Comparison between Radial Basis and Backpropagation Neural Network in Face Detection

مقارنة بين الشبكة العصبية ذات الأساس الشعاعي وذات الأنتشار العكسي في كشف الوجه

Abdulbasit Kadhim Shukur

عبد الباسط كاظم شكر

Diyala University\College of Science\ Computer Science Dep.

جامعة ديالى / كلية العلوم / قسم علوم الحاسوب

Abstract

Computer vision is a computer science field belonging to artificial intelligence. The purpose of this branch is allowing computers to understand the physical world .This document proposes comparison between two types of an artificial neural network based face detection system .The proposed system consists of two subsystems ,first subsystem is face detection depending on radial basis neural network after some preprocessing operations, the second subsystem uses backpropagtion neural net to classify the image if it is face or not, also after some preprocessing operations .The final step is comparison between two subsystems in the final step.

المستخلص

عتبر مجال (computer vision) احد فروع الذكاء الصناعي ،الغاية من هذا الفرع هو تمكين الحاسوب من فهم العالم المادي. يقدم هذا البحث مقارنه بين نوعين من الشبكات العصبية لتحديد الوجه، يتكون النظام المقترح من نظامين فرعيين ،الأول يتضمن عملية تحديد الوجه معتمدا على شبكة القاعدة القطرية بعد مجموعة معالجك مسبقة ويتضمن النظام الفرعي الثاني عملية تحديد الوجه معتمدا على شبكة الانتشار الخلفي ،وتكون الخطوة النهائية بعملية المقارنة بين مخرجات الأنظمة الفرعية.

Keywords: Face localization, Face Detection, Face recognition, Expression recognition.

Comparison between Radial Basis and Backpropagation Neural Network in Face Detection

Abdulbasit Kadhim Shukur

Introduction

Computer vision is a computer science field belonging to artificial intelligence. The purpose of this branch is allowing computers to understand the physical world by visual media means[1]. Any camera (with sufficient resolution) can be used to obtain the image of face .Any scanned picture can be used as well. Generally speaking the better the image source (i.e. camera or scanner) the more accurate results we get .The lighting conditions required are mainly dependent on the quality of the camera used .In poor light condition, individual features may not be easily discernible. Face represents complex multidimensional meaningful visual stimuli and developing computational model for face recognition is difficult.[1]

Neural network

The ability to automatically learn from examples makes neural network approach attractive and exciting . Moreover ,it is well known that neural networks are very robust and adaptive. Therefore for the applications which undergo many variation factors such as biometrics systems (e.g., face ,palm ,fingerprint ...)neural networks seem to be a good remedy to solve the recognition problem.[1]

Radial Basis Neural Net in Face Detection

Radial basis neural networks techniques are very suitable for face detection ,instead of detecting a face by fallowing a set of human designed rules ,radial basis neural network learning the underlying rules from the given collection of representative examples. Since face detection can be treated as two class pattern recognition.The advantage of using neural network for face detection is the feasibility of training a system to capture the complex class conditional density of face patterns. [1] . An RBF neural network structure is shown in Fig.(3), which has architecture similar to that of a traditional three-layer feed forward neural network. The construction of the RBF neural network involves three different layers with feed forward architecture.

Architecture

Here is a radial basis network with R inputs.

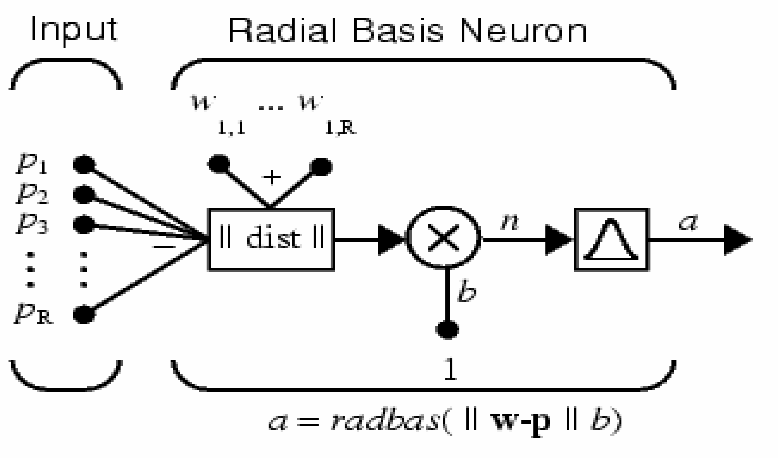


Figure (1)
Radial basis neuron

The expression for the net input of a radbas neuronin figure(1) is different from that of other neurons. Here the net input to the radbas transfer function is the vector distance between its weight vector w and the input vector p, multiplied by the bias b. (The || dist || box in thisfigure accepts the input vector p and the single row input weight matrix, and produces the dot product of the two.).

the transfer function for a radial basis neuron is

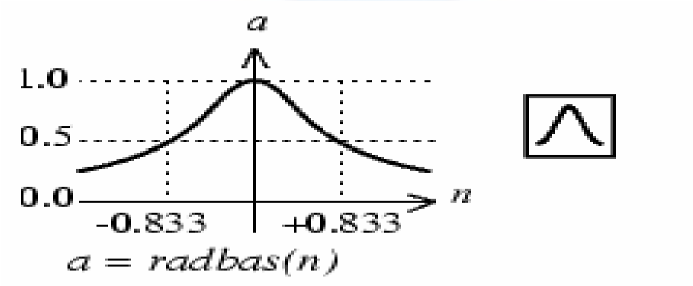


Figure (2)
Radial basis function

The radial basis function has a maximum of 1 when its input is 0. As the distance between w and p decreases, the output increases. Thus, a radial basis neuron acts as a detector that produces 1 whenever the input p is identical to its weight vector w .The bias b allows the sensitivity of the radbas neuron to be adjusted. For example, if a neuron had a bias of 0.1 it would output 0.5 for any input vector p at vector distance of 8.326 (0.8326/b) from its weight vector w.

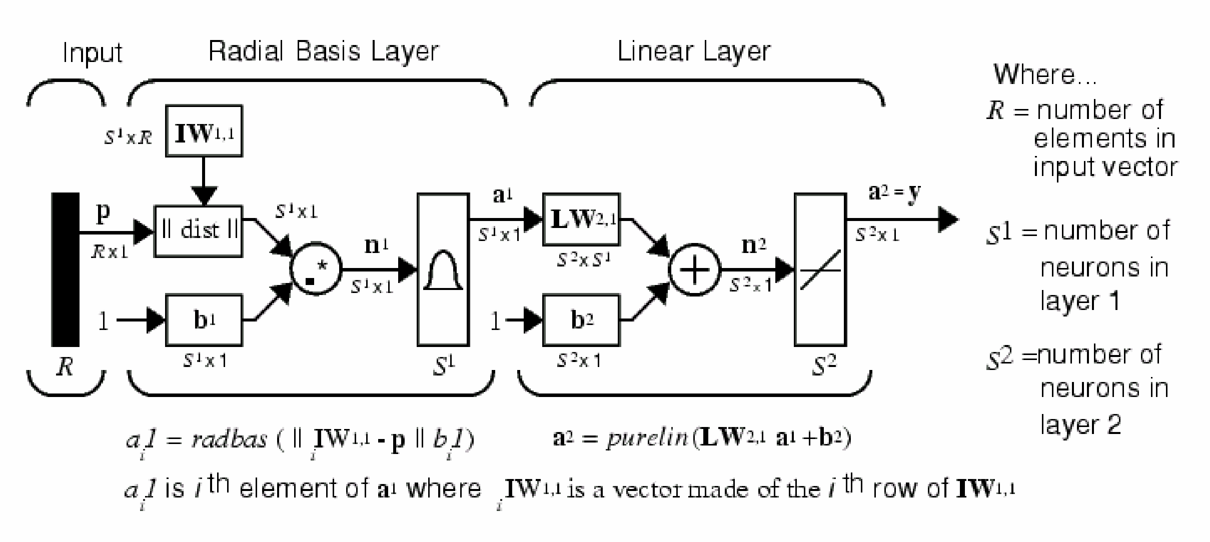


Figure (3)

Radial basis network

Radial basis network consist of two layers: a hidden radial basis layer of S1 neurons, and an output linear layer of S2 neurons.

IW1,1: Weight metrics to the first layer .

//DIST//: Euclidean distance weight function.

b1: basis.

LW2, 1: Weight metrics to the second layer.

The neural networks used in the proposed system are Radial Basis networks which consists of two layers .

Backpropagation in face detection

Backpropagation is the generalization of the Widrow-Hoff learning rule to multiple-layer networks figure(4)and nonlinear differentiable transfer functions . Input vectors and the corresponding target vectors are used to train a network until it can approximate a function, associate input vectors with specific output vectors, or classify input vectors in an appropriate way as defined by you. Networks with biases, a sigmoid layer, and a linear output layer are capable of approximating any function with a finite number of discontinuities.[4]

Architecture

An elementary neuron with R inputs is shown below. Each input is weighted with an appropriate w. The sum of the weighted inputs and the bias forms the input to the transfer function f. Neurons can use any differentiable transfer function f to generate their output.[3]

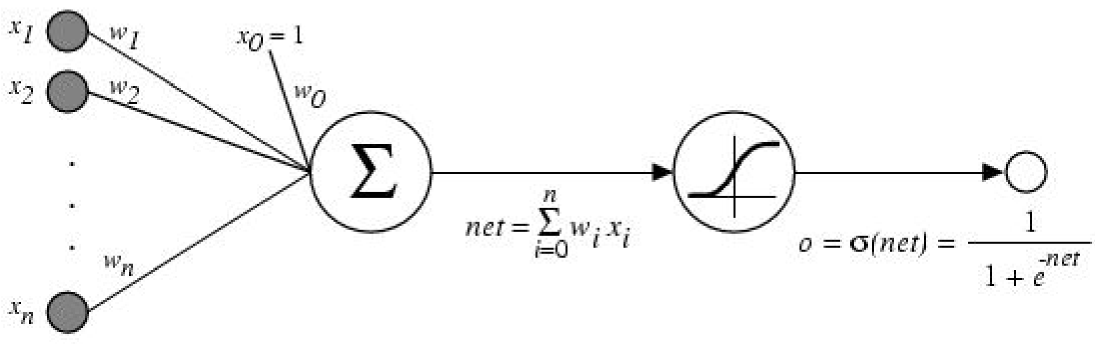
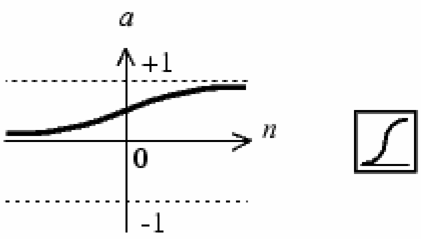


Figure (4)

An elementary neuron

Multilayer networks often use the log-sigmoid transfer function logsig.

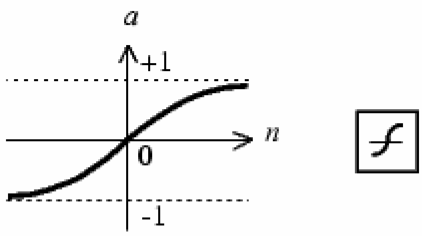


$a = logsig(n)$

Log-Sigmoid Transfer Function

The function logsig generates outputs between 0 and 1 as the neuron's net input goes from negative to positive infinity.

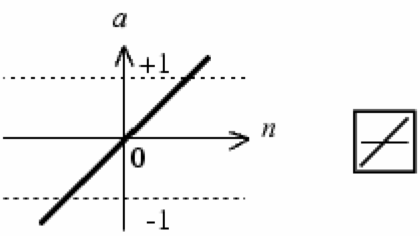
Alternatively, multilayer networks can use the tan-sigmoid transfer function tansig



$a = tansig(n)$

Tan-Sigmoid Transfer Function

Occasionally, the linear transfer function purelin is used in backpropagation networks.



$a = purelin(n)$

Feedforward networks often have one or more hidden layers of sigmoid neurons followed by an output layer of linear neurons. Multiple layers of neurons with nonlinear transfer functions allow the network to learn nonlinear and linear relationships between input and output vectors. The linear output layer lets the network produce values outside the range -1 to +1. for multiple-layer networks the number of layers determines the superscript on the weight matrices. The appropriate notation is used in the two-layer tansig/purelin network shown next.[3]

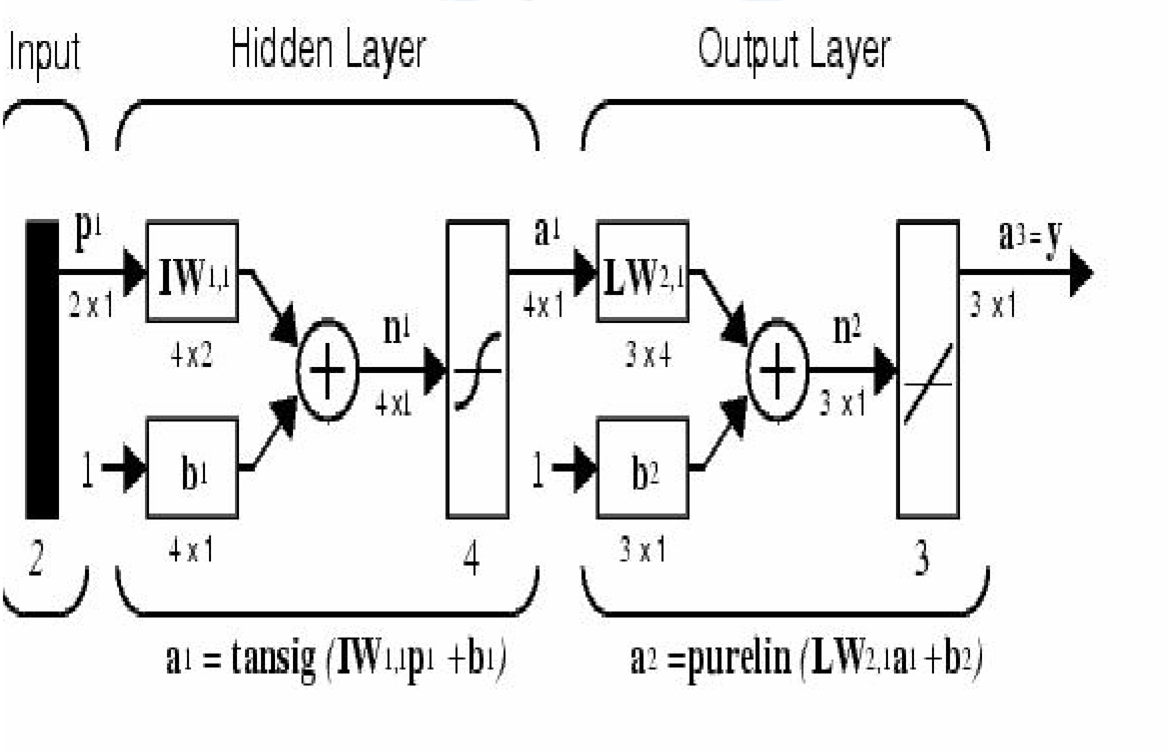


Figure (5)

Backpropagation Neural Network

The Proposed system

The aim of the proposed system is to a compare between two types of neural network in the face detection, in the other words ,the ability of neural (backpropagation and radial basis) to detect face pattern in image and compare the results .

The proposed face detection image system uses preprocessing stage because it is too difficult to use an image (in the neural network to training) without preprocessing operations ,the preprocessing operation make an image without noise or decrease the noise from the original colored input face image ,so it is necessary to use the further preprocessing . A preprocessing stage includes algorithms that concerned with the operations used to fined the object of interest(in the proposed system) to be prepared for the next levels .

Figure (6) represent the diagram of the proposed system, it contains the following operations

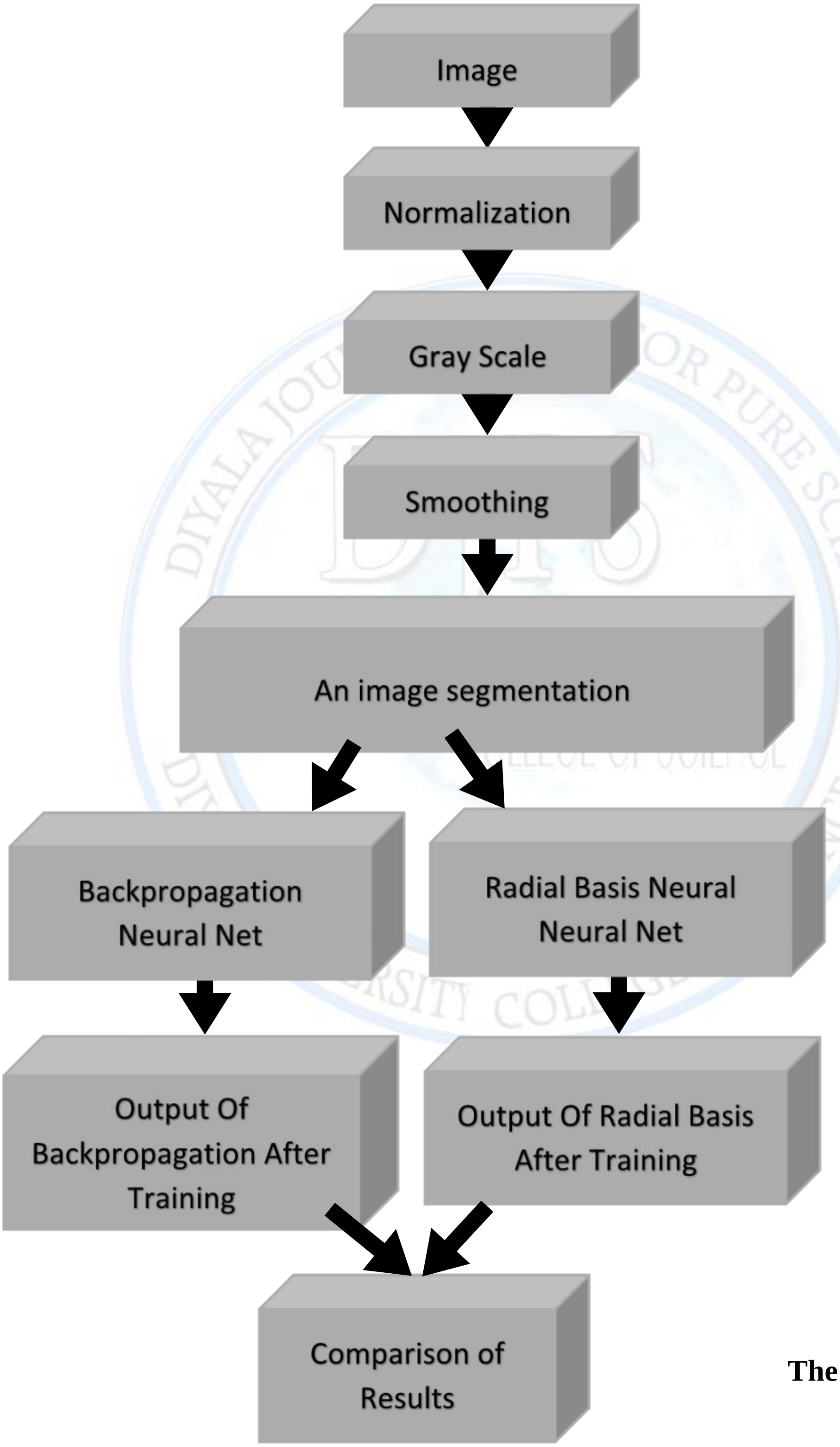


Fig (6)

The proposed system

Reading Image

The first function of the proposed system is reading the image ,the user can read image of file format JBEG, from any location in PC .

Normalization

In this operation ,gray scale image of face that entered to the proposed system normalized to (64*64 pixel) ,as described in algorithm (2) .

ALGORITHEM 2: normalization algorithm

The Normalization algorithm is as fallow

```
Begin
function oim=reduc(im,p1)
nr=p1(1);nc=p1(2);or=64;oc=64;
[or oc]=size(im);
fr=mod(or,nr); sr=fix(or/fr);
fc=mod(oc,nc); sc=fix(oc/fc);
j=0;k=0;
for i=1:or
   if i<=fr*sr
      if mod(i,sr)~=0
         j=j+1;
         im1(j,:)=im(i,:);
      end
   else
      j=j+1;
      im1(j,:)=im(i,:);
   end
end
j=0;k=0;
```

```
for i=1:oc
   if i<=fc*sc
      if mod(i,sc)~=0
         j=j+1;
         nim(:,j)=im1(:,i);
      end
   else
      j=j+1;
      nim(:,j)=im1(:,i);
   end
end
[r c]=size(nim);
str=r/nr;
stc=c/nc;
if str>1
om1=nim(1:str:end,:);
om2=nim(2:str:end,:);
om3=(double(om1)+double(om2))/2;
om=om3;
end
if stc>1
om4=om3(:,1:stc:end);
om5=om3(:,2:stc:end);
om6=(om4+om5)/2;
om=om3;
end;
oim=uint8(om);
end
```

Transform Color Image To Gray level

In this operation ,the colored image of face that entered to the proposed system will transform to gray level .each colored input image must convert to gray scale image,in other words (this operation will convert 24-bit/pixel images to 256 gray scale image,as described in the algorithmand shown in figure (7) .

Figure (7)

a- color image **b- converted to gray scale**

ALGORITHEM 1:conversion algorithm

The conversion algorithm is as fallow :

Begin

Step 1: load image file from memory

Step 2: For i=0 to image width

For j=0 to image width

Split the components (red ,green and blue) of pixel[i,j] as follows :

C= pixel [i,j];

Red[i,j]= (C and (logical and) HFF),

C= (C – Red[i,j])/256 ;

Green [i,j]=(C and (logical and) HFF);

C= (C – green[i,j])/256 ;

Blue [i,j]=(C and (logical and) HFF);

r= Red[i,j]/ (Red[i,j]+ Green [i,j]+Blue[i,j]);

g= Green[i,j]/ (Red[i,j]+ Green [i,j]+Blue[i,j]);

b= Blue[i,j]/ (Red[i,j]+ Green[i,j]+Blue [i,j]);

gray[i,j]= (Red[i,j]*r+ Green[i,j]*g +Blue [i,j]*b);

Draw the gray [i,j];

Step 3: Save the data of image;

End

Where:

H: hexadecimal.

F: value in hexadecimal=1111.

r : red value .

g :green value.

b:blue value .

Smoothing Operation

Image smoothing is used to give an image a softer or special effect to eliminate noise. Image smoothing is accomplished in the spatial domain by considering a pixel and its neighbor and eliminating any extreme value in this group .

These smoothing filters typically operate on small neighborhoods, (3*3) to (11*11). The two categories of smoothing filter for noise removal are order filters and mean filters. The order filters are implemented by arranging the neighborhood pixels in order from smallest to largest and using this ordering to select the "correct " value, while the filter determine the average value.

Median filter

One of the principle difficulties of the smoothing method is that it blurs edges and other harp details. If the objective is to achieve noise reduction rather than blurring, an alternative approach is to use median filter. That is, each image pixel is replaced by the median of the neighborhood of the pixel, instead of by the average. As indicated earlier, median filter is nonlinear filter .

For example, suppose that a (3*3) neighborhood has values (2,5,4,4,4,4,4,3,3). These values are stored as (2,3,3,4,4,4,4,4,5), which results in Meiden of 4 .

Mean filter

The mean filter function by finding some from an average within the (N*N) windows. The most basic of these filters is arithmetic mean filter, which fined the arithmetic average of the pixel values in the window .

The arithmetic mean filters smoothes out local variations within an image, so it is essentially a low pass filter. It can be implemented with a convolution mask . This filter will tend to blur an image while mitigating the noise effects .

For a (3*3) mean filter, the simplest arrangement would be a mask in which all coefficients have value of 1. However, the response would then be sum of nine pixels to result R, which could cause R to be out of the valid range. The solution is to scale the sum by dividing R by 9.

The various types of mean filters are most effective with different types of noise.

In the proposed system, the median filter used in smoothing operation, That is, each image pixel is replaced by the median of the neighborhood of the pixel, instead of by the average. As indicated earlier, median filter is nonlinear filter [2] because the mean filter depend on the blurring idea to remove noise. The blurring will remove the unneeded details from image.

ALGORITHEM 3: smoothing algorithm
The smoothing algorithm is as follow

```
    Begin
Step 1: load image from memory
Step 2: For i= 2 to image width-1
              For j= 2 to image height–1
                K=0
                For r= i-1 to i+1
```

```
            For c= j-1 to j+1
          Begin
              K=K+1;
             Buf [k]=Iimage[R,C];
           End
For f=1 to k-1
For h= f+1 to k
If buf[f]□ buf[h]
then
   begin
        x= buf[f];
        buf[f]= buf[h];
       buf[h]=x;
   end
     smooth [i,j]= buf[5];
Step 3: Save the result
Step 4: show the resulted image
End
```

Neural network stage

The following operation applied before / to both types of neural networks .to get the input vectors ,we should divide each image in the training image set to 2 dimensional blokes with same size .this operation called segmentation operation .we convert each block to one dimensional vector as follows :

We assume each block in an image has

u*v :size of block

g(p,q):gray level value for each coordinate (point)in the block at (p,q) position, now we represent each block as vector as follows

$\bar{x}$ =(x1 x2………. xi,………….. xn) where

xi : g(p,q)

n=u.v

i: vector coordinate

$\bar{x}$: input vector

to convert from position coordinate (p,q) to vector coordinate (i) do the following operation

i =(s-l)u+p…………(1) where

p= 1,2………u

s=1,2……..…v

to training set we do segmentation operation .the segmentation operation create a GM(training set) array with two dimensions(n*l),where

GM= [$\bar{x}$1 $\bar{x}$2 $\bar{x}$3…………$\bar{x}$k ………..$\bar{x}$l] where

$\bar{x}$t = { x1 x2………. xi,………….. xn } ,column vector

no of block in each image (f)=(N*N)/n

N*N: an image dimensions

n=u.v

l=no of training images *f

each column in TS array represents input vector to both neural nets .in each iteration we apply one column in GM array. select of vector (optionally) randomly or sequentially from first to last vector in GM array .

Implementation Level

It can be supposed that the union dataset of face-like and no face-like patterns is a nonlinearly separable set, so a non-linear discriminator function should be used. Artificial neural networks in general, and a multilayer feed forward learning rule in particular fit this role. The classifier training process consists of a supervised training. The patterns and the desired output for each pattern are showed to the classifier sequentially. It processes the input pattern and produces an output. If the output is not equal to the desired one, the internal weights that contributed negatively to the output are changed by the learning rule; it is based on a partial derivates equation where each weight is changed proportionally to its weight in the final output. In this way, the classifier can adapt it neural connections to improve its accuracy from the initial state (random weights) to a final state. In this final state the classifier should be able to produce correct (or almost correct) outputs. The network performance is measured by the Mean Squared Error (MSE). MSE is the sum of the squared absolute values of the difference between network outputs and desired outputs. Two neural network (radial and backpropagation) trained by 100 class to represent two classes(face and non) based on the teacher information because the training with supervisor used. so 100 class(face and non face)used in the test .7

Results

The following sections summarize the results of the detection to the both neural net

4 Experiment one : table 1and 2 show the ability of both neural (backpropagation and radial basis)to accommodate complex decision region .tens patterns are used in each experiment.

Table 1

Face no	Face	Target	Out of backpropagation
1		0.8976891	9.0912000
2		0.8976891	-14.6471000
3		0.9241793	11.0946000
4		0.97629813	3.79210000
5		0.951756900	2.22420000
6		0.894517451	-2.460700000
7		0.915719012	-5.282200000
8		0.895166734	9.297300000
9		0.889321853	-1.89600000
10		0.996341	2.040600

Table 2

Face no	Face	Target	Out of Radial basis
1		0.8976891	0.89770000
2		0.8976891	0.89770000
3		0.9241793	0.89770000
4		0.97629813	0.32420000
5		0.951756900	0.97630000
6		0.894517451	0.95180000
7		0.915719012	0.89450000
8		0.895166734	0.91570000
9		0.889321853	0.89520000
10		0.996341	0.88930000

5 Experiment two: table 3 show the results of both neural (backpropagation and radial basis).

Tabel3

Face no.	Face	Target	Out of backpropagation
11		0.996341	3.5396
12		0.888943231	1.0062
13		0.894596543	-11.7246
14		0.8867125	-16.9973
15		0.891321	-1.4292
16		0.93979211	14.1399
17		0.98715011	17.2682
18		0.86891429187	6.2797
19		0.8794567	-7.8061
20		0.974532445	23.2743

Face no	Face	Target	Out of Radial basis
11		0.996341	0.89770000
12		0.888943231	0.89770000
13		0.894596543	0.8946
14		0.8867125	0.8867
15		0.891321	0.8913
16		0.93979211	0.9398
17		0.98715011	0.9872
18		0.86891429187	0.8689
19		0.8794567	0.8795
20		0.974532445	0.9745

Conclusion

1- The experiments appears that the error rate is very high in the backpropagation as compared with radial basis .

2- The training time in the radial basis is less than backpropagation .

3- The performance of radial basis is higher than backpropagation

4- Radial basis networks can require more neurons than standard feed forward backpropagation networks, but often they can be designed in a fraction of the time it takes to train standard feed forward networks. They work best when many training vectors are available.

Reference

1. David Perera1,"Neural& Adaptative Computation " Computational Neuroscience Research Lab.Dept. of Computer Science & Systems, Institute for Cybernetics.
2. Chen, S., C.F.N. Cowan, and P.M. Grant, "Orthogonal Least Squares Learning Algorithm for Radial Basis Function Networks," IEEE Transactions on Neural Networks, Vol. 2, No. 2, March 1991, pp. 302-309.
3. Hagan, H.B. Demuth, and M.H. Beale, "Neural Network Design" ISBN 0-9717321-0-8 (available from John Stovall, john.stovall@colorado.edu.
4. D.E Rumelhart, G.E. Hinton, and R.J. Williams, "Learning internal representations by error propagation," D.E. Rumelhart and J. McClelland, editors, Parallel Data Processing, Vol.1, Chapter 8, The M.I.T. Press, Cambridge, MA, 1986, pp. 318-362.
5. Abdul Ridha " Face Detection ,Localization and Recognition using Neural Network" ,Thesis ,Baghdad University.2000.

6- شفاء عبد الرحمن "التمييز غير المتاثر للصور باستخدام الشبكات العصبية "،رسالة ماجستير،جامعة الموصل،تموز، 2000.