) Use the Sobel edge detection operator [14] to find the edge image.
- 2) Delete short segments that contain less than five pixels. This step is executed by using connected component analysis. The algorithm for finding 8-connected component of an image is found in [14]. Here at first the 8-connected components of the edge image are found and then the connected components having less than 5 pixels are removed from the edge image.
- 3) Use a $w_1 \times w_2$ window to scan the processed edge image from top to bottom and from left to right. We use the above mentioned d_{ps} distance to measure the degree of symmetry of the object within the window relative to the center of the window. Here corresponding to each window one counter, $Count_w$ is maintained. For a particular window w , where there are a total of N_w number of edge pixels, for each edge pixel \bar{x}_i , $1 \leq i \leq N_w$ of that particular window w , calculate $d_{ps}(\bar{x}_i, \bar{c})$ (by Equation 2) where \bar{c} is the center of the window. If $d_{ps}(\bar{x}_i, \bar{c}) < \theta$, then $count_w$ is increased by 1. Therefore the final value of $count_w$ represents the number of symmetrical pairs in the window. We also provide a rough guideline of the choice of θ , the threshold value on the PS-distance. It is to be noted that if a point is indeed symmetric with respect to some cluster centre then the symmetrical distance computed in the above way will be small, and can be bounded as follows. Let d_{NN}^{max} be the maximum nearest neighbor distance in the data set. That is

$$d_{NN}^{max} = \max_{i=1, \dots, N} d_{NN}(\bar{x}_i), \quad (3)$$

where $d_{NN}(\bar{x}_i)$ is the nearest neighbor distance of \bar{x}_i . Assuming that \bar{x}^* lies within the data space, it may be noted that

$$d_1 \leq \frac{d_{NN}^{max}}{2} \quad \text{and} \quad d_2 \leq \frac{3d_{NN}^{max}}{2}, \quad (4)$$

resulted in, $\frac{d_1 + d_2}{2} \leq d_{NN}^{max}$. Ideally, a point \bar{x} is exactly symmetrical with respect to some \bar{c} if $d_1 = 0$. However considering the uncertainty of the location of a point as the sphere of radius d_{NN}^{max} around \bar{x} , we have kept the threshold θ equals to d_{NN}^{max} . Thus the computation of θ is automatic and does not require user intervention.

- 4) Sort the value of $count_w$ in the decreasing order. Then the window with the largest $count_w$ locates the most possible region containing a face.

IV. EXPERIMENTAL RESULTS

Our face detection algorithm was tested on a face data base collected from CMU [15]. Some examples of how the proposed face detection algorithm work to detect faces from some images are shown in Figures 1-6. Figures 1(a), 2(a), 3(a), 4(a), 5(a), 6(a) show respectively the six original images on which the proposed algorithm is applied. Figures 1(b), 2(b), 3(b), 4(b), 5(b), 6(b) show respectively the six edge

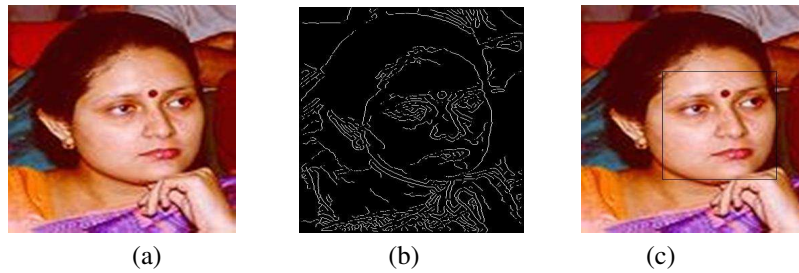


Fig. 1. (a) Original image (b) the result after applying Sobel filter (c) the detected face region

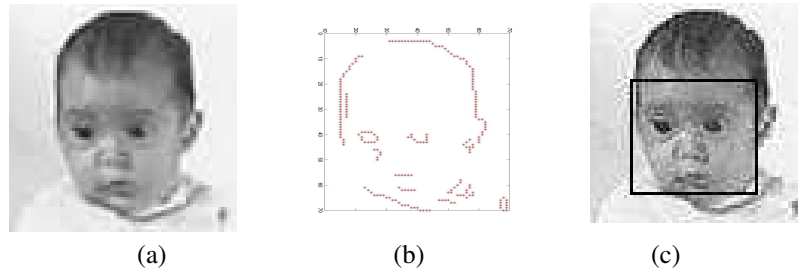


Fig. 2. (a) Original image of a baby face (b) the result after applying Sobel filter (c) the detected face region

images after application of the sobel edge detection operator. Figures 1(c), 2(c), 3(c), 4(c), 5(c), 6(c) show respectively the detected human faces from the corresponding images after application of the proposed method. Note that the present work can only detect one face from a particular image. For that reason, from Figures 5(a) and 6(a) where there are two human faces in the images, the proposed method is only able to detect one single human face. But the proposed method is easily extended to detect multiple faces from an image by keeping track of the windows having second highest number of symmetrical pairs etc. Note that in Figure 6(a), a balloon is present which is of ellipsoidal shaped. But the proposed algorithm is still able to detect the window containing face because number of symmetrical points present in the window containing human face is much larger than that present in the window containing the balloon.

V. CONCLUSION AND FUTURE WORK

This paper reports about the development of a face detection system where the symmetry exists in the human faces is used to detect human face from an image. The effectiveness of the proposed method is demonstrated in detecting faces from six real-life images. The proposed method can be extended in many ways. Current work is going on in order to extend it to detect multiple face regions from an image containing multiple faces. The effectiveness of the proposed method should be evaluated on some standard face databases. Comparisons of the present algorithm with the existing techniques is another major future work.

REFERENCES

[1] R. Brunelli and T. Poggio, "Face recognition: Features versus templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, pp. 1042–1052, 1993.

[2] L. Harmon and W. Hunt, "Automatic recognition of human face profiles," *Computer Graphics and Image Processing*, vol. 6, pp. 135–156, 1997.

[3] D. Valentin, H. Abdi, A. J. O'toole, and G. W. Cottrell, "Connectionist models of face processing a survey," *Pattern Recognition*, vol. 27, no. 9, pp. 1209–1230, 1994.

[4] W. Zhao, R. Chellappa, A. Rosenfeld, and P. Phillips, "Face recognition: A literature survey," *ACM Computing Surveys*, pp. 399–458, 2003.

[5] T. Sakai, M. Nagao, and S. Fujibayashi, "Line extraction and pattern recognition in a photograph," *Pattern Recognition*, vol. 1, pp. 233–248, 1969.

[6] V. Govindaraju and M. Rajapakse, "Verification of faces in cluttered photographs," *IEEE International Conference on Systems, Man and Cybernetics*, pp. 2028–2033, 1997.

[7] A. Yuille, P. Hallinaw, and D. Cohew, "Feature extraction from faces using deformable templates," *Internationsl Journal of Computer Vision*, no. 2, pp. 99–111, 1992.

[8] K. K. Sung and T. Poggio, "Example-based learning for view-based human face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 39–51, 1998.

[9] S. Bandyopadhyay and S. Saha, "GAPS: A clustering method using a new point symmetry based distance measure," *Pattern Recog.*, Accepted (March, 2007), URL: www.isical.ac.in/~sriparna_r/paper.html.

[10] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf, *Computational Geometry: Algorithms and Applications*, 2nd ed. Springer-Verlag, 2000. [Online]. Available: <http://www.cs.uu.nl/geobook/>

[11] M.-C. Su and C.-H. Chou, "A modified version of the k-means algorithm with a distance based on cluster symmetry," *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 674–680, 2001.

[12] D. M. Mount and S. Arya, "ANN: A library for approximate nearest neighbor searching," 2005, <http://www.cs.umd.edu/~mount/ANN>.

[13] J. H. Friedman, J. L. Bently, and R. A. Finkel, "An algorithm for finding best matches in logarithmic expected time," *ACM Transactions on Mathematical Software*, vol. 3, no. 3, pp. 209–226, 1977.

[14] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Massachusetts: Addison-Wesley, 1992.

[15] <http://www.ius.cs.cmu.edu/IUS/usrp0/har/FaceDemo/gallery-inline.html>.

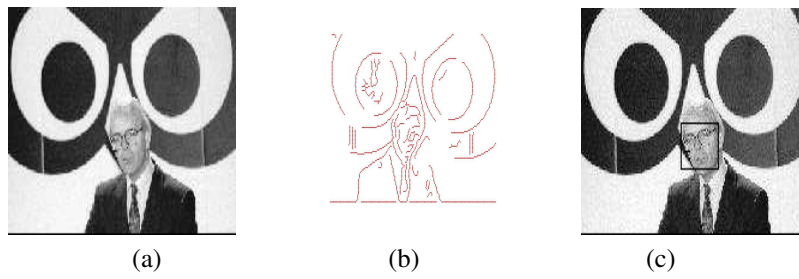


Fig. 3. (a) Original image of a human face (b) the result after applying Sobel filter (c) the window having largest symmetrical edge pairs, i.e., the detected face region

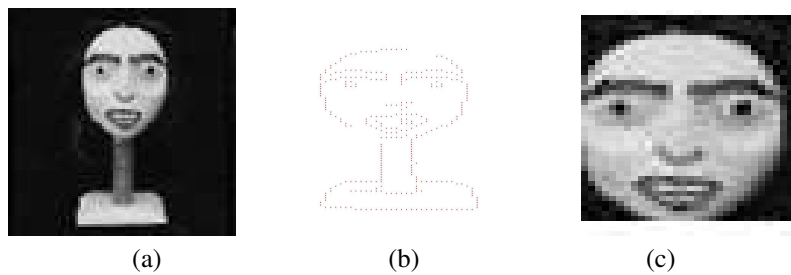


Fig. 4. (a) Original image of face of a puppet (b) the result after applying Sobel filter (c) the detected face region

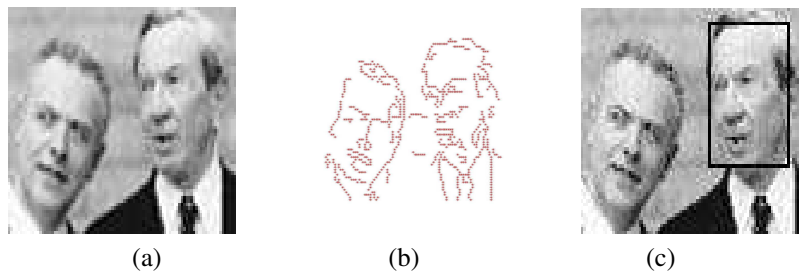


Fig. 5. (a) Original image of two human faces (b) the result after applying Sobel filter (c) the window having largest symmetrical pairs e.g., the detected face region

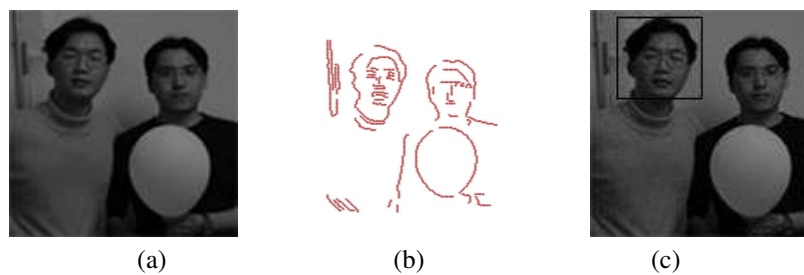


Fig. 6. (a) Original image of a two human faces and a balloon (b) the result after applying Sobel filter (c) the window having largest symmetrical pairs e.g., the detected face region