Face Recognition based on Curvature Estimation and Neural Networks

M. S. Eng. M. Aiman Al-Akkad
CAD Department
Izhevsk State Technical University, Izhevsk, Russia
aiman@udm.net

Abstract

In this paper, we present a method for constructing a 3D net in the eyes regions by detecting the iris and calculating Gauss curvature, and then regarding to this we plot the 3D net of the human face. After this the Gauss curvature data of the image local regions get to the classifier, which gives weight coefficients vector of the probability that the original image belongs to some class of the overall number of classes recognized by the system.

Keywords: *Recognition, 3D model construction, Curvature Estimation, Neural Networks.*

1. INTRODUCTION

The basic neural network (NN) systems applications in image processing and recognition are:

- Images attributes extraction.

- Classification of images or characteristics extracted from them.

- Optimization.

In the algorithm of the local attributes analysis, the technology of NN is used.

NN consists of elements called neurons, which are bonded with other ones, where every neuron transforms a set of signals coming to its input into an output signal. The bonds between neurons, which are coded by weights, play the key role.

NN have good generalizing ability, because they can distribute the experience got in the final learning set for the whole images set.

It is necessary to pick out key attributes, characterizing visual image, determine the attributes relative value choosing their weight coefficients and taking into consideration mutual relations between them. The recognition system consists of:

- Scanning and preprocessing stage (location aligning, segmentation, noise removal, and other improvements).

- Image description receiving (extracting the qualities characterizing the image in view of saved templates, for example of surface curvature).

- Classifying algorithms based on artificial NN and Expert system block (analyzing extracted qualities and making the final decision about the identification procedure execution). The classifier is built on the base of artificial neural network, multilayer perceptron topology.

1.1. Review Of Neural Network Approaches

The earliest methods of NN for face recognition used Kohonen's associative map [3], where a small set of face images was used, and accurate recognition was reported even when the input image was noisy and distorted. This capability has also been demonstrated using optical hardware [4].

Later NN approach for gender classification [5] was used, which automatically extracted a 16-dimensional feature vector such as eyebrow thickness, widths of nose and mouth, six chin radii, etc. Two Hyper Basis Function (HyperBF) networks [6] were trained, one for each gender. The input images were normalized with respect to scale and rotation relative to the positions of the eyes. The two outputs were compared and the one with greater value determines the gender label for the test image. In the actual classification experiments only a subset of the feature vector was used. The database consisted of 21 males and 21 females. The leave-one-out strategy [2] was employed for classification. When the feature vector from the training set was used, 92.5% correct recognition accuracy was reported and for faces not in the training set, the accuracy dropped to 87.5%. By using an expanded 35-dimensional feature vector, and one HyperBF per person, the gender classification approach has been extended to face recognition.

Methods [7] and [8] were based on the Dynamic Link Architecture (DLA), in an attempt to solve the main problem of the expression of syntactical relationships in conventional ANN. Both methods used Gabor based wavelets for the features. DLAs use synaptic plasticity and are able to instantly form sets of neurons grouped into structured graphs and maintain the advantages of neural systems. For DLA an image domain and a model domain are needed. The image domain corresponds to primary visual cortical areas and the model domain to the inter-temporal cortex in biological vision. The DLA architecture has been recently extended to Elastic Bunch Graph Matching (EBGM) [9, 10], similar to the method described above, but with attaching a set of jets instead of one, each derived from a different facial image. To handle the pose variation problem, the face pose is first determined using prior information [1] and the transformations of the sets under pose variation are learned [11].

2. DESCRIPTION RECEIVING

Description should include the largest possible number of qualities discriminating certain man image from another. Figure 1 shows the diagram of the general algorithm.

Constructing the polygonal net in the eyes region consists of:

- Find eyes irises and to plot the net regarding to them.

- Calculate Gauss curvature in this region.

Input data: The output of the scanning module.

Output data: Gauss curvature values of local regions around eyes.

Left Eye matrix of 5×7 size, elements of which $L_{ij} \in R$, where $i \in [1,5]$; $j \in [1,7]$.

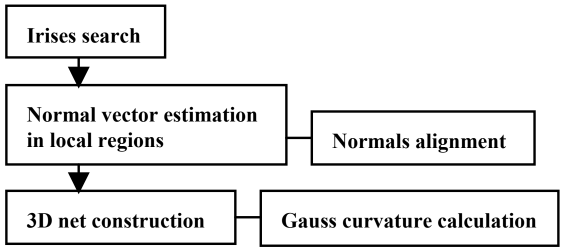


Figure 1: The algorithm of the image characteristics extraction.

2.1. Eyes Localization Algorithm (Irises Search)

Eyes are the main detail on the face for automatic recognition. As the distance between the Irises centers does not change with different expressions, it can be used for defining the other face elements. Early defining techniques were based on finding a pair of local minima. Later template comparison methods were used.

Input data: Processed raster image. Processed Image matrix of size $W \times H$, elements of which $E_{ij} \in Z^+$, where $i \in [1, H]; j \in [1,W]; H \in Z^+; W \in Z^+. W= 200; H= 200$

Output data: Raster images of local regions around eyes. Left Eye matrix of 50×70 size, elements of which $L_{ij} \in Z+$, where $i \in [1,50]; j \in [1,70]$.

Algorithm contains two steps:

a) Threshold filtration: Irises localization is based on the suggestion that it is a dark disk with light edge surrounding. Face image is segmented with very low threshold for isolating 5% of the darkest pixels [13,14], where these pixels can belong to irises as shown in Figure 2. Threshold is calculated as $T=\frac{\sum_{i=1}^{H}\sum_{j=1}^{W}E_{ij}}{H*W*3}$ where E_{ij} is pixel brightness in the point (i,j). Then we construct filtered 2-gradational raster image:

$$F_{ij} = \begin{cases} 0, & \text{if } E_{ij} \le T \\ 1, & \text{otherwise} \end{cases}$$

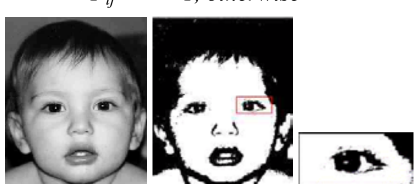


Figure 2: An original image from the database, the *200 x 200* raster image produced by the system, the *50 x 70* window around the left eye region, and the thresholded iris disk for the left eye.

b) Template comparison: The template will concern only one eye, for the second one everything is similarly done. Correlation between disk D_{ij} and filtered image pixels F_{ij} is the number of coinciding points calculated as

$$C=\sum_{i,j\in w} F_{ij}\Big|_{F_{ij}=D_{ij}=0}$$

where w is the left or right half of eye window for two irises correspondently, D_{ij} is a dark or light point on the disk described as

$$D_{ij} = \begin{cases} 0, & \text{if } (i-x_0)2+(j-y_0)2 \le r2 \\ 1, & \text{otherwise} \end{cases}$$

for the circle with center (x_0,y_0) and radius r described as

$x_0 \in (W/2,W-(20+r))$ *for the right eye*

$x_0 \in (20+r,W/2)$ *for the left eye*

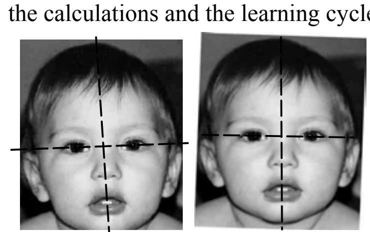
$y_0 \in (20+r,H/2-r)$

$r \in [8,12]$

From the x_0, y_0, r values and correlational maximum C_{max} we get the hypothetic centers (x_h, y_h) and the resulting matrix will be $L_{i,j} = E_{i,j}$ where $i \in [x_h-25,x_h+25]$ and $j \in [y_h-35,y_h+35]$ and analogously for the right eye.

2.2. Face image Alignment

The line drawn between the two white points inside the pupils, we call it the eyes-centers line and a perpendicular line on it passing through the nose tip we call it the face-perpendicular line. Then we align the photo according to the eyes-centers line by rotating the face-window by an amount equal to the angle between the horizontal line and the eyes-centers line, so that the eyes-centers line becomes coincident with the *x*-axis as shown in Figure 3. This alignment unifies the face bending between all photos and reduces the calculations and the learning cycle.



(a) (b)

Figure 3: (a) Photo with the main alignment lines (b) the aligned photo by 2 degrees rotation.

2.3. Polygonal net drawing

- Normal vectors construction of the surface depicted on the image (For normal vector determination, iteration Horn method is used [12,13,14]).
- Calculating the depth Z according to the normal vectors.

2.3.1. Iterational Normals Definition (Normals Vectors Finding and Aligning)

The method suits diffusively dispersive surfaces (Lambertian) with same light reflecting/absorbing ability (albedo/reflectance) all over the area, with known light source direction. Object form reconstruction from photo will be produced for noisy (real) photos, data of which cannot have theoretical solution.

Input data: Raster images of local regions around eyes. Left eye image matrix of size 50×70, elements of which $L_{ij} \in Z+$, where $i \in [1,50]$ and $j \in [1,70]$.

Output data: Normals vector field of local regions around eyes. Left eye image matrix of size 50×70, elements of which $N_{ij} \in L$, where $i \in [1,50]$ and $j \in [1,70]$ and L is the 3D space.

Algorithm: Iterational equations system is used for normal vectors calculation.

We calculate the normal vector for every point of left eye image where the normals vectors initial values are $(0,0,1)$. Experiments proved, that for our purposes *4* iterations are enough $k \in [0,3]$. After iteration cycles complete, normals aligning is carried out. We'll find average normal as $n_{avg}=n^{(3)}_{i/2,j/2}$, we set the value $n^{(3)}_{i/2,j/2} = (0,0,1)^T$, then any normal on the image will be:

$$N_{ij} = n^{(3)}_{i,j} - n_{avg}$$

2.3.2. Surface Curvature Definition (3D Net Construction, Angles and Curvature Calculation)

For a surface described by $z=f(x,y)$, R_1 , R_2 are its major curvature radii, its mean curvature in a point is $H = ½ (1/R_1 + 1/R_2)$ and its Gauss curvature is $K = 1/R_1R_2$.

For a cylinder with radius a: $H =1/2a$, $K = 0$; for elliptical points $K > 0$, for hyperbolic $K < 0$, for parabolic $K = 0$.

H and K are calculated [12,13] using the following formulas:

$$H = r(1 + q^2) – 2 p q s + t (1 + p^2) / 2(1 + p^2 + q^2)^{3/2}$$

$$K = r^t - s^2 / (1 + p^2 + q^2)^2$$

To calculate the derivatives $p, q, r, s, t,$ we will use the discrete approximation of the surface using final differences method, thus transforming the net depth function Z into continuous surface function z. Z is defined discretely in the entire surface multitude with the unit net step.

For p and q instead of using the left difference derivatives, we will use central ones:

$$P = \partial z/\partial x = Z_{x+1,y} - Z_{x-1,y} /2$$

$$q = \partial z/\partial y = Z_{x,y+1} - Z_{x,y-1} /2$$

Then we derive:

$$r = \partial^2 z/\partial x^2 = p_{x+1,y}-p_{x-1,y}/2 = Z_{x+2,y}–Z_{x,y}/4-Z_{x,y}–Z_{x-2,y} /4$$

$$s = \partial^2 z/\partial x\partial y = p_{x+1,y}-p_{x,y-1}/2 = Z_{x+2,y}–Z_{x,y}/4-Z_{x,y}–Z_{x,y-2}/4$$

$$t = \partial^2 z/\partial^2 y = p_{x,y+1}-p_{x,y-1}/2 = Z_{x,y+2}–Z_{x,y}/4-Z_{x,y} – Z_{x,y-2}/4$$

By computing p, q, r, s, t values, it is possible to calculate H and K in every image point.

Input data: Normals vector field of local regions around eyes. Left eye matrix of size 50×70, elements of which $N_{ij} \in$ L, where $i \in [1,50], j \in [1,70]$, L is the 3D space.

Output data: Gauss curvature values of local regions around eyes. Left eye matrix of size 5×7, elements of which $K_{ij} \in R$, where $i \in [1,5], j \in [1,7]$.

Algorithm: 3D net (depths map) is constructed by z coordinate calculating in every point

$$N_{ij} = \{LENx, LENy, LENz\} . Z_{ij} \in R$$

$$z_0 = LENz_{11}$$

$$Z_{ij} = LENz_{ij} + z_0$$

$$z_0 = LENz_{ij}$$

Figure 4 shows the constructed 3D net for the local regions around the left eye.

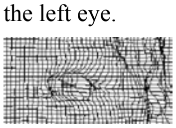


Figure 4: The calculated 3D net for the left eye.

3. CLASSIFIER AND EXPERT SYSTEM MODULE

Input data: Gauss curvature values of local regions around eyes. LeftEyeK matrix of 5×7 size, elements of which $LeftEyeK_{ki,kj} \in R$, where $ki \in [1,5]; kj \in [1,7]$.

Output data: Weight coefficients column vector of the probability of original image belonging to some class of $Class\ Amount \in N$ dimension, overall number of classes recognized by system.

$Output_{ca} \in R$, where $ca \in [1, Class\ Amount]$

Gauss curvature data of image local regions get to classifier.

Classifier is a calculating model on the base of artificial neural network (ANN) consisting of simple elements (neurons) and bonds between their outputs (synapses). ANN topologies stable to affine transformation and noise. Classifier is realized on the base of multilayer perceptron [15,16,17]. Input layer neurons number equals $k_0 =35$, there are $k_1 =70$ neurons in hidden layer, and in the output layer neurons number coincides with the number of classes to be recognized $k_2 = Class\ Amount$.

Learning is carried out using reverse error distribution method. As a reducing (logistic) function sigmoid is taken: $y_j^{(n)} = \dfrac{1}{1 + e^{s_j^{(n)}}}$

3.1. The Learning Algorithm:

With the help of reverse distribution, the procedure is constructed as follows:

a) Send to the net inputs one of the images and calculate the values of the outputs.

b) Calculate $\delta(N)$, $\delta(2)$ for the output layer according to the formula:

$$\delta_l^{(N)} = (y_l^{(N)} - d_l).\frac{dy_l}{ds_l}, \text{ where } l \in [1,k_2]$$

$$dl = \begin{cases} 1, \text{ if neuron on the output coincides with given class,} \\ 0, \text{ if neuron does not coincide with given class.} \end{cases}$$

c) Calculate $\delta(n)$ and $\Delta w(2)$ for all the rest layers according to the formula:

$$\Delta w_{ij}^{(N)} = -\eta.\delta_j^{(N)}.y_i^{N-1}$$

d) Correct all the weights in NN.

If the net error is essential, go to step (a), otherwise end.

The output of the expert system module is the belonging class number: $Answer \in [1, Class\ Amount]$.

3.2. Database

The training database is an object-oriented database, which contains initially *216* images, where a single image of each person exists during recognition. Face images are main classes/objects, which consist of sub-classes/sub-objects, which actually are the face features. Multiple images of every person are created by a synthesis process from the initial images [13] during training in a one-to-many fashion.

4. EXPERIMENTAL RESULTS

Initially to test our system we chose a test image set of *15* face images, preprocessed and reduced to size *200x200* pixels and *16* million shades of gray. Images are in different shooting conditions, various head turn angle, lighting conditions, and face expressions.

First in some of the samples, irises were not found; in others irises were not correctly found, probably due to non-uniform illumination, which led us to equalize the brightness in the sample photos and scale the images to the same size.

After preprocessing we noticed considerable enhancement in identifying the irises and therefore in the overall system performance.

The results showed that our system is promising, and leads to satisfactory face recognition results as shown in figure 5.

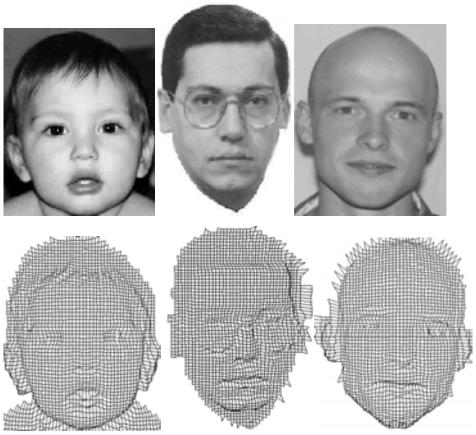


Figure 5: Examples of the constructed 3D models.

5. CONCLUSION

We illustrated an image recognition method based on neural network model, where we realized:

- Eye irises search algorithm.
- Average and Gauss curvature defining module.
- Images classifier based on the Multilayer Perceptron NN topology.

The advantages of NN are the universal learning mechanisms and ability of all their elements to function parallel thus increasing the efficiency of solving the task in comparison with other methods as probabilistic methods, linear separators, decisive trees, etc.

5.1. Comparison with Other Approaches

Usually NN methods are dedicated either for real-time [19] or none real-time [18] applications. Also NN methods use more than one photo in the database for every person [15]. Our method is good for both real-time and non real-time applications, and it doesn't need more than one photo for each person. This consumes less storage space in the database system, and makes our method faster and more versatile.

REFERENCES

[1] N. Kruger, M. Potzsch, and C.v.d. Malsburg, "Determination of Face Position and Pose with a Learned Representation Based on Labelled Graphs", Image and Vision Computing, Vol. 15, pp. 665-673, 1997.

[2] K. Fukunaga, Statistical Pattern Recognition, New York: Academic Press, 1989.

[3] T. Kohonen, Self-Organization and Associative Memory, Berlin: Springer-Verlag, 1988.

[4] Y. S. Abu-Mostafa and D. Psaltis, "Optical Neural Computers", Scientific American, Vol. 256, pp. 88-95, 1987.

[5] R. Brunelli and T. Poggio, "HyperBF Networks for Gender Classification", in Proceedings, DARPA Image Understanding Workshop, pp. 311-314, 1992.

[6] T. Poggio and F. Girosi, "Networks for Approximation and Learning", Proc. IEEE, Vol. 78, pp. 1481-1497, 1990.

[7] J. Buhmann, M. Lades, and C.v.d. Malsburg, "Size and Distortion Invariant Object Recognition by Hierarchical Graph Matching", in Proceedings, International Joint Conference on Neural Networks, pp. 411-416, 1990.

[8] M. Ladesraj, J. Vorbruggen, J. Buhmann, J. Lange, C.v.d. Malsburg, and R. Wurtz, "Distortion Invariant Object Recognition in the Dynamic Link Architecture", IEEE Trans, on Computers, Vol. 42, pp. 300-311, 1993.

[9] L. Wiskott, J.M. Fellous, N. Kruger, and C.v.d. Malsburg, "Face Recognition and Gender Determination", in Proceedings, International Workshop on Automatic Face and Gesture Recognition, pp. 92-97, 1995.

[10] L. Wiskott, J.M. Fellous, and C.v.d. Malsburg, "Face Recognition by Elastic Bunch Graph Matching", IEEE Trans, on Pattern Analysis and Machine Intelligence, Vol. 19, pp. 775-779, 1997.

[11] T. Maurer and C.v.d. Malsburg, "Single-View Based Recognition of Faces Rotated in Depth", in Proceedings, International Workshop on Automatic Face and Gesture Recognition, pp. 176-181, 1996.

[12] M. J. Brooks and B. K. P. Horn, Shape and Source from Shading. Number AIM-820. Artificial Intelligence Lab., MIT press, 1985.

[13] M. A. Al-Akkad and V. N. Kochuhanov, "A framework for building a general 3D model", Izhevsk, May 2003.

[14] G.G. Gordon, "Face recognition based on depth and curvature features", In Proc. IEEE Conf. Comput. Vis. Patt. Recogn., pp. 808–809. 1992.

[15] A. C. Tsoi, S. Lawrence, C. L. Giles and A. D. Back, "Face Recognition: A Convolutional Neural Network Approach", IEEE Transactions on Neural Networks, Special Issue on Neural Networks and Pattern Recognition, 1998.

[16] J. Okamoto, M. Milanova, P. E. Almeida, and M. G. Simoes, "Applications of Cellular Neural Networks for Shape from Shading Problem", 1999.

[17] S. Gutta and H. Wechsler, "Face recognition using hybrid classifiers", 1997.

[18] A. Pentland, T. Starner, N. Etcoff, A. Masoiu, O. Oliyide, and M. Turk, "Experiments with eigenfaces", In *Looking at People Workshop, International Joint Conference on Artificial Intelligence 1993*, Chamberry, France, 1993.

[19] R. Chellappa, C.L. Wilson, and S. Sirohey, "Human and machine recognition of faces: A survey", *Proceedings of the IEEE*, 83(5): 705–740, 1995.

About the author

Aiman Akkad is a Ph.D. student at Izhevsk State Technical University, CAD Department. His contact email is aiman@udm.net.