

A Study Regarding the Generalization Capacity of Image Classification by Using Neuroal Networks In Matlab

Al Rubaie Evan Madhi Hamzh
University of Babylon, College of Engineering
haidar_iraq77@yahoo.com

Abstract

The paper performs an algorithmic and experimental study regarding the generalization capacity of the scheme based on neuronal networks for the recognition of new images of the face. This enables both a rendering of graphic representations and the classification of image classes in Matlab. The purpose is to describe the recognition algorithm, to project and implement an application which proposes both the graphic representation of the images used by the neuronal training algorithm but also the implementation of the perceptron neuronal algorithm and the determination of the generalization capacity of the separating hyper plane of the considered image classes.

Keywords: web design, web application, web programming, software methodologies

الخلاصة

الغرض من البحث هو تنفيذ دراسة حسابية وتجريبية بشأن تعميم قدرة الخطة المبنية على الشبكات العصبية للتعرف على الصور الجديدة للوجه. وهذا يمكن من معالجة بيانات وتصنيف الصورة في لغة الماتلاب (Matlab). والغرض من ذلك هو لوصف خوارزمية التعرف، لعرض وتنفيذ الطلب الذي يقترح كل من التمثيل البياني للصور المستخدمة في تدريب خوارزمية الخلايا العصبية وأيضاً تنفيذ المستقبلات العصبية للخوارزمية وتحديد قدرة تعميم الفصل الفائق لطبقات الصور المعنية.

الكلمات المفتاحية: تصميم مواقع الانترنت، تطبيقات مواقع الانترنت، برمجة مواقع الانترنت، منهجيات البرنامج.

1. Introduction

MATLAB is a program package dedicated to numerical calculus and graphic representations. The basic elements with which it operates is the matrix, from which it takes its name: MATrixLABoratory. Its calculus and graphic representation resources are rich, allowing basic mathematic operations, data analysis, programming, 2D and 3D graphic representations, the creation of graphic interfaces etc (Arnold& Wagner,2007) From the point of view of its construction, MATLAB is made up of a basic nucleus around which the TOOLBOXES are grouped. These are specific applications, extended collections of MATLAB functions which develop the programming environment from one version to another, in order to solve different problems.

The full version of the MATLAB program package contains an entire family of specific modules, named toolboxes, respectively block sets, (Blaga,2002) which enable us to solve some applications in different fields such as: machines, devices and electric operations, system control, DSP applications, material processing and electro-technologies, signal processing, mechanics, aircraft and car industry, statistics, finances etc.

These modules are collections of MATLAB functions (M-files), easy to assimilate, which extend the computing power of the MATLAB program package in order to solve some particular classes of problems. The MATLAB collection of modules contains: (Ghinea & Firețeanu, 2003) Simulink, DSP, Control System, SimPowerSystems, SimMechanics, Data Acquisition, Fuzzy Logic, Image Processing, Partial Different Equations, Neural Network, Optimization, System Identification, Financial, Statistics, Communications, Database, Virtual Reality etc. (MATLAB, 2000).

The algorithmic scheme for image classification using neuronal architectures in Matlab.

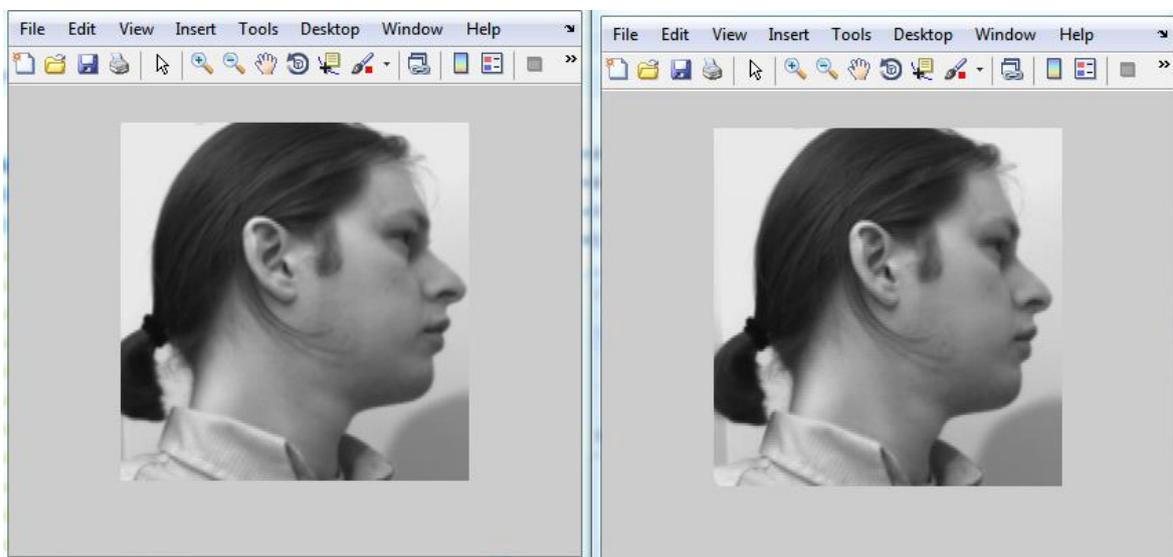
This part of the paper deals with the classification of the images coming from two different classes of data. (Moler, 2004) The classes are represented by persons, which enables us to consider a series of face images for the two classes which, after being processed, will determine the accuracy percentages regarding the recognition of new images of the considered persons. (The MathWorks)

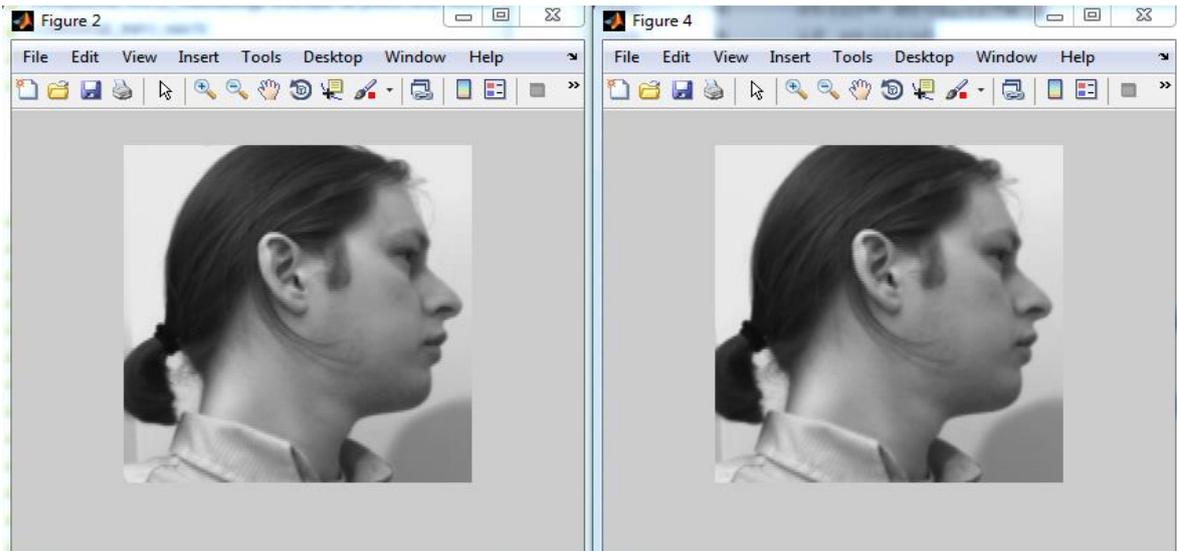
The work stages regarding the design of the application comprise the following stages: (Iatan, 2009)

1. the graphic representation of the considered images belonging to the two classes of images corresponding to the two persons
2. processing the images derived from the two classes by means of a neuronal training algorithm based on perceptron architecture
3. testing the quality of the separating hyperplane obtained under the previous step by recognizing the images used for the training
4. testing the generalization capacity of the separating hyperplane determined under step 2 by testing new images of the faces derived from the two considered persons.
5. the graphic representation of test images and establishing the image recognition percentages.

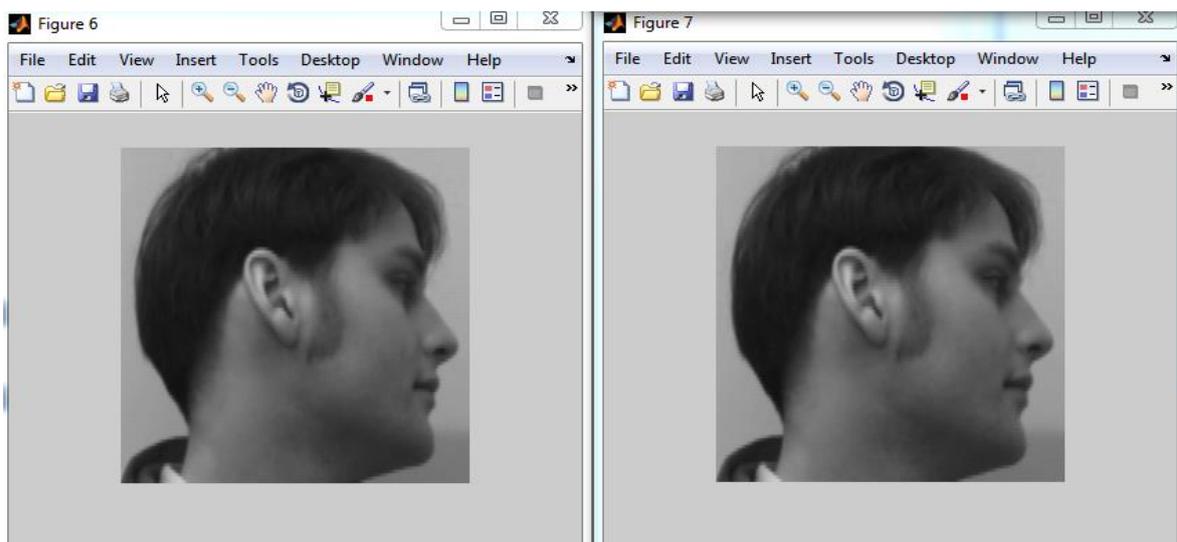
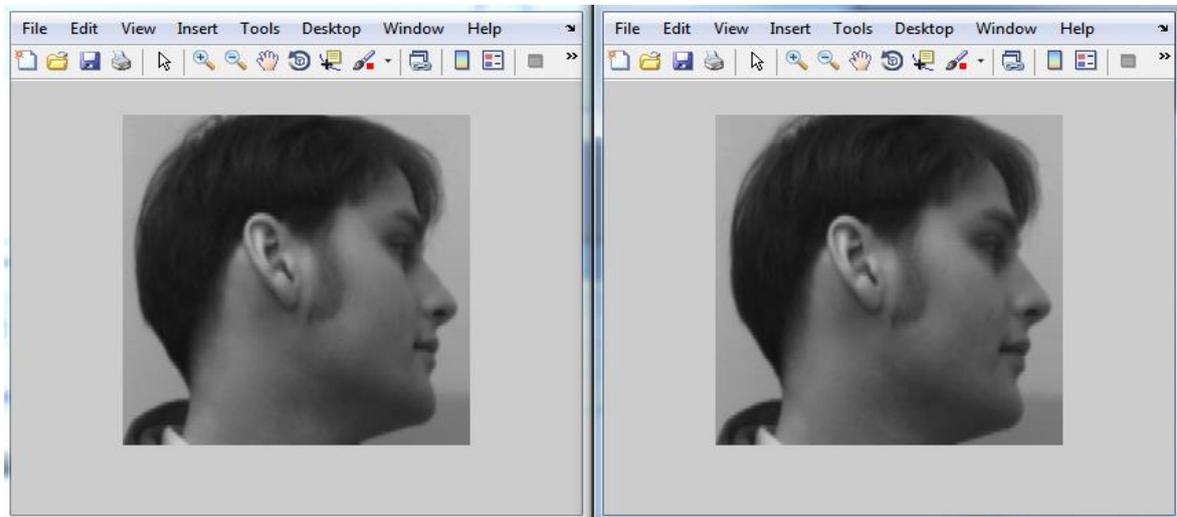
Stage 1. The graphic representation of the images corresponding to the two considered persons

Training images derived from class I :





Training images derived from class 2:



Stage 2. Processing the images derived from the two classes through a neuronal training algorithm based on perceptron architecture

This stage involves loading the images belonging to the two persons, representing them numerically through two-dimensional matrixes with numeric values that correspond to nuances after reading the component pixels. (DoruConstantin, Emilia Clipici, 2016).

In order to determine the separating hyperplane of the images belonging to the two persons, we must apply a pre-processing of the numeric images consisting in the linearization of each matrix of the loaded images, adding a value equal to 1 to each image vector, the image vectors belonging to the second class are multiplied by -1 and then we implement the neuronal training algorithm corresponding to a simple perceptron architecture.(DoruConstantin, 2008) This training algorithm involves determining the equation of the separating hyperplane (vector w) by running through the initial images in various stages and updating it each time the classification is inaccurate (the condition $Z(i,:) * w' \leq 0$) is fulfilled)

```

% The normalization of second class images
for i=1:size(Z,1)
if Y(i)==-1
    Z(i,:)=-Z(i,:);
“end
end

% The initialization of the separating hyperplane
w=zeros(1, size(Z,2));
out=false;
maximum number of runs =100;
nr=0;

% Determining the separating hyperplane
while (~out) & (nr<=maximum number of runs )
    nr=nr+1;
    out=true;
for i=1:size(Z,1)
if Z(i,:)*w'<=0
    w=w+Z(i,:);
    out=false;

```

```
end
end
end”
```

Stage 3 Testing the quality of the separating hyper plane obtained under the previous step through the recognition of the images used for the training.

The quality of the classification of the images used for training is determined by displaying the accurate percentage of recognition of the training data. (Doru Constantin,2009).

```
“r=[];
%YH hyperplane requests
for i=1:size(S,1)
    r(i)= S(i,:)*w';
    if r(i)>0
        YH(i)=1;
    else
        YH(i)=-1;
    end;
end;”

“percentage=0;
for i=1:size(S,1)
    if Y(i)==YH(i)
        percentage=prcentage+1;
    end;
end;

percentage=percentage/size(S,1)*100”
```

Stage 4: Testing the generalization capacity of the separating hyperplane determined under step two by testing new images of the faces derived from the two considered persons

```
“%% Testing new images for the persons in question
```

```

% Loading the test images from the 2 classes
imgdir = 'Cale__\Face2';
cd (imgdir);
r = dir;
t = r(3).name;
I1=imread(t);
dx=size(I1,1);
dy=size(I1,2);

C = [];
for i = 3:(size(r,1))
    t = r(i).name;

    I1=imread(t);

    I = imresize(I1,[dx dy])
    imshow(I);if i<(size(r,1)) figure; end;

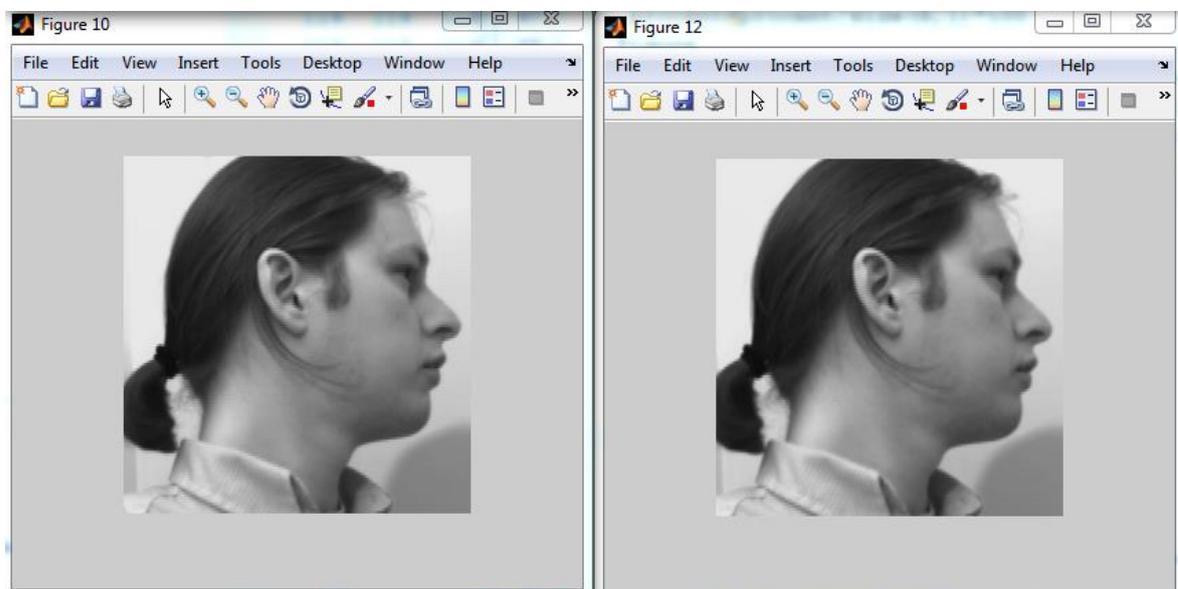
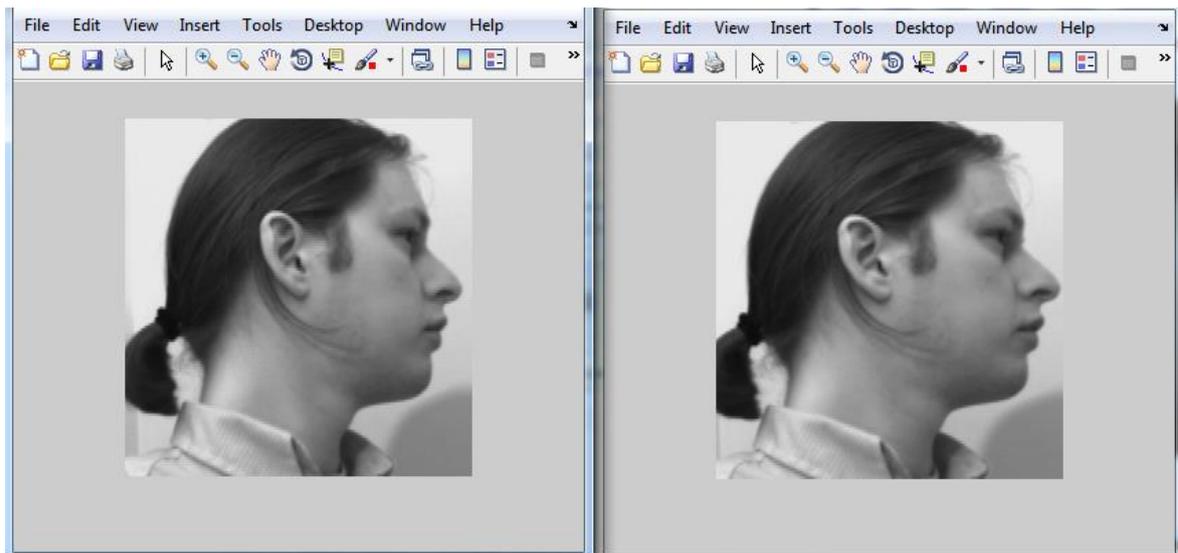
    tmp=mat2gray(double(I));
    tmp = reshape(tmp,1,size(tmp,1)*size(tmp,2));

    C = [C;tmp];”
“end;
size(C);
ST=C; % TEST sequence
YT=[1 1 1 1 -1 -1 -1 -1]; % replies regarding the origin of the images
for i=1:size(ST,1)
    ZT(i,:)= [1 ST(i,:)];
end;”

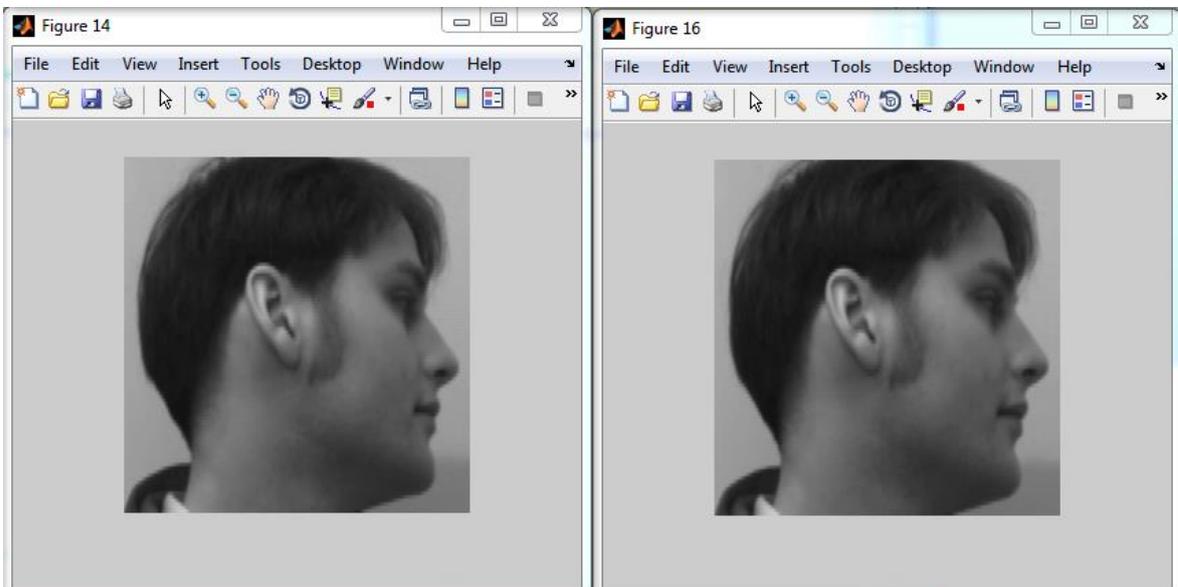
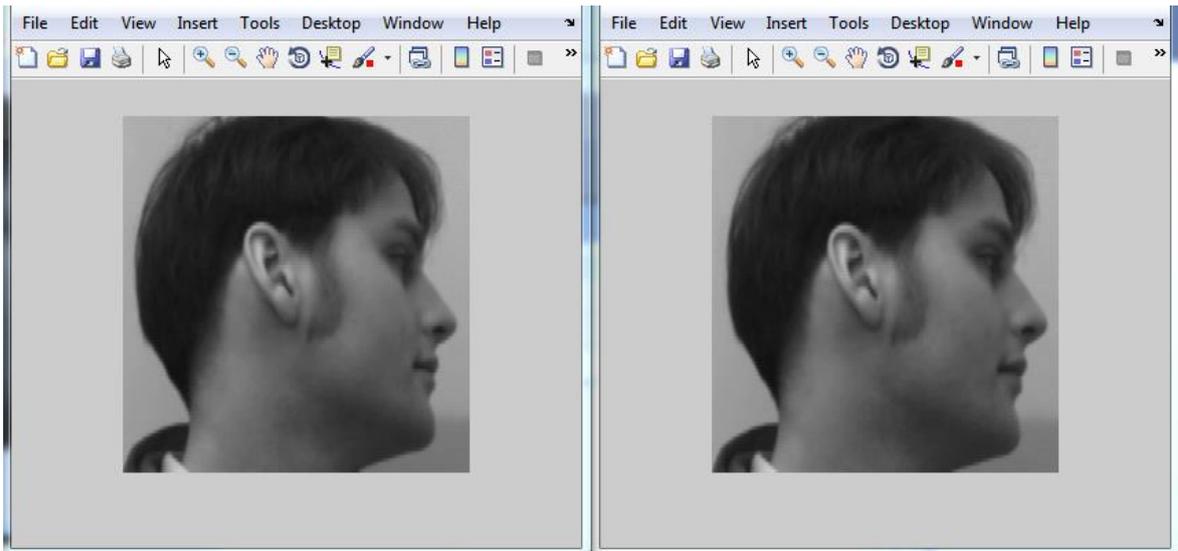
```

Stage 5: The graphic representation of test images and establishing the image recognition percentages

Test images derived from class 1:



Test images derived from class 1:



The quality of the classification of test images shall be determined by displaying the accurate recognition percentage of the new data for the two persons

```
“ST=ZT;
rt=[];

%YHT hyperplane replies for test images
for i=1:size(ST,1)
rt(i)= ST(i,:)*w';
    if rt(i)>0
        YHT(i)=1;
    else
        YHT(i)=-1;
    end;
end;”

“percentage_test=0;
for i=1:size(ST,1)
    if YT(i)==YHT(i)
        percentage_test=percentage_test+1;
    end;
end;
percentage_test=percentage_test/size(ST,1)*100”
```

3. Experimental study regarding the image recognition performance with the rendered basic neuronal algorithm

In order to test the generalization capacity of the above algorithm for the classification of images, a study was performed aiming at the experimental evaluation of the results regarding the accurate recognition of face images derived, by turn, from two classes and subsequently, through the extension of the basic algorithm, for three images derived from three classes of images. (Doru Constantin, 2016)

No. Classes	Data volume		Recognition percentage
	Input date	Test date	
2 classes	10	10	84%
	10	20	82%
	10	30	81%
	20	10	85%
	20	20	83%
	20	30	81%
	30	10	86%
	30	20	84%
	30	30	82%
	50	10	89%
	50	30	87%
	3 classes	10	10
10		20	78%
10		30	76%
20		10	80%
20		20	78%
20		30	79%
30		10	81%
30		20	78%
30		30	75%
50		10	90%
Table 1 – Experimental results for the neuronal algorithm proposed			

4. Conclusions

The purpose of this paper was to propose an image classification scheme through neuronal architectures and to demonstrate the importance of the MATLAB language, both as a programming environment and a high-performance language used in technology, which integrate numerical computing, advanced graphics, visualization and programming. It provides researchers, engineers and any scientist with an interactive, powerful and easy-to-use system where the problems and the solutions are expressed in a natural manner. Its typical uses include: mathematical computing (numeric and symbolic); development algorithms; modelling and simulation; data analysis; their exploration and visualization; scientific and engineering graphics; development applications including the creation of graphic user interfaces. The experimental results show a good recognition of the used data and thus the neuronal algorithmic scheme may be used in various applications which involve processing through two-dimensional signals.

Bibliography

- Arnold, D.; Wagner, B.; 2007, Matlab Programming, Department of Mathematics, College of the Redwoods, <http://msenux.redwoods.edu/Math4Textbook/>.
- Blaça, P.; 2002, Statistics through...MATLAB, Presa Universitară Clujeană, Cluj-Napoca.
- Doru Constantin, 28 June - 7 July, 2016, Emilia Clipici, "A new model for estimating the risk of bankruptcy of the insurance companies based on the artificial neural networks", Proceedings of the 16th edition of the SGEM International GeoConferences (International Multidisciplinary Scientific Geo Conferences), Albena Resort, Bulgaria.
- Doru Constantin, March 15-18, 2009, "Cebaysev Coefficients - Based Algorithm for Estimating the ICA Model", Conference Information: 8th International Conference on Independent Component Analysis and Signal Separation, Source: INDEPENDENT COMPONENT ANALYSIS AND SIGNAL SEPARATION, Book Series: LECTURE NOTES IN COMPUTER SCIENCE, Volume: 5441, Pag: 219-226, SPRINGER-VERLAG BERLIN, HEIDELBERGER PLATZ 3, D-14197 BERLIN, GERMANY, ISSN: 0302-9743 Paraty, BRAZILIA.
- Doru Constantin, 2-4 Mai, 2008, "Principal Directions for Local Independent Components Analysis", ADVANCED TOPICS ON NEURAL NETWORKS, Proceedings of the 9th WSEAS International Conference on NEURAL NETWORKS (NN'08), Artificial Intelligence Series, WSEAS Press, ISBN: 978-960-6766-56-5, ISSN: 1790-5109, pg. 127-130, Sofia, Bulgaria.
- Doru Constantin, June 02 - 05, 2016, Emilia Clipici, "Backpropagation neural scheme for estimating the risk of bankruptcy of the Romanian insurance and reinsurance companies", Proceedings of the 15th International Conference on INFORMATICS in ECONOMY (IE 2016), Education, Research & Business Technologies, ISSN 2284-7472, ISSN-L 2247-1480, pg 415-421, Bucharest University of Economic Studies Press, Cluj-Napoca, Romania.
- Ghinea, M.; Firețeanu, V.; 2003, MATLAB-numerical computing, graphics, applications, Teora.
- Iatan, I.; 2009, Lab Guide in Matlab 7.0, Ed. Conspress, București.
- MATLAB-, 2000, The Language of Technical Computing, MATLAB 6 (Release 12), The MathWorks Inc.
- Moler, C.; 2004, Numerical computing with MATLAB, Electronic Edition: The MathWorks, Inc., Natick, MA, <http://www.mathworks.com/moler.192>.
- The MathWorks, Inc., MATLAB – The Language of Technical Computing, Getting Started with MATLAB, www.mathworks.com.