Developing Art-1 Neural Network Interface Model for Processing and Recognition Color Images

Rashad J. Rasras
rashad_ras @yahoo.com
D.E. Skopin
Qazem M. Jaber

Faculty Of Engineering Technology, Al-Balqa' Applied University

ABSTRACT

The theoretical outcomes and experimental results of new input interface for ART1 neural network applying in algorithms and software of image analysis are presented in the paper. It is well known, that standard ART-1 network may process only binary images. The given fact limits application of ART-1 network for full color images processing and recognition because conversion from full color to binary form causes loss of the information, as example during conversion to binary form a source image lost a color and intensity components and as result two different color images look like same images in binary form, that produce a mistake during recognition. In this paper a new color interface for ART1 neural network has been developed, implemented in software and tested in real examples. As it will be shown, new interface allow to produce a full color image processing with using standard ART1 neural network.

Key words: Neural networks, interface model, processing, recognition, color images

1. Introduction

The ART-1 neural network is considered, for working with binary input pattern[1,2]. It relates input pattern of images to one of the learned classes by vigilance parameter, which determines the degree of similarities and how close a new input image to a stored prototype.When the input image is considered to be different from all prototypes, it is added to neural net as new image class. New patterns may create new classes, but could not deform an existing memory[1].

The basic architecture of ART-1 net as it is shown in Fig. 1 consists of 3 groups of neurons. Field of computational unit which consists of 2 layers of neurons: input neurons layer S and interface neurons layer Z, supplementary neurons layer Y and control neurons R, G1, G2[3]. Input binary images are presented to input layer S; signals are sent to corresponding neurons in Z layer from S layer neurons and to control neurons R, G1, G2. Each neuron of interface layer Z_i (I=1,..,m) is connected each neuron of Y layer Y_j(j=1,..,m) by two weighted pathways. The signals broadcast from Z layer neurons to neurons of Y layer over connections pathways with bottom –up weight W_{ij} and from Y units to Z unites over connections pathways with top-down weights W (j=1,..,m, i=1,..,n). Due to great number of connections in Fig. 1, it will be shown only two weights of connections pathways W. and W between interface layer and supplementary layer.

Layer Y is a layer of competitive neurons. At any time each neuron Y_j (j=1,.., m) May be in one of three states: active ("on", output signal from neuron Y_j, $Y_{jout} = 1$); inactive ("of",output signal from neuron Y_j, Y_{jout} = 0 but a available to participate in competition); inhibited (output signal of neuron Y_j, Y_{jout} =-1, neuron prevented from participating in any further computation during the presentation of the current input vector). Each neuron in either Y layer or Z layer has three sources from which it can receive signals: from the interfaces neurons and from control neurons R and G2. Similarly

for Z layer, each neuron Z_i (i=1,..,.n) can receive signals: from Y neuron, input neurons S_i and from neuron G_1 Z layer neurons or Y layer neurons are activated by at least two excitatory signals. The majority of connections pathways shown in Fig. 1 are considered to be excitatory signal: from S input layer neurons to neuron R, G_1 and Z layer neurons and from G_1, G2,R to corresponding neurons of Z layer and Y layer. Inhibitory signals send only sets of connections from neurons of interface layer to R neuron and from Y layer neurons to neuron G_I.

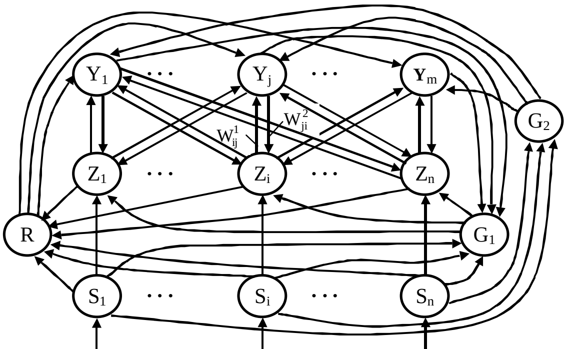

Fig.1 The basic architecture of ART-1 net

2.Developing new interface model:

The review of modern condition of a question related with practical applications of full color image processing and recognition shows that in current moment no methods and applications implemented the mechanisms of ART model for full color image processing. However numerical publications shows that the works going in two different directions: implementation of ART models for image segmentations as well as for binary and analog data processing.

The structure of standard ART1 model deals with binary images supposing that the input image is in binary form. The given fact limits application of a ART-1 network for full color images recognition because conversion from full color to binary form cause loss of the information. For adaptation the ART1 network to full color image recognition we suggest modification the network input interface (S layer) according with method described below.

True color image is bitmap where each pixel is convolution of RGB components[4]. Usually the value of each pixel in true color format represented as one

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Fig.2: Standard Bits format for pixel representation

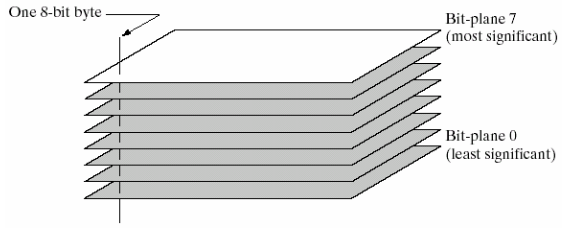

Fig.3: The bit slicing

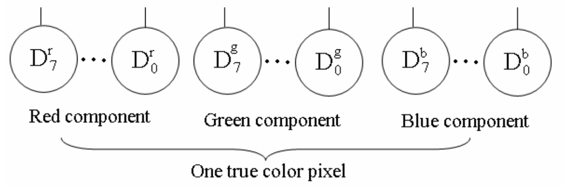

Fig. 4: The structure of ART1 input interface for true color images

integer number, but this number contain all information about RGB components. Any true color pixel is set of 3 numbers or R, G,B components. Each component is number in the range {0...255} or one byte with bits format described in Fig. 1.

One way is to display a high value for all gray levels in the range of interest (to the binary range 0...1) and a low value for all other gray levels is method called gray scale slicing [5-7]. For true color format it is possible to use this method apply bit plane slicing for R,G,B components of pixel separately.

Instead of highlighting gray-level ranges, highlighting the contribution made to total image appearance by specific bits might be desired. Imagine that the image is composed of eight 1-bit planes, ranging from bit-plane 0 for the least significant bit to bit-plane 7 for the most significant bit. In terms of 8-bit bytes, plane 0 contains all the lowest order bits in the bytes comprising the pixels in the image and plane 7 contains all the high-order bits as shown in Fig. 2 [8].

Note that the higher-order bits (especially the top four) contain the majority of the visually significant data. The other bit planes contribute to more subtle details in the image[9-12]. Separating a digital image into its bit planes is useful for analyzing the relative importance played by each bit of the image, a process that aids in determining the adequacy of the number of bits used to quantize each pixel. Also, this type of decomposition we can use for ART1 adaptation to true color image recognition, as shown in Fig. 3.

According with image we produced bit plane slicing for pixel decomposition and put bits separately to input of ART1 network.. As our investigation shows,

(A)

(B)

(C)

(D)

Fig. 5: Result of bit plane slicing. A - source image; B - image after applying the mask "00011111" (D7-D5 bits represented), C-mask "00111111", D - mask "01111111" (only bit D7)

this method is fully correct in practical implementation, but there are only one disadvantage - the numbers of network structures increased 24 times to compare with standard ART1 model. To simplification of network structure let's suppose and check the fact, that more important image information

Table 1: Changing of Image statistics according with bits number representation

Bits	Average of distribution		%Average of distribution	Standard
deviation	%of standard deviation			
D7-D0	131,23	100	86,59	100
D7-D1	131,84	98,92964	86,47	99,46996
D7-D2	132,51	97,75399	86,42	99,24912
D7-D3	133,41	96,17477	86,38	99,07244
D7-D4	138,52	87,20828	85,74	96,24558
D7-D5	145,84	74,36392	84,33	90,01767
D7-D6	159,71	50,02632	79,99	70,84806
D7	188,22	0	63,95	0

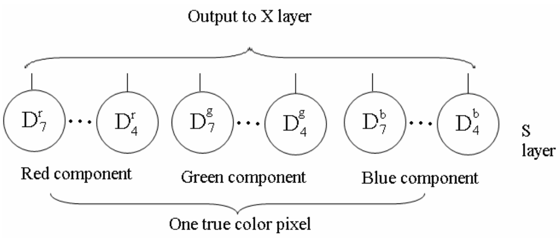

Fig. 6: The input interface of ART1 final structure network for true color image recognition

concentrated in the firsts significant bits (beginning from D7, Fig. 1). To confirm this fact we create the program that get source image and produce few other image applying bit slicing method step by step (Fig. 4).

So, in first step we have source image, second step each pixel converted with using "OR" operation with mask "00000001" (it cause bit D0 be in "1" for all image pixels), in third step program produce same operation with mask "00000011" and continue until the mask "011111111". As we can see, image A and image B visually the same quality, but we represent this image only with using bits D7,D6 and D5. Last image (D) is binary color, each color component (R,G,B) present in binary form. To study in details changing of image descriptiveness depending on masked bits we add to program feature to calculate image statistics (average of distribution and standard image deviation) before and after masking. Results of research of 35 various images are submitted in Table 1.

For better representation we recalculate percent of statistics changing according with formula 1 [13,14]

$$\%L = \frac{L - L_{\min}}{L_{\max} - L_{\min}} \cdot 100\% \qquad (1)$$

where %L is parameter of changing in percent, L - current value of statistics, Lmin and Lmax - minimum and maximum member of set.

According with data in Table 1 we can come to a conclusion that more significant statistics changing in image become after masking bit set after bit D3. It means that in practical application we can reduce ART1 input interface length (use the bits D7-D4 only) as minimum twice without losing image quality. So, final structure of input interfaces in Fig. 5.

3. Results

To study the practical approaches of new ART1 structure the program implemented a color interface of ART1 neural network has been created with using Borland Delphi 6.0 compiler[15,16]. Produced by Borland International, Delphi is a powerful development environment used primarily to build client/server applications for Microsoft Windows, with an emphasis on databases. Based on Object Pascal, it is object-oriented and was designed to give developers the ability to build applications easily, with minimal coding required. Delphi is similar to Visual Basic from Microsoft, but whereas Visual Basic is based on the BASIC programming language, Delphi is based on Pascal. The graphical user interface of program, that we created with using Delphi 6.0. illustrated in Fig. 6.

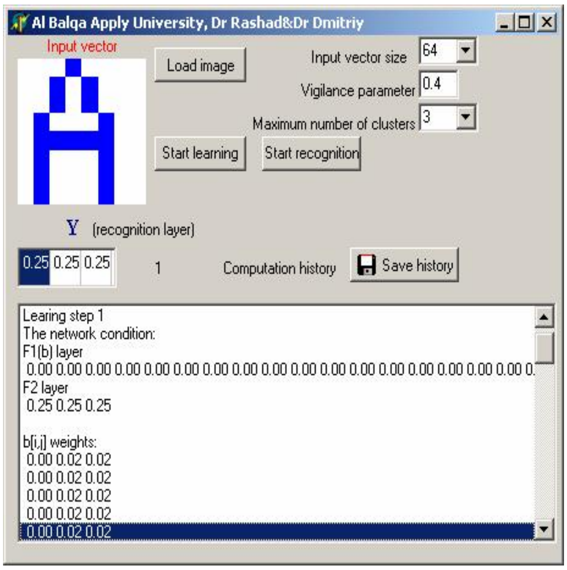

Fig. 7: Graphical user interface of ART1 color recognition program

The program allows loading up 200 input vectors (images) with different vigilance parameter in the range 0.1...1 with maximum number of clusters equal 200. To control a learning and recognition process the full computation history list with step by step explanation of calculation sets were included to this program. During experimental work we selected input vector size equal 64 to be able to process images with 8X8 bits resolution. The structure of ART1 interface (this structure explained in case of 1 recognition cluster) used during experimental work represented in Fig. 7.

For simplification a work and analysis we chose three-bits color encoding (it is situation when color image represented by one bit per each color component red, green and blue) whereas in normal condition there is 24 bits encoding (one byte per color component). That's why only bit D7 presented in model (the bits D6-D1 were deleted, Fig. 7). In other words we have simplified the input structure of color interface showed in Fig. 5 and represented each pixel as binary sequence

Table 2: The colors range of ART1 interface $D7_r$, $D7_g$, $D7_b$. As result its cause transformation of input colors to the range, described in Table

$D7_r$	$D7_g$	$D7_b$	Color meaning
0	0	0	Black
0	0	1	Blue
0	1	0	Green
0	1	1	Cyan
1	0	0	Red
1	0	1	Violet
1	1	0	Yellow
1	1	1	White

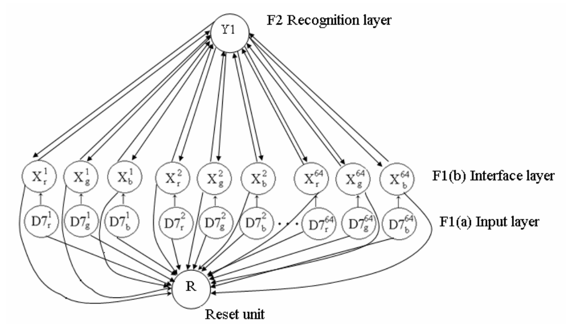

Fig. 8: The structure (1 recognition cluster) of ART1 neural network for full color processing

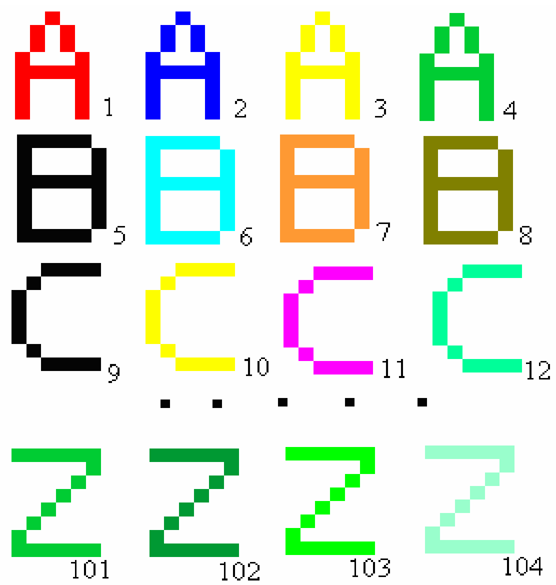

Fig. 9: The samples of chars used during network testing

.

For testing the accuracy of neural network recognition the set of samples representative letters of the Latin alphabet (A... Z) have been constructed. The set included 26 different chars, 4 random colors per each char, it's totally 104 symbols located in external graphical files (BMP format). The appearance of sample chars inside the set showed in Fig. 8, note that each char was numbered.

Experimental work has been broken on 2 parts: learning of a neural network and recognition. During learning the learning set (chars recorded in BMP file numbered in ascending order) forwarded to the network input. During recognition initial numbers of files were shuffled in random order and then sent to the input again. The output of the network (one output per one input file) is the number of recognized cluster.

Thus, entrance parameters of a neural network are:

1. Input vector size =64
2. Vigilance parameter (was vary from 0.4 until 0.8)
3. Maximum number of clusters = 104
4. Input image file in BMP format

The output parameter: number of cluster (or clusters if input char belong to few learning images)

We have provided two researches, first of them purposed to estimate abilities of the network to recognize the patterns with same object by different colors and second one to estimate statistical properties of the network at various values of vigilance parameter.

The result of first research represented in Table 3, input sequence means the numbers of input samples (Fig. 9).

In first test we used lower vigilance parameter equal 0.4 and result shows that ever in low vigilance it is possible to recognize the color of pattern. In this test the red, green and blue A chars recognized correctly but yellow char (sample number 3) clustered as green (sample number 4). We can explain this fact that binary representation of red, green and blue color is 100,010,001 accordingly (all bits is different), yellow 110 - first bit cross linked with middle bit of green color. In second test all chars recognized correctly due moderate vigilance parameter. The third test shows ability of network for color interpolation, so pattern 8 (dark grey) interpolated as pattern 5 (black). This ability is easy to explain in next example: the hexadecimal representation of black is 00H(Red),00H(Green),00H (Blue), for dark grey - 00H (Red), 70H (Green), 70H (Blue), but in 3 bits notation the bit D7 equal zero for both colors and as result dark grey color interpolated to black in input interface.

Table 3: Testing the recognition ability of the network

Test number	Vigilance parameter	Learning sequence	Input sequence	The network outputs (number of cluster)	Note
1.	0.4	1,2,3,4	1,4,3,2	1,3,3,2	Small vigilance parameter caused clustering into 3 clusters
2.	0.75	1,2,3,4	1,4,3,2	1,4,3,2	All patterns recognized correctly
3.	0.75	5,6,7,8	8,6,5,7	5,6,5,7	Pattern 8 (dark grey) recognized as pattern 5 (black)
4.	0.75	101,102,103,104	104,101,103, 102	102,101, 101,101	All patterns clustered into two clusters (see explanation below)

Table 4: Testing of recognition ability of the network

Test number	Vigilance parameter	Learning sequence	The number of true positives	The number of mistakes	Sensitivity
1.	0.4	124 files	70	54	0.56
2.	0.5	124 files	96	28	0.77
3.	0.6	124 files	112	12	0.90
4.	0.7	124 files	119	5	0.96
5.	0.8	124 files	119	5	0.96

The fourth test also demonstrated good network color interpolation, during test patterns 101-103 interpolated as green, last pattern number 104 recognized as white color because source color too bright (all D7 bits equal one).

To study the recognition accuracy of the network we generated additional set of Latin alphabet (A...Z), but only native colors were included to this set (red, green, blue, yellow). The statistical parameter sensitivity calculated according with equation 1[13,14]

$$\text{Sensivity} = \frac{\text{Number of true positives}}{\text{Number of truepositives} + \text{Number of false Negatives}} \quad (1)$$

The result of research represented in Table 4.

The results shows that the network is able to recognize different colors in vigilance parameter ranged 0.7 - 0.8. Less value of vigilance reducing sensitivity of network, on the other hand moderate vigilance cause dead loop in algorithm of pattern recognition.

4. Conclusion

It is known, that the standard ART1 network is intended for work with a binary input vector[17-23] (bit map image) and this fact complicates using of ART1 technology for color image recognition applications because real images represented in true color format and conversion to binary form caused information loss. We show that with using of bit slicing method it is possible to represent the color pixel as the set of binary numbers and use this binary sequence as input of neural network.

Also it was shown (Table 1), that in real images most significant information concentrated in first 4 bits (D7-D4) that allows to use 12-bit encoding without loss of the image visual quality (Fig. 4). The Fig. 4 also shows that in practical applications it is possible to use binary colors. It is three bits color encoding, reduced model of input interface (using significant bit D7, Fig. 7) as we show in previous paragraph and in this case ART1 neural network has interpolation ability that allow to interpolate any input color to the nearest color listed in Table 2.

Our investigations show, that good results of ART1 recognition (the sensitivity at level 0.96) it is possible to get when the value of vigilance parameter located in the range 0.7-0.8. The less value reducing network sensitivity, high value can cause dead loop in algorithm of pattern recognition.

Conducted researches have shown efficiency of proposed approach to color image recognition, but suggested method has one disadvantage: even with using binary color images (1 bit encoding) the size of ART1 neural network increased three times to compare with standard ART binary network. If we deal with 4 bits encoding the size of network will be growth up 12 times. That is why we apply this technology for chars recognition because for real images with size more than 100X100 pixels computations hold too much time (up few minutes). That is why our future work concentrated in the field of parallel processing implementation for full color processing and recognition with using ART1 neural network.

References

[**1**] Carpenter, G.A. and S. Grosberg, 1987. Massively parallel architecture for self-organizing Neural pattern recognition, Machine// computer vision. Graphics and Image Processing, 37: 45115.

[**2**] Faulset, L., 1994. Fundamentals of Neural Networks. Architectures, Algorithm and Applications. New Jersey, Prentice Hall Int., pp: 461.

[**3**] B. Moore, D. Touretzky, ART1 and pattern clustering, in //Proceedings of the connectionist models summer school. Morgan Caufmann, LosAltros, CA, 1998, pp. 174–185.

[**4**] R.E. Blake, Development of an Incremental Graph Matching Device, Pattern Recognition Theory and Applications, NATO ASI Series, Vol 30, pp355-366, Springer, Berlin, 1987.

[**5**] Toru Nakanishi and Yuji Sugiyama: Sealed-bid Auctions with Efficient Bids Using Secure Bit-slicing Conversion, IEICE Trans. Fundamentals, Vol.E87-A, No.10, pp.2533-2542, October 2004.

[**6**] Peter M. Maurer, William J. Schilp, Software Bit-Slicing: A Technique for Improving Simulation Performance\\Design, Automation and Test in Europe (DATE '99) p. 786

[**7**] Gonzalez, R.C. and R.E. Woods, Digital Image Processing. 1992, Reading, Massachusetts: Addison-Wesley. 716.

[8] Castleman, K.R., Digital Image Processing. Second ed. 1996, Englewood Cliffs, New Jersey: Prentice-Hall.

[9] F. van der Heijden, Image Based Measurement Systems, Wiley, 1994.

[10] B.J. Schachter, L.S. Davis, A. Rosenfeld, Some experiments in image segmentation by clustering of local feature values, Pattern Recognition 11 (1979) 19–28.

[11] Skopin D.E. Optical topographer for diagnosis of support-locomotion system diseases //"Medical and ecological information technology". Kursk, 2005, P48-

[12] J. Liu, Y.H. Yang, Multiresolution color image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 16 (7) (1994) 689–700

[13] C.Rose, M.Smith. Mathematical statistics with Mathematica. Springer publisher co. 2003.
ISBN: 0152263522

[14] C.Chatfield. Statistics for Technology: A Course in Applied Statistics, Third Edition CRC press, 2005. ISBN: 0412253402

[15] Kerman F., Miitchell C. Programming and Problem Solving With Delphi. UK INTERNATIONAL ED, Pearson Higher Education Date Published: 2001 ISBN: 0321204417

[16] Rachele, Warren Learn Object Pascal with Delphi. Wordware Publishing inc. ISBN: 1556227191.

[17] Carpenter GA, Markuzon N. ARTMAP-IC and medical diagnosis: Instance counting and inconsistent cases. Neural Networks 1998; 11:323-336

[18] Markram H, Tsodyks M. Redistribution of synaptic efficacy between neocortical pyramidal neurons. Nature 1996; 382:807–810

[19] In L.C. Jain, B. Lazzerini & U. Halici (Eds.), Innovations in ART Neural Networks ISBN: 3790812706 Pub. Date: January 2000 Series: Studies in Fuzziness and Soft Computing , Vol. 43 2000, XII, 258 p., 69

[20]G.T.Raju,S.Khandige. ART Neural Network based Clustering Agent for Use acces

Patterns\\Computational Intelligence 2005, 487-042. ISBN: 0-88986-479-9;

[21]Hecht-Nielsen,Robert, Neurocomputing, Addison-Wesley, ISBN 0-201-09255-3, 1990.

[22] Maren, Alianna, and Harston, Craig, and Pap, Robert, Handbook of Neural Computing Applications, Academic Press, ISBN 0-12-546090-2, 1990

[23] Schalkoff, R. J., Pattern Recognition: Statistical, Structural, and Neural Approaches, John Wiley & Sons, New York, NY, 1992.