15

A Novel Approach to Using Color Information in Improving Face Recognition Systems Based on Multi-Layer Neural Networks

Khalid Youssef and Peng-Yung Woo
Northern Illinois University
USA

1. Introduction

Nowadays, machine-vision applications are acquiring more attention than ever due to the popularity of artificial intelligence in general which is growing bigger every day. But, although machines today are more intelligent than ever, artificial intelligence is still in its infancy. Advances in artificial intelligence promise to benefit vast numbers of applications. Some even go way beyond that to say that when artificial intelligence reaches a certain level of progress, it will be the key to the next economical revolution after the agricultural and industrial revolutions (Casti, 2008). In any case, machine-vision applications involving the human face are of major importance, since the face is the natural and most important interface used by humans. Many reasons lie behind the importance of the face as an interface. For starters, the face contains a set of features that uniquely identify each person more than any other part in the body. The face also contains main means of communications, some of which are obvious such as the eyes as image receptors and the lips as voice emitters, and some of which are less obvious such as the eye movement, the lip movement the color change in the skin, and face gestures. Basic applications involve face detection, face recognition and mood detection, and more advanced applications involve lip reading, basic temperature diagnoses, lye detection, etc. As the demand on more advanced and more robust applications increase, the conventional use of gray-scaled images in machine-vision applications in general, and specifically applications that involve the face is no longer sufficient. Color information is becoming a must.

It is surprising that until recent study demonstrated that color information makes contribution and enhances robustness in face recognition. The common belief was the contrary (Yip & Sinha, 2001). Thus, gray-scaled images were used to reduce processing cost (Inooka, et al., 1999; Nefian, 2002; Ma & Khorasani, 2004; Zheng, et al., 2006; Zuo, et al., 2006; Liu & Chen, 2007). Simply speaking, we know from nature that animals relying more on their vision as a means of survival tend to see in colors. For example, some birds are able to see a wider color spectrum than humans due to their need to locate and identify objects from very high distances. The truth is that due to its nature, color can be thought of as a natural efficiency trick that gives high definition accuracy with relatively little processing cost as will be shown later in this article. Up to a certain point in the past, a simple yes or no to a still image of a face with tolerable size and rotation restrictions was good enough for

Source: Recent Advances in Face Recognition, Book edited by: Kresimir Delac, Mislav Grgic and Marian Stewart Bartlett, ISBN 978-953-7619-34-3, pp. 236, December 2008, I-Tech, Vienna, Austria

face recognition applications. For such a requirement, gray-scaled images did the job pretty well. Some even claim to have achieved recognition rates of upto 99.3% under these circumstances (Ahmadi, 2003). On the other hand, the demand on achieving human like level of recognition is ever-increasing, which makes the requirements even tougher. To be able to achieve these requirements, it is only intuitive to investigate how humans are able to do this. The human decision process in face recognition does not rely on mere fixed features extracted from shape information such as the shape of the nose, eyes, and lips. Humans use a combination of these and more sophisticated features that are ignored by the conventional face recognition techniques that rely on gray-scaled images, features like movement patterns, gestures, eyes color, and skin color.

Since color information plays a major role in achieving more advanced vision applications, and since the main reason behind using gray-scaled images is to reduce the processing cost, our goal in this article is to introduce the use of color information in these applications with minimal processing cost. This will be achieved by demonstrating new techniques that utilize color information to enhance the performance of face recognition applications based on artificial Neural Networks (NN) with minimal processing cost through a new network architecture inspired by the biological human vision system. It will be shown that conventional training algorithms based on Back Propagation (BP) can be used to train the network, and the use of the Gradient Descent (GD) algorithm will be explained in details. Further more, the use of Genetic Algorithm (GA) to train the network, which is not based on back propagation will also be explained. Although the application discussed in this article is face recognition, the presented framework is applicable to other applications.

The rest of this article is organized as follows. Section 2 gives a glimpse on previous work related to this subject. The proposed approach and the data processing behind it are given in section 3. Two basic trainig methods are explained in section 4. Experimental results and some observations are demonstrated in section 5, and the article is concluded in section 6.

2. Related work

Until recently, very little work where color information is used in face recognition applications could be found in the literature. Fortunately, this subject is attracting the attention of more researchers and the number of publications on this subject increased significantly in the last few years. However, most of the work that has been done so far basically belongs to at least one of two groups. The first group does not fully utilize color information, while the second group make better use of color information but at the cost of processing efficiency. The following is an example of each case respectively.

One approach suggests using gray-scaled images with an addition of the skin color as a new feature (Marcal & Bengio, 2002). This approach enhances the accuracy of face recognition with little extra processing cost. A 30x40 gray-scale image is used, which gives an input vector of dimension 1200. The additional vector that represents the skin color feature is of dimension 96. Thus, the input vector is of a total dimension 1296. This approach is good from the processing cost point of view and gives a better performance over similar approaches that only use gray-scale images, but it does not make a full use of the color information of the images. Marcal & Bengio also mentioned in their paper that their method has a weak point due to the color similarity of hair and skin pixels, which brings up an uncertainty to the extracted feature vector.

Another approach suggests to use color channel encoding with non-negative matrix normalization (NMF) (Rajapakse, et al., 2004) where the red, green, and blue (RGB) color channels act as separate indexed data vectors representing each image. NMF is then used for color encoding. Although this method makes better utilization of color information, there is a big inherent processing cost due to the encoding and the excessive iterative matrix operations that includes matrix inversion. Thus, in this case the performance enhancement is at the cost of processing efficiency.
NNs have proved to be among the best tools in face recognition applications and are widely used in approaches based on gray-scaled images. The approach followed in this article is originally proposed by us in a paper published in the proceedings of ICNN'07 (Youssef & Woo, 2007). This approach permits the use of NNs with colored images in a way that makes optimal use of color information without extra processing cost when compared to similar approaches that use gray-scaled images. In this article the original approach is elaborated with more illustration and demonstration of new methods to train the NN.

3. Proposed approach

3.1 Data processing preliminaries

Before the proposed approach is discussed, an introduction to the data processing behind it should be given.
All visible colors are a combination of three main color components, i.e., the red, green, and blue. In the human biological vision system, images are preprocessed before they are sent to the cortex which is responsible for the perception of the image. The image processing starts at the retina of the eye, which is not merely a transducer that translates light into nerve signals. The retina also extracts useful data and ignores redundant data before propagating it to the next stages. The retina of each eye contains 125 million receptors, called rods and cones. Cones are responsible for color vision, and rods are responsible for dim light vision. Naturally, rods cannot attain detailed vision, and they can merely identify shapes. Cones, on the other hand, use color information and are responsible for detailed vision. There are three basic types of cones: red, green, and blue.

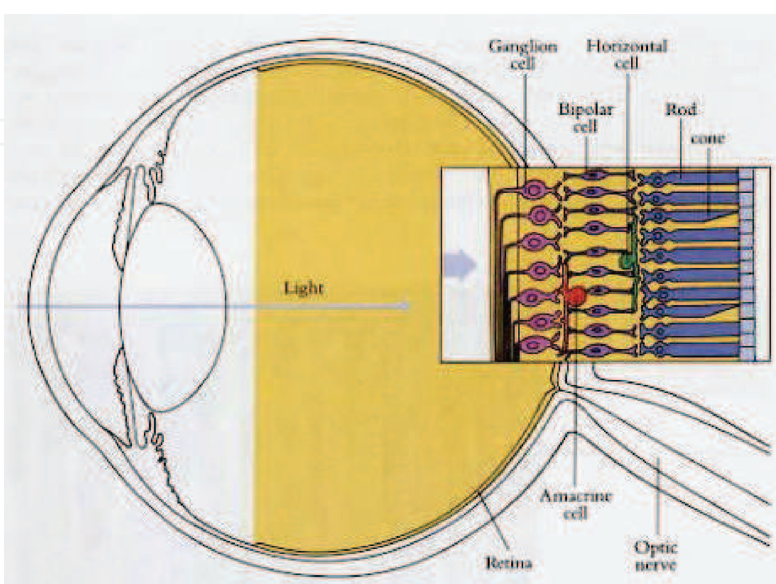


Fig. 1. Human Retina (Hubel, 1988)

In the RGB color system which is mostly used in computers, each of the color components is represented by a number ranging from 0 to 255 with 0 and 255 describing the absence and the full saturation of the color component, respectively. The combination of the different values of these components gives 256^3 = 16777216 different possible colors. A comparison between a face picture and its three RGB color components is shown in Fig. 2, where the top left picture is the original, the red component is on the top right, the green component on the bottom left and the blue component on the bottom right.

Fig. 2. RGB color components

Digital images are composed of pixels. The data contained in a pixel may vary depending on the pixel format. In general, this data carries information about the color, brightness, hue, and saturation of the pixel. Gray-scale images represent the luminance of the picture, and are usually achieved by extracting the luminance component from color spaces as YUV (Y is the luminance channel, and U and V are the color components), or by using a conversion method to convert pictures in RGB format into gray-scale images. One way to do this is to calculate the average of the red, green, and blue color components. In the method proposed in this article, the 24bit RGB format is used. The input data is divided into 4 channels, i.e., the red, green, and blue color channels, and the luminance channel which is attained by using the following formula:

$$\textit{Luminance} = 0.299\text{x}R + 0.587\text{x}G + 0.114\text{x}B \quad (1)$$

Each color commponent in Fig. 2 is composed of the shades of one color channel that vary between 0 and 255 and can be considered a gray-scaled image, since in RGB, the shades of gray can be obtained by setting the values of all the components to the same number, i.e. (0,0,0), (1,1,1,) and (255,255,255) correspond to different shades of gray.The shades of gray gets darker as the number increases such that (0,0,0) represents white (complete absence), and (255,255,255) represents black (full saturation).
Note the similarity among the three filtered images of the same picture in Fig. 2. This similarity is inspected further by plotting a graph for a series of consecutive pixels that have the same location in each of the filtered images as shown in Fig. 3. Each line in Fig. 3 corresponds to the graph of a color component. The dashed line, the solid line, and the dotted line correspond to the red, green and blue components respictively. In this case, the pixels are presented in a 1-D vector that is a mapping of the 2-D positions of the pixels such that the vertical axis represents the value of the component, and the horizontal axis represents the pixel‘s position in the image.

Fig. 3 shows that in real face images, although the individual values of color components for a certain pixel are different in general, the relations among the values of different color components share a certain degree of similarity. Also, it is seen that the average magnitude of the blue color component is less than those of the red and green, because the best overall image quality is given, for the range of wavelengths, between 500 and 700 nanometers on the visible-light spectrum, which corresponds to colors between red and green wavelengths, while far-red and far-blue wavelengths provide little resolution (Elliot, 1999).

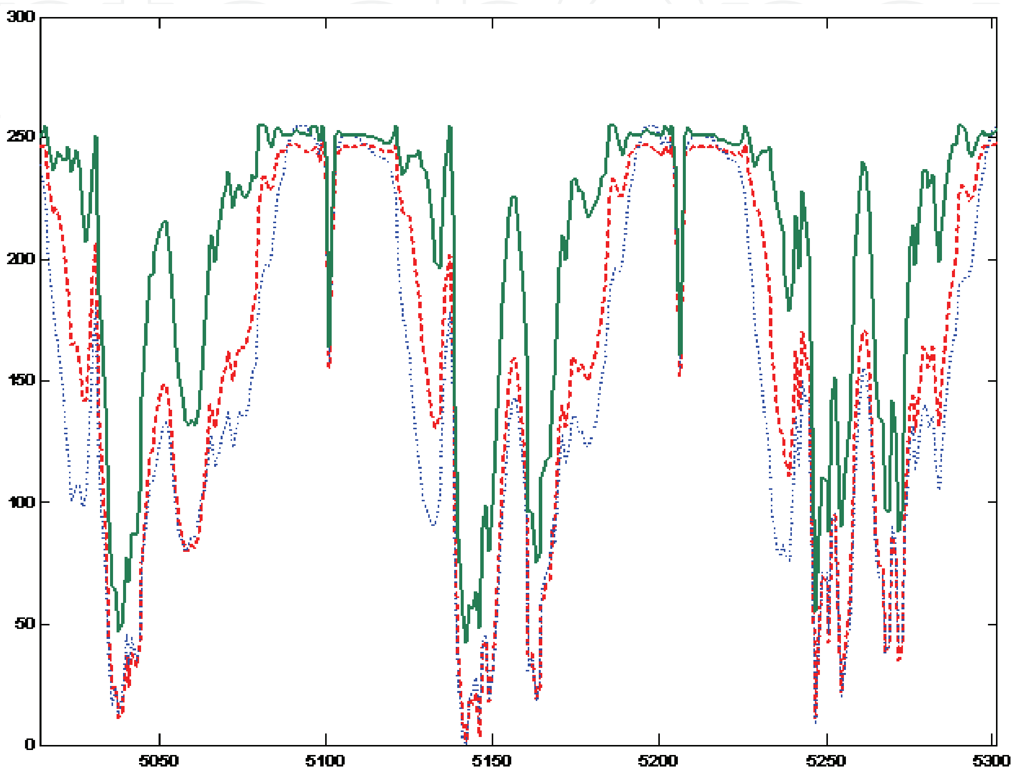


Fig. 3. 1-D pixel representation of color components

However, despite the similarity of the relation between pixel values of the color components, they are definitely not completely the same. In fact, the difference between the color components is where the extra information that is lost in gray-scaled images is embedded. Fig. 4 gives an idea of the difference between the color components. It shows the gray version of each indivdual component separately. The top left picture in Fig. 4 is a combination of all the components, the gray version of the red component is on the top right, the gray version of the green component on the bottom left and the gray version of the blue component on the bottom right.

In a complete face recognition application, face recognition is preceded by a face detection step that locates the face in the picture. Accurate face detection methods are studied by (Anifantis, et al., 1999; Curran, et al., 2005). However, in this article faces are manually cropped and resized to a face image of 19200 (160x120) pixels. The red, green, and blue color channels are extracted from the mage and the luminance channel is obtained by using Eq.(1), yielding four images that are different versions of the original picture. Each version of the image then goes through a process of smoothing and edge enhancement using convolution masks before it is mapped to a feature vector of length 19200. The images are also scaled such that all input values lie between zero and one.

Fig. 4. Gray version of the RGB color components

Convolution masks are used in image processing to detect the edges of objects in an image. Center-surround convolution masks are important in theories of biological vision. They implement an approximation to the Laplacian mathematical operator which is closely related to taking the second-order derivative of a function, and can be applied to digital images by calaculaing the difference between each pixel and the average of the pixels that surround it. Practical computer vision systems use masks that are related to the vertical and horizontal difference operations. Fig. 5 shows an example of a convoluted image and the detected edges of the image. On the left is the original picture, the convoluted image is shown in the middle and the edges after applying a threshold to the convoluted image is shown on the right. The edges are then enhanced.

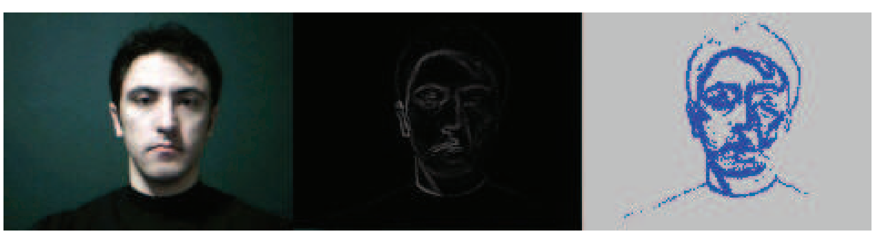


Fig. 5. Edge detection

3.2 Network architecture

Motivated by the biological vision system where the color information input is derived from the three cone types that represent the red, green, and blue colors, and due to the similar nature of the relation among the color components of the pixels in real pictures, we propose an overall structure of a multi layer neural network (MLNN) where the neurons in the input layer are divided into three groups, each of which is connected to a separate input vector that represents one of the three color channels. This modification allows us to make full use of color information without extra processing costs. The overall structure of the MLNN is shown in Fig. 6 where a conventional MLNN is shown on the left and the proposed MLNN is shown on the right. It can be seen that the inputs of the proposed MLNN in Fig. 6 are not connected to each neuron in the first hidden-layer. Thus, although the number of inputs used here is three times larger than the number of inputs used in the conventional methods where only the gray-scale image is processed, the number of connections is still the same and therefore the processing cost remains the same. As we mentioned before, each training picture in the experiments conducted in this article consists of 19200 pixels. In a standard MLNN that uses the gray-scale images, all the N neurons in the first hidden-layer are connected to the input vector representing the luminance channel and therefore there are 19200N connections. On the other hand, in the proposed method, the neurons in the first hidden-layer are divided into three groups with each consisting of $N/3$ neurons that are connected to one of the three RGB color channels. Thus, there are [19200*(N/3)]*3=19200N connections. It is seen that no extra processing cost is involved. The shades of gray using 32 and 256 levels are shown in the right and left sides of Fig. 7 respectively. In the proposed approach three color channels are used each of which has 256 levels giving a total number of 16777216 combinations.

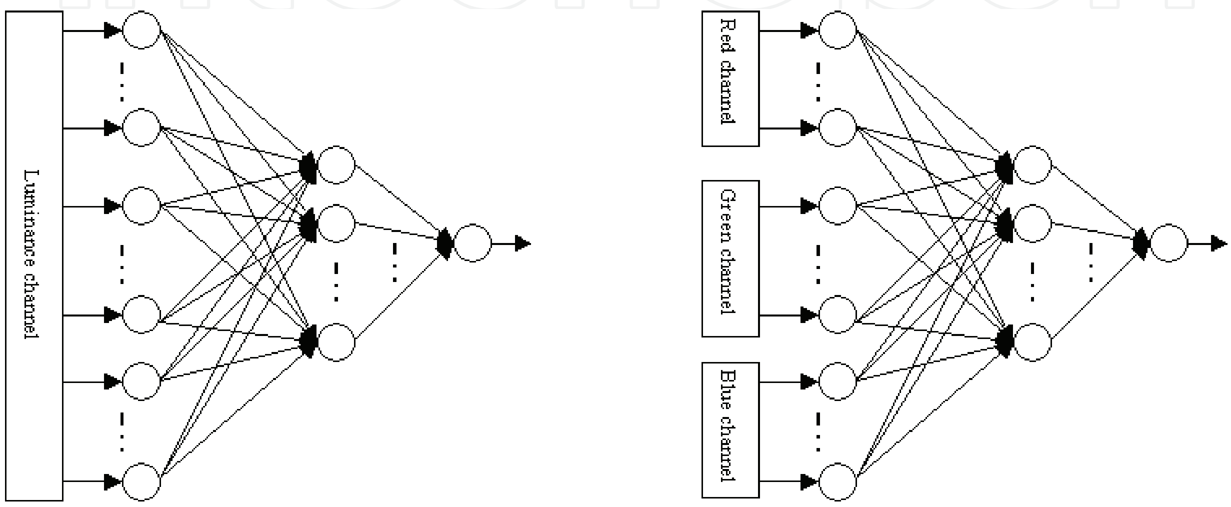


Fig. 6. Conventional v.s. proposed MLNN structures

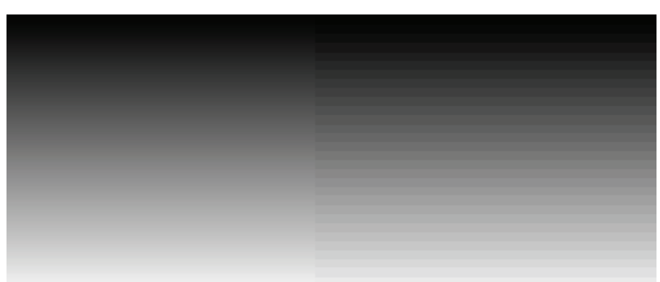


Fig. 7. Shades of gray

4. Training methods

4.1 Back propagation

In a standard multi-layer neural network, each neuron in any layer is connected to ***all*** the neurons in the previous layer. It can be represented as follows:

$$a_i^0 = P_i, i = 1,2,\dots,N^0 \tag{2}$$

$$n_i^{m+1} = \sum_{j=1}^{N^m} w_{i,j}^{m+1} a_j^m + b_i^{m+1} \quad ,m = 0,1,\dots,M-1, i = 1,2,\dots,N^{m+1} \tag{3}$$

$$a_i^{m+1} = f_i^{m+1}(n_i^{m+1}) \tag{4}$$

*w*here p is the input of the MLNN, n and a are the input and output of a certain neuron, respectively, w is the weight, b is the bias and f is the activation function. The superscript m and the subscripts i and j are the indexes for the layers and neurons, respectively.
The MLNNs are usually trained by using a supervised training algorithm based on the EBP a.k.a. back propagation algorithm (BP) that was developed by P.J. Werbos whose Ph.D. thesis "Beyond Regression" is recognized as the original source of back propagation (Werbos, 1994). The EBP algorithm was rederived independently by D.B. Parker. Parker also derived a second-order back propagation algorithm for adaptive networks that approximates the Newton's minimization technique (Parker, 1987). After a careful review of the derivations in Werbos' thesis (Werbos, 1974) and Parker's paper (Parker, 1987), we conclude that the algorithms they derived can also be applied to an "incomplete" MLNN where not each neuron in a layer is connected to all the neurons in the previous layer.
The following steps summarize the back propagation algorithm:

- Propagate the input forward through the network to the output.
- Propagate the partial derivatives of the error function backward through the network.
- Update the weights and biases of the network.
- Repeat until stop condition is reached.

In the case of our proposed architecture, the inputs are not fully connected as usually assumed by the algorithms. The weights corresponding to missing connections can be simply ignored in the updating step, and when included in the calculation of the error they can be considered of zero value.

4.2 Genetic algorithm

Genetic algorithms (GA) are derivative-free stochastic optimization methods based on the features of natural selection and biological evolution (Siddique & Tokhi, 2001). The use of GAs to train MLNNs was introduced for the first time by D. Whiteley (Whiteley, 1989). Whiteley proved later, in other publication, that GA can outperform BP (Whiteley, et al., 1990). A good comparison between the use of BP and GA for training MLNN is conducted by Siddique & Tokhi. Many studies on improving the performance and efficiency of GA in training MLNNs followed, and the use of GA was extended further to tune the structure of NNs (Leung, et al., 2003).
The GA is a perfect fit for training our system. The structure of the MLNN does not matter for the GA as long as the MLNN's parameters are mapped correctly to the genes of the chromosome the GA is optimizing. For large-scale MLNNs as in our case, the preferred type of encoding is value encoding. Basically, each gene represents the value of a certain weight or bias in the MLNN, and the chromosome is a vector that contains these values such that each weight or bias corresponds to a fixed position in the vector as shown in Fig. 8.
The fitness function can be assigned from the recognition error of the MLNN for the set of pictures used for training. The GA searches for parameter values that minimize the fitness function, thus the recognition error of the MLNN is reduced and the recognition rate is maximized. The GA in general takes more time to train the MLNN than BP, thus the processing cost is increased. However, the processing cost of the training phase of the MLNN does not have a big weight, since the training is performed only once. Once the network is trained, it goes to what is known as the feedforward mode which is independent of the training algorithm. The feedforward mode is what counts the most when it comes to processing cost, since that is what is used in practice. When a search engine is searching for

a face in a database that contains millions of faces, the processing cost of the feedforward mode should be minimized as much as possible.

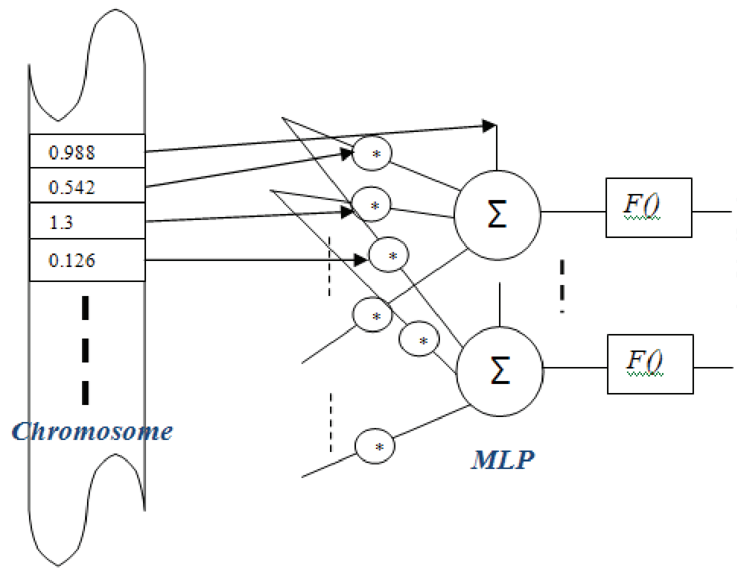


Fig. 8. Weight/bias mapping

5. Experimental results and observations

The modified MLNN used in the experiments consists of an input layer, two hidden-layers and an output layer. In our case each network is specialized in recognizing the face of one person, so there is only one neuron in the output layer. The network's output is a decimal number between 0 and 1. A threshold is then applied to the output to tune the network performance. A very high threshold enhances the false recognition rate (FRR), but at the same time decreases the true recognition rate (TRR). On the other hand, a very low threshold increases the TRR, but at the same time yields a higher FRR. The optimal threshold depends on the application requirement. In our experiments, the threshold is chosen in the middle point in order for the comparison between the performances of different systems to be fair.

The modified MLNN is trained to recognize the face of one person by using a number of pictures of the same person's face with different expressions, shooting angles, backgrounds and lighting conditions as well as a number of pictures of other people. Each MLNN is then tested by using pictures that have not been used in training. After the appropriate weights and biases that enable the system to recognize a certain person are found, they are saved in a file. This process is repeated for the faces of other people as well. The system can then be used to identify people that it has been previously introduced to. The system loads the files that it has in its memory, one file at a time, and performs the tests by using the data extracted from the picture to be identified and the weights and biases loaded from a certain file, keeping in mind that each file contains the weights and biases that correspond to a certain person. Note that the system now operates as a feedforward neural network, which simply calculates the output value. Dedicating a MLNN for each person makes it easier to expand to systems that search large numbers of faces, and it also simplifies adding new faces to the system.

The system is tested for a database of colored images of 50 people with 15 pictures for each including different expressions, shooting angles and lighting conditions. The pictures are obtained from the Georgia Tech face database. They are divided into two groups. The first group consists of 30 people, where nine pictures of each person are used for training the system, two pictures for validation and the remaining four pictures for testing. The second group consists of 20 people that are not used for training, but only for validation and testing. The same training and testing processes are also done for the standard MLNN by using gray-scale versions of the pictures. The success rates of the recognition tests without noise and with different noise levels for the standard and proposed MLNNs are recorded. Table 1 presents results of MLNNs trained using BP, and Table 2 presents results of MLNNs trained using GA.
It is clear from the results that the success rates of the recognition tests are higher for the proposed system. Furthermore, the difference gets more significant as the noise level increases. This demonstrates that our method is more robust to noise. It is not the objective of this article to find the optimal algorithm to train the MLNNs. Our objective is to demonstrate the superiority of the proposed system that operates with color pictures. Thus, a basic GA and the gradient descent algorithm are chosen for training, and a simple feature extraction technique is used in preprocessing. By using other GA and EBP algorithms and more advanced feature extraction techniques, the performance could be further enhanced.

Noise Mean Value	Color	Gray-scale
Without Noise	91.8%	89.1%
0.05	91.8%	89.1%
0.1	90.2%	88.4%
0.2	86.6%	81.3%

Table 1. Color v.s. gray-scale for BP

Noise Mean Value	Color	Gray-scale
Without Noise	94.3%	90.4%
0.05	94.1%	89.8%
0.1	91.6%	87.8%
0.2	84.7%	78.6%

Table 2. Color v.s. gray-scale for GA

Gray-scale methods basically use combinations of constant ratios of the RGB color channels to obtain the luminance channel. However, the ratios used do not necessarily correspond to the best distribution of the color channels. Furthermore, using constant ratios for color distribution does not dynamically adapt to the specific pictures in concern. On the other hand, in the system proposed in this article, the color distribution is determined by iteratively updating the weights that correspond to each color channel for the specific pictures in concern.
The inputs to the network are related directly to the pixels of the picture and therefore can be viewed as a 2-D grid of inputs, with each input giving a value of a certain color component for a certain pixel. Since each input has a weight related, the weights can also be viewed as a 2-D grid that corresponds to the 2-D grid of inputs. The obtained weights for one neuron in each of the three channels after training, between the input vector and the first hidden-layer, are put into three 2-D arrays. The obtained 2-D arrays are then plotted by

using the (surf) function in Matlab that produces a 3-D plot where the x-axis and the y-axis correspond to the position of the element in the 2-D array, and the z-axis corresponds to the value of that element. When the (surf) function is used with the default color-map, higher values are assigned shades of red colors, middle values are assigned shades of yellow colors, and lower values are assigned shades of blue colors. Fig. 9 demonstrate the x-y view of the progress of wiegts' values in training for three neurons corresponding to the green, red, and blue input vectors, respictively. The first row in Fig. 9 corrsponds to the value of the weights before the network is trained, where they are set to random values. The second row shows the weights' values after four epochs of training using the GD algorithm. The third row shows the weights' values after eight epochs, and the last row shows the weights' values after the desired small error is reached. The first column in Fig. 9 (left) corresponds to the weights of a neuron connected to an input vector in the green color channel, the middle column corresponds to the weights of a neuron connected to an input vector in the red color channel, and the last column (right) corresponds to the weights of a neuron connected to an input vector in the blue color channel.

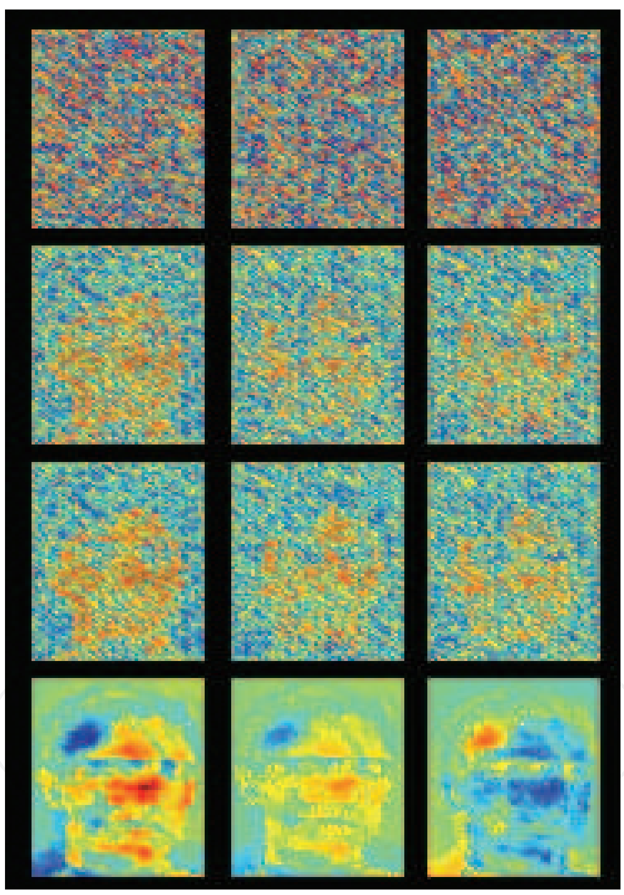


Fig. 9. Weights plot of neurons from the three color groups

The color distribution in the plots in Fig. 9 shows that neurons using the green channel input vector have the highest weight values at face features like the nose, the chin, and the forehead. Neurons using the red channel input vector have medium values, and neurons using the blue channel input vectors have the lowest values. This result agrees with what is depicted in Fig. 3.

6. Conclusion

Multi layer neural networks (MLNNs) have proved to be among the best existing techniques used in automatic face recognition due to their inherent robustness and generalizing ability. There are many papers in the literature that suggest different approaches in using neural networks to achieve better performance. The focus so far has been on studying different training algorithms and different feature extraction techniques. Yet, the majority of the work done in all face recognition methods in general and in methods that use neural networks specifically is based on grayscale face images. Until recent studies demonstrated that color information makes contribution and enhances robustness in face recognition. The common belief was the contrary. Grayscale images are used to save storage and processing costs. Colored images are usually composed of three components (Red, Green, Blue) while grayscale images are composed of one component only which is some form of averaging of the three color components, thus it basically requires one third of the cost. However, grayscale images only tell part of the story, and even though the enhancement that color adds might not seem very important to systems that use face recognition for basic identification applications, colors will be crucial for more advanced future applications where higher levels of abstraction is needed. There are many growing areas of computer vision in applications such as robotics, intelligent user interfaces, authentication in security systems and face search in video databases that are demanding color information important for future progress.

This article illustrates the importance of using color information in face recognition and introduces a new method for using color information in techniques based on MLNNs without adding mentionable processing cost. This method involves the new network architecture, and it can be used to enhance the performance of systems based on MLNNs regardless of the training algorithm or feature extraction technique. Motivated by how each pixel in a picture is comprised of its own unique color and the relation among those color components, we have proposed a new architecture. The new proposed network architecture involves the neurons in the input layer to be divided into three groups, each of which is connected to a separate input vector that represents one of the three color channels (Red, Green, Blue). This way, the modification allows us to make a full use of color information without extra processing costs. In addition, even though the input of the three color channels is larger than the standard input for the grayscale image, there is still the same number of connections, resulting in no further processing cost than the standard MLP. Experimental results that compare the performance of different approaches with and without using the proposed approach are demonstrated. Based on the aforementioned theoretical analysis and experimental results, the superiority of the proposed approach in face recognition is claimed, where the color distribution is determined by iteratively updating the weights that correspond to each color channel for the specific pictures in concern.

7. References

Ahmadi, M.; Faez, K. & Haddadnia, J. (2003). An Efficient Human Face Recognition System Using Pseudo-Zernike Moment Invariant and Radial Basis Function Neural

Network, *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 17, No. 1, Feb 2003, pp. 41-62, 0218-0014

Anifantis, D.; Dermatas, E. & Kokkinakis, G. (1999). A Neural Network Method For Accurate Face Detection on Arbitrary Images, *Proceedings of the International Conference on Electronics, Circuits, and Systems*, pp. 109-112, IEEE

Casti, J. (2008). Economics of the Singularity, *IEEE Spectrum*, Vol. 45, No.6, June 2008, pp. 45-50

Curran, K.; Li, X. & Caughley, N. (2005). The Use of Neural Networks In Real-time Face Detection, *Journal of Computer Sciences*, Jan 2005, pp 47-62, Science Publications

Elliott, C. (1999). Reducing Pixel Count Without Reducing Image Quality, *Information Display*, Vol. 15, December 1999, pp22-25

Hubel, D. (1988). *Eye, Brain, and Vision*, W H Freeman & Co, 978-0716750208, USA

Inooka, Y.; Fukumi, M. & Akamatsu, N. (1999). Learning and Analysis of Facial Expression Images Using A Five-Layered Hourglass-Type Neural Network, *Proceedings of the International Conference on Systems, Man, and Cybernetics*, pp 373-376, October 1999, IEEE

Leung, F.; Lam H.; Ling S. & Tam P. (2003). Tuning of the Structure and Parameters of a Neural Network Using an Improved Genetic Algorithm, *IEEE Transactions on Neural* Networks, Vol. 14, No. 1, Jan 2003, pp. 79-88

Liu, Y. & Chen, Y. (2007). Face Recognition Using Total Margin-Based Adaptive Fuzzy Support Vector Machines. *IEEE Transactions on Neural Networks*, Vol. 18, Jan 2007, pp. 178-192

Ma, L. & Khorasani, K. (2004). Facial Expression Recognition Using Constructive Feedforward Neural Networks, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, Vol. 34, No. 3, June 2004, pp. 1588-1595

Marcel, S. & Bengio, S. (2002). Improving Face Verification Using Skin Color Information, *Proceedings of the International Conference on Pattern Recognition*, pp 378-381, August 2002

Nefian, A. (2002). Embedded Bayesian Networks for Face Recognition, *Proceedings of the International Conference on Multimedia and Expo*, pp. 133-136, 0780373049

Parker, D. (1987). Optimal Algorithms for Adaptive Networks: Second Order Back Propagation, Second Order Direct Propagation, and Second Order Hebbian Learning, *Proceedings of the International Conference on Neural Networks*, pp. 593-600, IEEE

Rajapakse, M.; Tan, J. & Rajapakse, J. (2004). Color Channel Encoding with NMF for Face Recognition, *Proceedings of the International Conference on Image Processing*, pp 2007-2010, October 2004

Siddique, M. & Tokhi, M. (2001). Training Neural Networks: Back Propagation vs. Genetic Algorithms, *Proceedings of International Joint Conference on Neural Networks*, pp. 2673-2678, 9780780370449,USA, July 2001, IEEE, Washington D.C.

Werbos, P. (1974). *Beyond Regression: New Tools for Prediction and Analysis Behavioral*, PhD Thesis, Harvard Univirsity

Werbos, P. (1994). *The Roots of Back Propagation: From Ordered Derivatives to Neural Networks and Political Forecasting*, Wiley-IEEE, 0471598976

Whiteley, D. (1989). Applying Genetic Algorithms to Neural Networks Learning, *Proceedings of Conference of the Society of Artificial Intelligence and Simulation of Behavior,* pp. 137-144, England, Pitman Publishing, Sussex

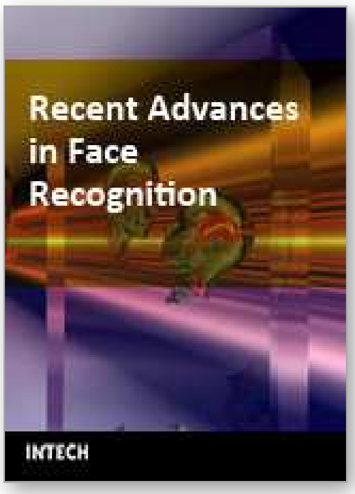
Whiteley, D.; Starkweather, T. & Bogart, C. (1990). Genetic Algorithms and Neural Networks: Optimizing Connection and Connectivity, *Parallel Computing*, Vol. 14, pp. 347-361

Yip, A. & Sinha, P. (2001). Role of Color in Face Recognition, *MIT AI Memos*, AIM-2001-035, (Dec 2001)

Youssef, K. & Woo, P.-Y. (2007). A New Method for Face Recognition Based on Color Information and a Neural Network, *Proceedings of the International Conference on Natural Computation*, pp. 585-589, 9780769528755, China, Aug 2007, IEEE, Haikou

Zheng, W.; Zho, X.; Zou, C. & Zhao, L. (2006). Facial Expression Recognition Using Kernel Canonical Correlation Analysis (KCCA), *IEEE Transactions On Neural Networks*, Vol. 17, No. 1, January 2006, pp 233-238

Zuo, W.; Zhang, D.; Yang, J. & Wang, K. (2006). BDPCA Plus LDA: A Novel Fast Feature Extraction Technique For Face Recognition, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, Vol. 36, No. 4, August 2006, pp. 946-953



Recent Advances in Face Recognition

Edited by Kresimir Delac, Mislav Grgic and Marian Stewart Bartlett

ISBN 978-953-7619-34-3
Hard cover, 236 pages
Publisher InTech
Published online 01, June, 2008
Published in print edition June, 2008

The main idea and the driver of further research in the area of face recognition are security applications and human-computer interaction. Face recognition represents an intuitive and non-intrusive method of recognizing people and this is why it became one of three identification methods used in e-passports and a biometric of choice for many other security applications. This goal of this book is to provide the reader with the most up to date research performed in automatic face recognition. The chapters presented use innovative approaches to deal with a wide variety of unsolved issues.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821