Detection and Recognition of the Face from the Picture using Associative Neural Networks

Author – Ognjen Zelenbabić

Abstract – This paper presents an approach to analyze the ability of Hopfield network to remember and recognize human faces. Special attention will be devoted to the process of selecting the appropriate neural network architecture and the proper processing of digital images. Algorithm developed for the purpose of testing of above mentioned is robust and can be put to practical use.

INTRODUCTION

The problem of face recognition from the picture is modern problem for which there is still no solution that would represent the equivalent in quality to the successfulness of recognition that is inherent in human beings. However, the opportunity to deal with this issue in general was created thanks to the creation of the theory of artificial neural networks. The publication of Hopfield's seminal papers [1982, 1984] started the modern era in neural networks.

Problem definition: The neural network must be trained to remember a number of different people. If we bring the image of a person that exists in neural network's database to the network input, neural network has to recognize and identify that person. Remembered images and image that is subject of testing for the same person must be different. If we bring the image of the unknown person to a networks input, neural network has to identify that person as an unknown person.

The problem can be divided into two parts:

1. Appropriate digital image processing.
2. Making the correct choice of Hopfield network architecture and its software implementation.

Proper digital image processing is very important for the testing of the effectiveness of Hopfiled network recognition quality. The demands placed on the algorithm for digital image processing are as follows:

1. The algorithm has to be able to extract face from the picture of a person.
2. The resultant image must have enough identity properties of the person's face.
3. The algorithm must be resistant to light conditions under which a person is photographed.

No matter how important digital image processing is for solving of this problem, the selection of Hopfield network architecture is as equal important. In fact, these two problems are not two separate issues. The manner of dealing with either one of them is directly conditioned by the manner of solving of the other one.

The demands placed on the algorithm for defining the structure of the Hopfield network are:

1. Neural network has to be able to remember arbitrary number of different persons.
2. Neural network must recognize the person's image that is brought to the network's input even if the person is photographed under different conditions and has different identity properties (haircut, beard, age, facial expression etc.).
3. Algorithm's execution time for the testing of the recall should be little so that program can be used in practice.

As I have already mentioned earlier, these two problems are not separate entities and they must be solved simultaneously. It is for this reason that I conducted extensive testing both to the manner of the digital image processing and to the structure of associative neural network. Further development of the algorithm was dependent on the previously obtained results. Testing was conducted on 60 different pictures made for five different persons.

The focus of this paper are the results obtained by testing, therefore theory will be concise as much as possible.

1. Discrete Hopfield networks

For the programming of the Hopfield networks, I used the algorithms shown in the following theory.

Hopfield network is a single layer network with feedback, its detailed network configuration is shown in Figure 1.

When a single-layered recursive network performs a sequential process of updating, we first input a series of data and then the output is generated. Then the input data set is removed and the initialized output becomes the new, updated input through the feedback. The first updated input gives us first updated output which now actually represents the second updated input through the feedback, which then gives us another updated output. The transition process continues until the network does not receive new updated output, that is until the network reaches equilibrium.

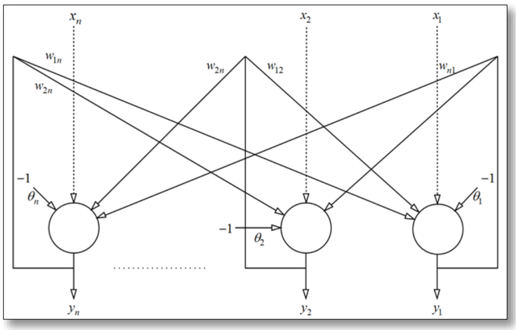


Figure 1: Hopfield network.

Update rule of each node in the discrete Hopfield network is given by:

$$y_i^{(k+1)} = \operatorname{sgn}\left(\sum_{\substack{j=1 \\ j\neq i}}^{n} w_{ij} y_j^{(k)} + x_i - \theta_i \right) \qquad (1)$$

for $i = 1, 2, \ldots, n$.

2. Associative memory

An associative memory can store a set of patterns as memories. When the associative memory is presented with a *key pattern,* it responds by producing whichever one of the stored patterns most closely resembles or relates to the key pattern. Hence, the recall is through association of the key patter with the information memorized. Such memories are also called *content-addressable memories* in contrast to the traditional *address-addressable memories* in digital computers in which a stored pattern is recalled by its address.

The basic concept of using Hopfield networks as associative memories is to interpret the system's evolution as a movement of an input pattern toward the one stored pattern most resembling the input pattern.

3. Recurrent Autoassociative Memory

A Hopfield memory is able to recover an original stored vector when presented with a probe vector close to it. In Hopfiled memory, Eq. (1) is the data *retrieval rule*. The remaining problem is how to store data in memory.

Assume bipolar binary vectors that need to be stored are x^k for $k = 1, 2, \ldots, p$. The *storage algorithm* for finding the weight matrix is:

$$w_{ij} = \sum_{k=1}^{p} x_i^k x_j^k \, , \; i \neq j; \; w_{ii} = 0 \qquad (2)$$

where $x^k = (x_1^k, x_2^k, \ldots, x_n^k)^T$. If x^i are unipolar binary vectors, that is, $x_i^k \in \{0,1\}$ then the storage rule is:

$$w_{ij} = \sum_{k=1}^{p} (2x_i^k - 1)(2x_j^k - 1) \qquad (3)$$

for $i \neq j$, $w_{ii} = 0$.

Additional autoassociations can be added to the existing memory at any time by superimposing new, incremental weight matrices. Autoassociations can also be removed by respective weight matrix subtraction.

4. Hopfield memory issues

There are two major problems of Hopfield memories. The first is the unplanned stable states, called *spurious stable states*. It can be shown that memory of transitions may terminate as easily at x as at $\overline{x}$. The second is uncertain recovery, which concerns the capacity of a Hopfield memory. Overloaded memory may result in a small Hamming distance between stored patterns and hence does not provide efficient recovery of stored patterns.

5. Digital image processing

When working on the problem of proper processing of images for the sake of using them in Hopfield network three things have been crucial:

1. Using of histograms in image processing.
2. Otsu's method for determining of the global threshold.
3. Algorithms for edge detection.

- Histogram

Intensity transformation functions based on information gathered from the image intensity histograms represent basics of digital image processing in areas such as improving, compression, image segmentation and description.

I used histograms to find the global intensity threshold of gray colour in order to obtain the appropriate black and white contrast for each image. Also, I used histogram equalization algorithm on picture to make them invariant to the intensity of light at which they were taken.

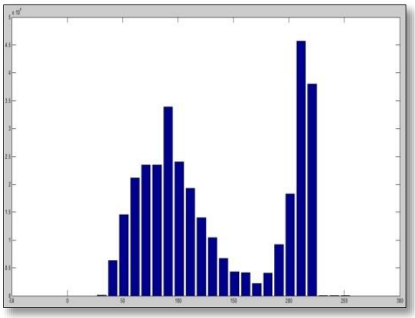


Figure 2: Intensity histogram of the two dominant groups of gray color.

- Otsu's method

In computer vision and image processing, Otsu’s method is used to automatically perform histogram shape-based image thresholding or the reduction of a gray-level image to a binary image. The algorithm assumes that the image to be thresholded contains two classes of pixels (foreground and background) then calculates the optimum threshold separating those two classes so that their combined spread is minimal.

Image Processing Toolbox (IPT) has a function *graytresh* that calculates threshold using Otsu's method.

- Image edge detection

Edge detection is an approach that is mostly used for detection of significant discontinuities in the image intensity. Such discontinuities are detected by using first and second order derivatives.

The basic idea behind edge detection is to find places in an image where intensity changes rapidly, using one of two general criteria:

1. Find places where the first derivative of the intensity is greater in magnitude than a specified threshold.
2. Find places where the second derivative of the intensity has a zero crossing.

IPT's function *edge* provides several derivative estimators based on the criteria just discussed.

NEURAL NETWORKS AND IMAGE PROCESSING

In the following paper I will describe in what ways I have tested algorithm, how the algorithm changed according to the results and what are the best results obtained from testing.

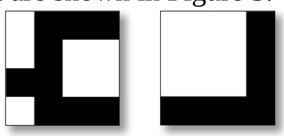
1. Programming Hopfield memory

Before I used the Hopfield neural network for solving of the mentioned problem I have performed exhaustive testing on it. I wanted to make sure in the quality of the association and to gain some idea of what can I expect from Hopfield memory irregularities.

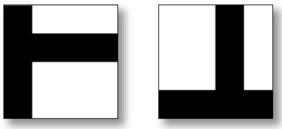
Based on these initial tests on the Hopfield network I can sum up its problematic features:

1. The network capacity must be determined experimentally for a give size of images.
2. There is a false steady state.
3. Improper associations as a steady state.

I tested the network on a simple binary images that are shown in Figure 3.



(a) (b)



(c) (d)

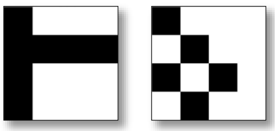
Figure 3: Binary images for testing of the Hopfield network.

- The capacity of Hopfield network

Capacity is the most important parameter of performance of associative memories, because if we exceed the storage capacity we cannot even expect to get a satisfactory association. While on this subject there are exhaustive analysis, the exact method of determining the capacity of Hopfield memory is still not found. It is safest to use one network for one picture, but this solution is not optimal.

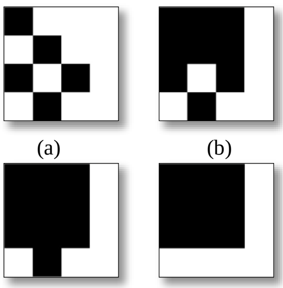
However, the results of tests that I conducted showed that the network is unable to memorize more than one facial image. The reason for this is that on the different facial images we have the same region of the image filled with valuable information. Therefore overlap occurs when we try to remember several facial images in the same network.

An example of poor association due to the overlap of information for the test image (4-a) with 50% of noise (4-b) is shown in Figure 5.



(a) (b)

Figure 4: (a) Original image,
(b) Image with 50% of noise.



(a) (b)

(c) (d)

Figure 5: (a)-(c) Association iterations.
(d) Result of association.

- False steady state

In most cases, the neural network provides good solutions, but in certain cases it gives unsatisfactory solutions. One of the reasons is the so-called false steady state mentioned above and which is manifested as a complement to a steady state value.

This is unacceptable to the problem of face recognition and thus such cases were treated as invalid.

- Incorrect associations as steady state

When we get, as a result of the association, a picture that does not resemble to any of the images in neural network's memory one of the reasons is that the memory capacity is exceeded. Easiest way to deal with this problem is to reduce the number of stored pictures.

With this simple testing of Hopfield memory I got a clear idea of how to pre-process images so that they can be successfully memorized and associated.

2. Image pre-processing

Image needs to be processed in an appropriate way to extract person's facial image from it. In solving of this problem, the most difficult issue was to find a way to make the PC „see“ face on an image. Image processing was performed in order to prepare photos for storage in neural networks. The neural network that uses this images accepts only two values (0 and 1) as inputs. This requirement set the final goal of image processing.

Problem formulation: It is necessary to handle colour image of a person in such way that the facial image is extracted from it and then converted into black and white picture with minimal possible loss of information during conversion.

Resultant image is used for storage and later for association in the neural network. For this reason it is necessary that the resultant black and white image carries as much information as possible and to be of minimal dimensions so that it can be processed faster.

- Detection and separation of face from the image

The first step in image pre-processing is a detection and separation of face from the image. To detect face on the image, I have tried two approaches:

1. Detection by using of colour of the skin.
2. Detection by forming of the person's silhouette.

Detection and separation of face from the image by using of colour of the skin gives good satisfactory results but under precisely defined conditions. The problem occurs when a person in the image has a beard or wears a garment of skin colour or is dark skinned. Also, if the intensity of light is not suitable, the person will not be fully detected.

For these reasons I decided to try another approach I had developed for the sake of testing Hopfield network.

- Detection with person's silhouette

The need to develop a different algorithm than the one that deals with face detection using skin colour arose due to the many issues that exist in this method. The aim of the new algorithm is to be independent of person's skin colour and image light intensity, in other words to be robust and applicable to the largest possible number of cases.

Basic idea of this new approach is to create a person's silhouette and use that image to find borders of head region and then to separate head from the image.

The first step is image conversion from colour to gray level image (Figures 6 and 7).

Figure 6: Original colour image.

Figure 7: Converted, gray level image.

After mentioned conversion a global threshold for the histogram using Otsu's method was calculated. Obtained threshold was then used to convert gray image to black and white image using Matlab's function *im2bw*. Two arguments of this function are gray image and conversion threshold. The result of this conversion is shown in Figure 8.

Figure 8: Image obtained using Otsu's method for threshold calculation.

To form a silhouette of a person in Figure 8, I used the following procedure:

1. Store all black pixels of the current row in variable *x*.
2. If variable *x* is not empty then the location of first black pixel is saved in variable *x_1* and location of the last black pixel is saved in variable *x_2*. Between the memorized locations set all pixels to black.
3. Procedure is repeated for each image row.

When we apply described procedure on Figure 8 the result is as shown in Figure 9. And indeed, as a result we get the silhouette image.

Figure 9: Obtained silhouette image from Figure 8.

However, we can notice that the silhouette image has several “errors”, but these “errors” do not represent a problem for further image processing and we do not have to devote special attention to them.

After forming of the silhouette image it is necessary to filtrate image so that the only information left is the information of interest for further processing (Fig. 10). Specifically, filtering was done in the following way:

1. Count the number of black and white pixels in the current column.
2. If the number of white pixels is greater than the number of black pixels then all pixels in current column are set to white pixels.
3. Procedure is repeated for each column of the image matrix.

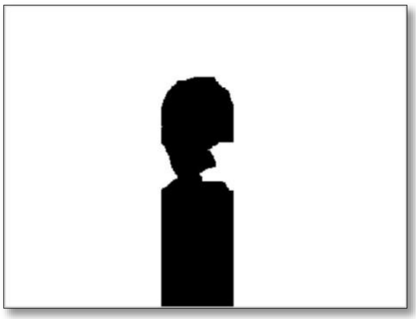
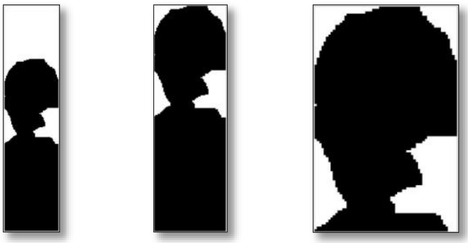


Figure 10: The result of image filtration.



(a) (b) (c)

Figure 11: Separating the head from the silhouettes image.

After image filtration we have such image form that can be used to find the left and right border of the head. The idea is to find the location of first and last black pixel which will then represent left and right border of the head (Fig. 11-a).

It is now easy to determine the location of the upper boundary, we just have to determine the location of the first black pixel from top to bottom (Fig. 11-b).

Lower head boundary can be found by checking the pixels bottom-up. We need to find first row in which the number of black pixels is lesser than the width of the picture, to remember that location and use it as the lower limit (Fig. 11-c).

As a result of a described procedure we obtain separated head from the person’s image (Fig. 12).

Figure 12: Separated image of the head.

Proposed procedure is robust and independent of light intensity. Examples of some other image processing for different light intensities are shown in Fig. 13 and 14.

Figure 13: Low light intensity, person is close to the lens.

Figure 14: High light intensity, person is away from the lens.

TESTING HOPFILD NETWORK

In addition to mentioned shortcomings, Hopfield network has another, for practical use very important, flaw. Namely, the computing speed of the weight matrix (memory) is determined by input image dimensions. Specifically, for an image of 100x100 pixels, weight matrix has 10^4x10^4 elements. This size of a weight matrix is a problem because it takes a lot of time for its processing.

To partly eliminate weight matrix large dimensions, we can further reduce obtained facial image to separate and use only the "eye region". This procedure is fully justified because tests show that the eye region bears enough facial information for recognition of persons.

Reducing of the facial image to eye region is justified for another reason. Namely, the image of the head has excess of information essential for recognizing of individuals. Even if there is no problem with the calculation speed of weight matrix, reducing of the head image to eye region is recommended. In this way we enhance recognition quality of associative neural networks.

Testing of association and memory quality of neural network was conducted on pictures prepared in different ways. Different types of image preparation are as follows:

1. Binary images.
2. Contours of the face images.
3. Combined binary and contour images.
4. Stratified gray images.

For the first three types of image preparation, neural network structure is the same. That is, for one image one weight matrix was calculated. It is important to note that the equalization of histogram was used on each of the tested pictures in order to provide approximately equal light intensity conditions. This was intended to provide approximately equal number of face identification characteristics for every image.

- Testing binary images

Binary images are obtained in the following way:

1. Firstly, equalization of histogram on gray level images was performed.
2. Then, Otsu's method for determining of the threshold for image conversion was used.
3. And finally, conversion to black and white image by using of equalized gray level image and Otsu's threshold.

For the next tests, I used the image of the same person photographed at different time moments (Fig. 15).

Figure 15: Two different images of the same person.

After conversion of these gray level images to binary images, I used one of them to be memorized and the second one as an input of the neural network to test the quality of association (Fig. 16).

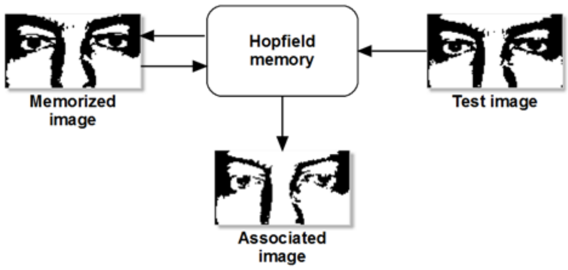


Figure 16: Association quality test for binary images.

- Image contouring

I wanted to try the association with contour images since binary images have too few details. Also it seems like too much information is lost during the conversion to binary image. For this task, I used Robert's edge detection algorithm which produced contour images of best quality.

However, the contours images alone are not sufficient to memorize and associate face images efficiently. Memorizing and association for the same two images in Fig. 15 is shown in the figure below.

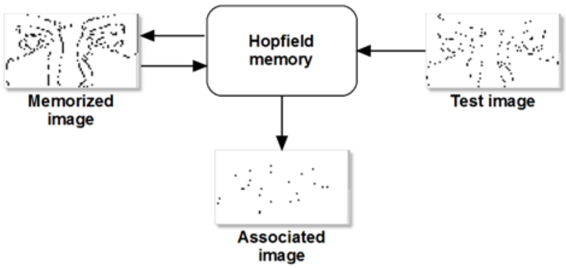


Figure 17: Association quality test for contour images.

- Testing combined images

Combined images approach involves the use of images converted using the Otsu's threshold method and images obtained by detecting edges.

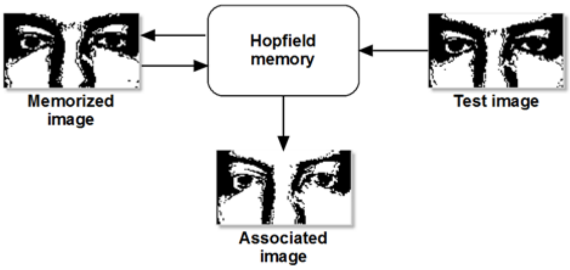


Figure 18: Association quality test for combined images.

- Testing stratified images

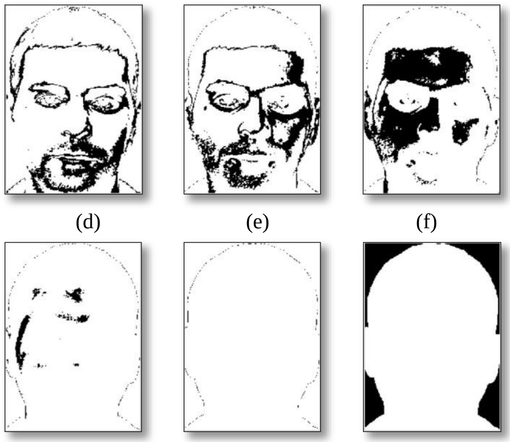
Although binary and combined images are informative enough for a person to be recognized, still they have much less information than the corresponding gray level images.

Stratified images are obtained by dividing each gray level image to eight different images with same dimensions. Each of these eight images carries one part of gray image information and the resulting images are binary.

Since gray images that I used have 256 gray levels, it was enough to split this interval into eight separate intervals in such way that each stratified image has 32 gray levels.

Figure 19 shows original gray image and eight corresponding stratified images.

(a) (b) (c)



(d) (e) (f)

(g) (h) (i)

Figure 19: Original and stratified images.

Figure 20 shows resultant image obtained when we join these stratified images in one gray image.

Figure 20: Original image with eight gray levels.

First, I tested stratified gray images on one Hopfield network, e.g. each one of these eight images are memorized in separate weight matrix. Figure 21 shows results of neural network memorization and association for stratified gray images.

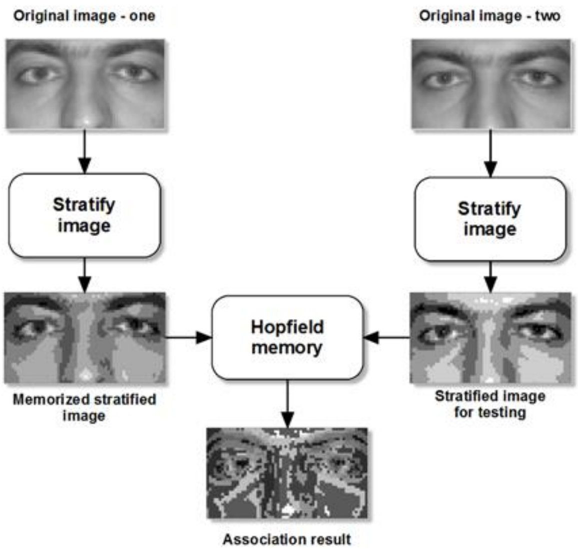


Figure 21: Association result for stratified gray images.

As we can see, this approach does not give satisfactory results. Unfortunately, the results are of same quality even if we use separate weight matrix for each stratified image.

- Testing Hopfield memory

Goal of this further testing is not just to determine the capacity of Hopfield memory but also to test the association quality in dependence to the number of memorized images in one weight matrix and to the Hopfield memory structure.

Testing is divided into two parts:

1. Testing of association quality when several images are memorized in one weight matrix.
2. Testing of association quality when several images of one person, photographed under different circumstances, are memorized in several weight matrices.

- Part One

If neural network ”passes” first test it would be possible to memorize several different images of one person in one weight matrix and in such way improve the association quality.

First test of association quality was conducted on only two images of different individuals memorized in one weight matrix. Memorized images are as shown below.

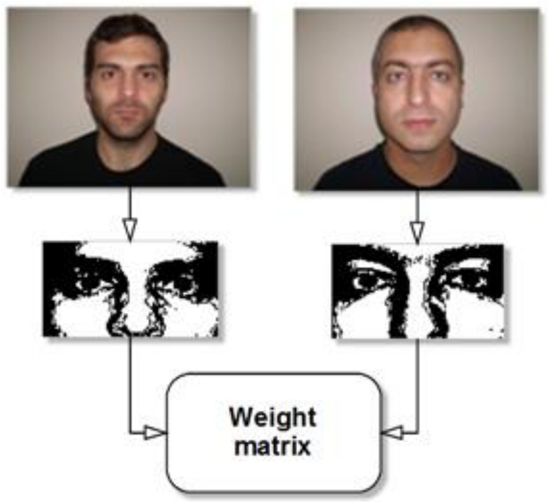


Figure 22: Two images of different individuals memorized in one weight matrix.

Weight matrix was first calculated for the first person from Fig. 22 and then updated for second person. Results are showed in Fig. 23.

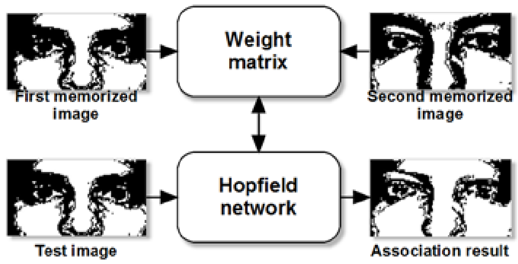


Figure 23: Association result for two different images memorized in one weight matrix.

Resulting image in Fig. 23 does not represent any of two individuals memorized in neural network but combination of their facial information. Therefore we cannot expect that Hopfield network can store several different images in its memory and still provide associations of good quality. On the other hand, that does not mean that the association will be unsuccessful if we memorize several different pictures of the same person in one weight matrix. Intuitively speaking, even if the network does not associate properly, resulting image will be one combination of memorized images, which in case of one person does not necessarily mean unsuccessful association. Still, testing showed that this approach is unusable too.

- Part Two

Focus of this part is to analyze the association quality when several different images of one person are memorized in several weight matrices. Final goal of this approach, to check if person in the image that is brought to the network’s input exist in its database, is based on testing whether it relates to some image from database. If we have several weight matrices used for association, then we also have several association results. If this is the case then we must answer three questions:

1. Does that person exist in database?
2. How to determine which association result is the best result?
3. How well does the best result relate to the input image?

Simple way to solve these problems is to check the “relation grade” for input image and images memorized in database. For each image in database we check similarity with input image "pixel by pixel" and if there is little discrepancy ($\pm 10\%$), then the association quality is marked as satisfying. If the discrepancy is not within defined boundaries then we can discard association result.

For the purpose of testing, let us choose the structure of three different weight matrices, and memorize one different picture in each of these memories (Fig. 24).

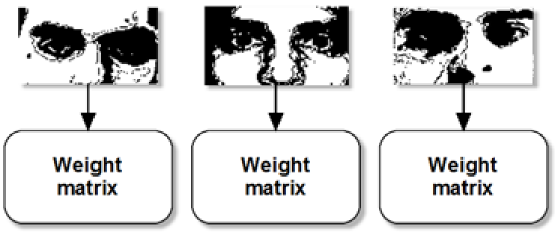


Figure 24: Different images of the same person memorized in separate weight matrices.

Testing was conducted with first picture in Fig. 24 and the result is shown in the next figure.

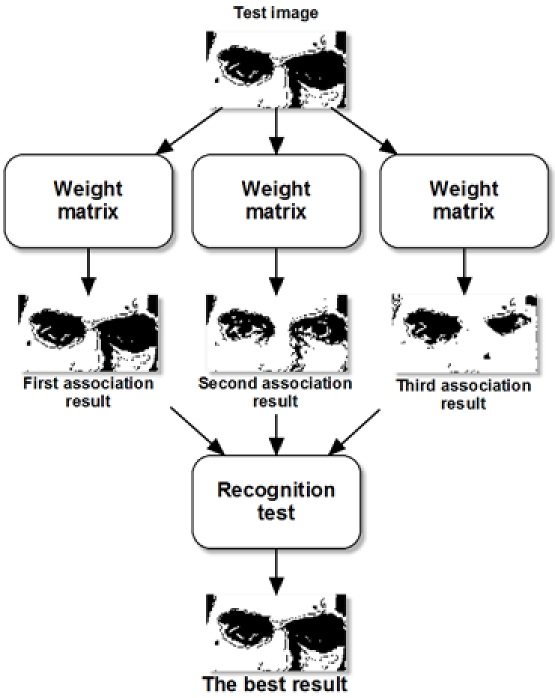


Figure 25: Association results.

The procedure that checks the associations quality recognized the first result as the best result. However, when the input image is different from database images the procedure for quality assessment identifies the association result as unsatisfying. The reason is the existence of noise in the picture. That is, the program identifies the contours of the face as noise. Thus recognition is poor when we use combined images. The best results are achieved for "ordinary" binary images.

In the next figure we can see binary images corresponding to combined images in Fig. 24.

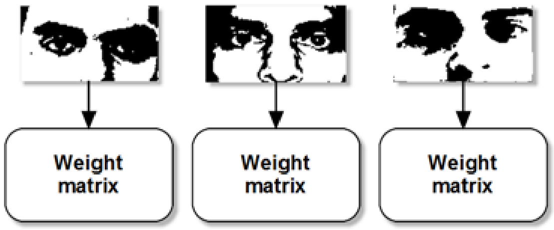


Figure 26: Memorized binary images in separate weight matrices.

Testing of this neural network was conducted with the new image as an input image (it does not exist in neural network's memory), see Fig. 27.

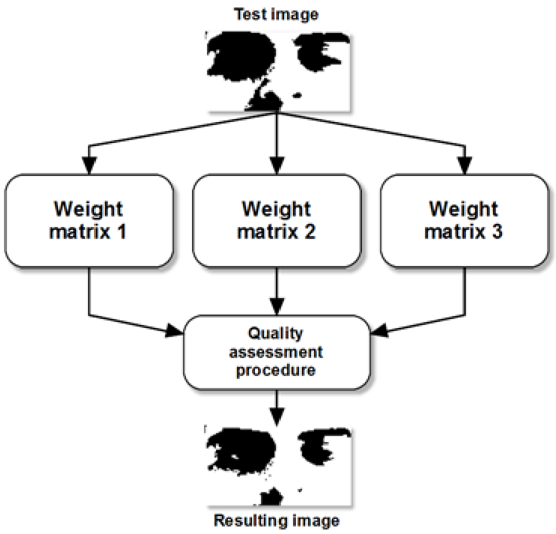


Figure 27: Result of association quality assessment procedure.

Indeed, this time the person on the input image is recognized as one of the persons from database. Comparing binary and combined images in general, we can say that the combined images really are more informative but are less efficient when used in Hopfield network.

After the last test, we can clearly see the advantage of binary images over combined images. This test also tells us how important is for facial picture to have minimum of details, since unnecessary details could lead to faulty recognition. The first reduction of details was performed by selecting only the eye region for analysis and the second one was when we converted gray image into black and white picture.

- Testing the tolerance of Hopfield network

Tolerance of Hopfield memory means successful recognition of face from the image even if the input image is significantly different than the memorized images.

Tolerance was tested for following conditions:

1. Light intensity.
2. Person's distance.
3. Head position.
4. Facial expression.

Since the conclusion is almost the same for each of these conditions, I will only show tests for the influence of light intensity on recognition quality. Original images and corresponding binary images are shown in the next figure.

Figure 28: Original colour pictures and corresponding binary images.

Let us look at the results when the first three images from Fig. 28 are memorized and the fourth image is test image (Fig. 29).

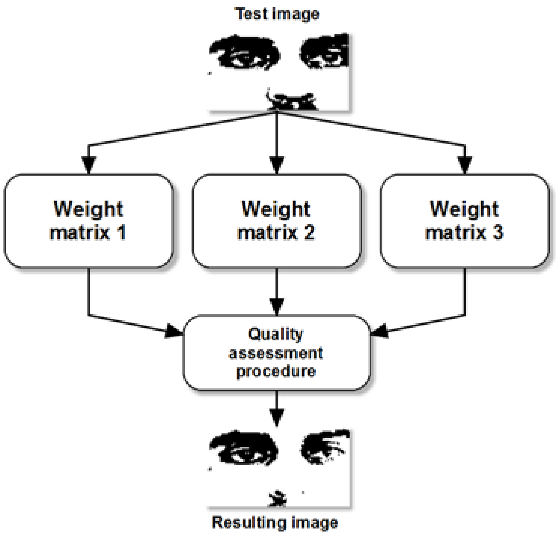


Figure 29: Result of association quality assessment procedure for images with different light intensity.

Note that, although the fourth picture for which the system is tested is significantly different from the three images that exist in the database, Hopfield network still successfully recognized that person from the picture.

Let me present another test with four different images on which the same person has different facial expressions. The result of this test is presented with real, colour images so that the results are more understandable.

Figure 30: Memorized images.

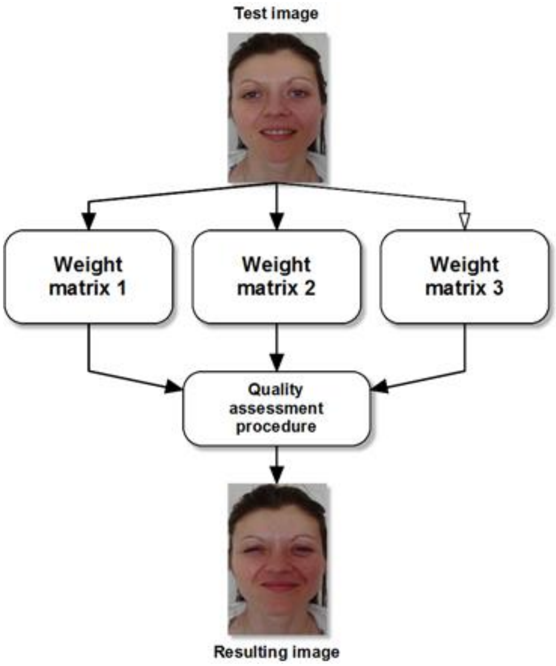


Figure 31: Example of Hopfield memory test.

- The crucial factors of successful recognition

Based upon the analysis, which is partially described in this paper, we can judge efficiency and possibilities of Hopfield networks regarding face recognition problem. For the effective face recognition it is essential to have exhaustive database, designed to encompass a variety of conditions. Properly prepared database has very important role and plays large part of success for face recognition. It is also crucial to have a proper Hopfield network structure which, unfortunately does not tolerate more than one image in one weight matrix.

We could see from several tests in this paper that Hopfield network is capable to recognize a person from the picture even though database is not prepared as recommended and input image is not properly illuminated. This testifies to the robustness of Hopfield networks.

CONCLUSION

In this paper I have tried to solve the problem of separation and recognition of faces from images using associative neural networks, in an efficient way. As we know, the problem consists of two main parts: image pre-processing and programming of neural networks. In both cases I tried to “soften” conditions under which the program gave good quality solutions.

As a result of the first part of problem solving, I wrote a procedure to detect and extract facial images by using silhouettes of the individuals in the picture. This procedure has proved to be extremely robust to light intensity and position of the person in the picture, and even to the type of image background. That is, detecting and extracting faces from the picture works flawlessly even in unforeseen circumstances. For the success of this part it was crucial to use histograms and Otsu’s method. Through the use of histogram equalization I provided approximately equal conditions for all processed images. Next step was removal of less important details with Otsu’s criteria, image filtration and with reduction of boundaries for region of interest.

For successful recognition, which I chose as the most influential factor for program quality, further conversion to black and white images was very important. I have tested various ways and forms of conversion, stratification and edge detection to find the best match for association of good quality. “Ordinary” black and white images, with minimum amount of information, proved to be the best choice for use with Hopfield network. To have the minimum amount of details demanded further reduction of images (e.g. reduction of region of interest). Tests showed that it is enough to use eye region for the successful recognition.

After the image processing I tested the different structures of Hopfield associative memory. The best results were given when the each image was memorized in separate weight matrix. Although it seems that the formation of additional weight matrices increases the execution time of program, it does not. Program execution time is the same as when only one weight matrix is used (we would still need to update weight matrix for each image).

Described image processing and Hopfield network programming resulted in a robust program that successfully detects people’s faces even when the tested picture of a person is significantly different from the pictures of the same person in the database (memory) of the neural network.

For the purpose of association performance testing, I wrote a separate function that compares input image and images stored in memory of neural network. If there is a picture in the memory that has less than $\pm 10\%$ of pixels different from the pixels in the associated image, we can call that association (recognition) successful.

Testing was performed with 60 different pictures photographed under different conditions. Some were photographed in daylight with the light source on the left or right side, some were captured at dusk using camera flash as a source of light. Distance of people in the pictures, their facial expressions and head positions also varied from picture to picture. So, photographing conditions were deliberately different in order to test the tolerance of Hopfield network. Nevertheless, Hopfield network managed to successfully recognize a person from the picture in all of these conditions.

If I would want to provide an ideal photographing conditions, and thus actually ensure the success of recognition, images would be captured with a light source centered in relation to a person’s position. For every person I would make a database so that it includes as many facial expressions and the positions of the head of one person. The larger the database for one person is, the higher probability of successful recognition is.

Note that it is necessary to for database only once so that program execution time is not in question.

However, this program is not written so that the face recognition can be conducted when the input image is arbitrary. The aim was to test whether the Hopfield's associative memory is able to solve the problem of this type and if yes, how effectively. In this paper I showed that associative neural network can indeed be used for this purpose.

REFERENCES

[1] Artificial Neural Networks – Colin Fyfe; Department of Computing and Information Systems, The University of Paisley (1996).
[2] Artificial Neural Networks in Real-Life Applications – Juan R. Rabunal, Julian Dorado; Idead Group Publishing (2006).
[3] Neural Networks: Algorithms, Applications, and Programming Techniques – James A. Freeman, David M. Skapura; Addison-Wesley Publishing Company (1991).
[4] A Neuro-Fuzzy Synergism to Intelligent Systems – Chin-Teng Lin, C.S. George Lee; Prentice Hall PTR (1996).
[5] Digital Image Processing, 2/E – Rafael C. Gonzalez, Richard E. Woods; Prentice Hall (2001).
[6] Digital Image Processing using MATLAB – Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins; Prentice Hall (2004).
[7] Essential MATLAB For Engineers and Scientists, 3/E – Brian Hahn, Daniel T. Valentine; Elsevier (2007).
[8] Graphics and GUIs with MATLAB, 3/E – Patrick Marchand, O. Thomas Holland; Chapman&Hall/CRC (2003).
[9] Machine Learning – Tom Mitchell; McGraw Hill (1997).
[10] Neural Network-Based Face Detection – Henry A. Rowley, Shumeet Baluja, Takeo Kanade; PAMI (1998).
[11] Face Recognition using Neural Networks with Multiple Combinations of Categories – T. Yahagi, H. Takano; International Journal of Electronics, Information and Communication Engineering, vol.J77-D-II, no.11, pp.2151-2159 (1994).
[12] Neural Networks for Pattern Recognition – C.M. Bishop; Oxford University Press (1995).