Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 84 (2016) 41 – 48

Procedia Computer Science

7th International conference on Intelligent Human Computer Interaction, IHCI 2015

Hand Drawn Optical Circuit Recognition

Mahdi Rabbani[a,*], Reza Khoshkangini[b], H.S.Nagendraswamy[a], Mauro Conti[b]

[a]*Department of Computer Science, University of Mysore, India*
[b]*Department of Computer Science, University of Padova, Italy*

Abstract

Electrical diagram is foundation of studies in electrical science. A circuit diagram convey many information about the system. Behind any device there are plenty of electrical ingredients which perform their specific tasks, today all the electrical software tools failed to effectively convert the information automatically from a circuit image diagram to digital form. Hence electrical engineers should manually enter all information into computers, and this process takes time and bring errors with high probability. Moreover, when the diagram is hand drawn, the problem is more complicated for any electrical analysis. Thus, in this paper we propose a new method using Artificial Neural Network (ANN) to make a machine that can directly read the electrical symbols from a hand drawn circuit image. The *recognition process* involves two steps: first step is feature extraction using shape based features, and the second one is a classification procedure using ANN through a back propagation algorithm. The ANN was trained and tested with different hand drawn electrical images.The results show that our proposal is viable and brings good performances.

Peer-review under responsibility of the Organizing Committee of IHCI 2015
Keywords: Optical circuit recognition; electrical circuit; moments of image; artificial neural network; back propagation algorithm.

1. Introduction

In electrical science, a circuit diagram is a graphical representation of an electrical circuit that contains simple picture in order to represent the electrical components and connectivities between components. For example, a resistor in an electrical diagram has a value (a scale of Ohm) and two connections with the other components. People can recognize these electrical components using their knowledge which they already trained. Thereafter, it is needed to manually enter the components into machine to perform the related process even the diagram is complicated with huge numbers of components. This procedure is not feasible in case of complexity of the diagrams.

Pattern recognition or object identification is a simple process in humans and other organisms, due to the highly developed sense organs. Pattern is something that repeats in a way that it can be predictable and identified. In order to handle the aforementioned problem in identification of electrical symbols in a hand drawn electrical image, we proposed a new circuit recognition method in which the machine can directly read a complex circuit diagram. After understanding the recognition process in a human brain, we created the same capability in machine. To end this,

* Corresponding author.
E-mail address: Mahdirabbani1983@gmail.com (M.Rabbani), khosh@math.unipd.it (R.Khoshkangini), swamy_hsn@yahoo.com (H.S.Nagendraswamy), conti@math.unipd.it (M.Conti).

Peer-review under responsibility of the Organizing Committee of IHCI 2015
doi:10.1016/j.procs.2016.04.064

we use Artificial Neural Network (ANN) to recognize and identify the electrical characters from an electrical circuit image. Therefore, the scanned circuit diagram image can be given to the machine. In the first step, the raw image is converted into black and white (binarization) pixels with a standard size (normalization). In the next step, future extraction takes place using image's moments, and then these extracted features are ready to feed in the input layer of artificial neural network for training at every feeding stage. The network calculates the error with respect to the desired class using back propagation algorithm[4].

The goal of this study is giving ability to computer to directly read the components from an image, which solves the manual data entry problems. In addition, the diagram can be digitized and preserved. Moreover, the main contribution of the paper is converting electrical images to an understandable format (none image) for computer, using back propagation neural network. Table 1 shows the symbols are used in this study.

The rest of the paper is organized as follows. In Section II related works are reported. In Section III the statement of the problem is explained. In Section IV image processing and feature extraction are provided. Design and methodology are introduced in Section V. Section VI points to the experimental result, and followed by the conclusion in section VII.

Table 1: The symbols are used in this study.

Symbol	Description	Symbol	Description
V	voltage	m	milli
Ω	Ohm	R	resistor
F	Farad	L	inductor
H	Henry	M	mega
k	kilo	μ	micro
C	capacitor	mh	mho

2. Related work

Optical character recognition (OCR) could be applied as an aid in two issues: growing telegraphy and creation of perusing gadgets for the visually impaired. Several studies in electrical image recognition have been intensively investigated to identify the electrical elements from images[5,13]. Radio Corporation of America (RCA) proposed an OCR, which was an early effort to help visually impaired individuals for US Veterans Administration. It changed the typewritten documents into punch cards in the magazine's membership office in order to help in preparing the shipment of 15-20 million books a year. Around 1965, Reader's Digest and RCA teamed up to fabricate an OCR document per user. In gesture recognition, Rubine[6] proposed an 11-dimensional element vectors to depict a solitary stroke. In spite of the fact that works in[2,7] are similar to Robine's method, which incorporates stroke segmentation, their methods are able to identify images which are made by different strokes. Sezgin and Davis[15] introduced full utilization of distinctive individuals attracting styles to enhance both effectiveness and execution. Later, they improved their method to recognize blurred symbols[15]. Researchers in[10] proposed a left right (LR) based representation parsing system. Gennari et al.[12] explained ink density and stroke attributes to specify competitor characters. Hammond and his colleague proposed a sketching language (LADDER)[16] such that in any cases their methodology can only depict regular shapes without details. Although the aforementioned studies focused on creating automatic recognition strategies, only few of the them have tended to the issue of managing scattered images[18,19,20].

A different kinds of network topology has been used in circuit diagrams to improve the recognition process[11,14]. Once more, due to the moderate dismissing capacity of most image's classifiers with the anomalies, we believe that using an artificial neural network greatly helps to solve such problems with high accuracy which it has been successfully applied in different domains such as stock market prediction, Malware recognition, Traveling Saleman's Problem, etc[21,22,23]. More studies on detached image recognition can be found in[8].

Table 2: The list of electrical circuit characters which are used in this study

Numbers	Character	Description	Total
1	Numbers	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	10
2	Alphabets	C, E, L, R, V	5
3	Units	micro, ohm, mho, milli, Mega, voltage, kilo, Henry, Farad	9
4	Symbols	resistor, capacitor, inductor, AC V source, DC V source (D1 and D2)	7
Total numbers of elements			31

3. Statement of the problem

Since the analyze of electrical circuit is extremely difficult where the structure of diagram is complex, *electrical circuit diagrams* have been provided to visualize the components in an electrical circuit and their connections. In order to use or improve them, usually electrical engineers must manually feed the diagram to machine via a design software and convert them from an image diagram to a particular format which is understandable for computer. In spite of the fact that the conversion process is slow with some errors (e.g., outline, overlap and unknown components which may happen during the drawing process), the process is dramatically fast when we use a machine. Hence there is a need in optical systems to intelligently recognize the isolated components (which are mentioned in Table 2) in an electrical circuit. Consequently, in this study we use supervised learning in artificial neural network to recognize the electrical symbols, which leads to design an optical circuit recognition system with following steps:

- Recognizing circuits symbols such as resistor, capacitor, inductor and voltage source.
- Identifying the associated text and numbers with the elements of the circuits.
- Announcing suitable error messages if a symbol cannot be recognized properly.

4. Image Pre-processing and Feature Extraction

Generally, the proposed methodology of this study consists of *Pre-proseccing* and *Recognition* modules. In the first module the image diagram should be prepared for feature extraction, and then we extract the important components using moments of image. In the second module, we develop an Artificial Neural Network to recognize the electrical symbols, which Figure 1 shows the conceptual overview of the proposed methodology.
First part of pre-processing section involves a set of operations which should be applied on a scanned input image (raw image). The operation essentially enhances the image rendering which is necessary for segmentation[13]. Basically, the role of Pre-processing's is to segment the interesting patterns from the background. In addition, it is in charge of *noise filtering, smoothing* and *normalization* of isolated components[13]. In the binarization process a gray scale image is converted into a binary image via suitable threshold, and in the normalization process input data is converted into a proper format, which is needed for the neural network[3]. Binarization and normalization processes are common processes in image processing and pattern recognition field, but the main task in the feature extraction part is to exploit the principal components and important specifications from the isolated image. For example; in order to detect digits via moments, first it is necessary to obtain all the moments, then extract and convert them into a format which can be given as input to the neural network. And finally, the output of neural network is the recognized digit[5].
As we mentioned above, the moment's extraction process includes the following steps:

- Loading image from the training data set.
- Image binarization.
- Image normalization.

- Converting input image to a matrix.
- Converting the matrix to vector of moments.
- Concatenate all vectors of moments (1705) to make one matrix.

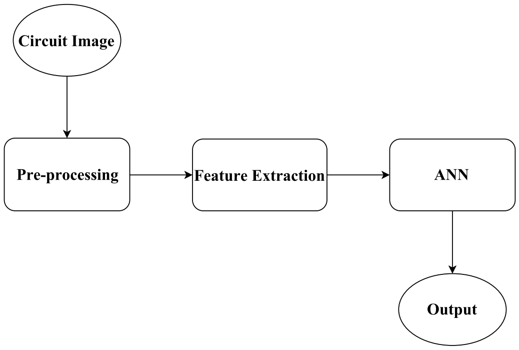


Fig. 1: Conceptual overview of the proposed method

4.1. Feature Extraction Using Moments of Images

Image moment is a statistical and geometrical attribute which is extracted from the configuration of an image. In addition, it is a scalar quantities to extract the significant features[24,25]. From the mathematical point of view, moment is a functionalization process to polynomial basis, which is useful to describe objects after segmentation. Image moments identify the properties of a simple image including *intensity, its centroid* and *orientation information*. In this study we use moments in order to represent an input image (in a matrix form) which indicates an image from 0 to 255 pixels. Thereafter, we use Otsu's method[26] to convert the image to black and white pixels. Since the size of the image is still too large, we convert the binary image to a normal image (e.g., 200×200). Hence, the output of the conversion is a one dimensional vector which contains image moments with 220 values. The process continues iteratively until all images in the training set are met (the training dataset contains 1705 images). In the end: the process generates a matrix with 1705 rows and 220 columns. The image moments consist of the following elements.

- Raw moments.
- Central moments.
- Scale invariant moments.
- Rotation invariant moments.

Eq. (1), shows a 2-dimensional function f(x,y) of raw moment of order (p + q):

$$M_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x^p y^q f(x,y) dx dy. \tag{1}$$

Where $p, q = 0, 1, 2, ..., \infty$. x and y denote the pixel densities. The central moments of function f(x, y) are defined in Eq. (2):

$$\mu_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (x - \overline{x})^p (y - \overline{y})^q f(x,y) dx dy. \tag{2}$$

Where $\bar{x}$ and $\bar{y}$ are the elements of the centroid such that:

$$\bar{x} = \frac{M_{10}}{M_{11}} \quad and \quad \bar{y} = \frac{M_{01}}{M_{00}}, \tag{3}$$

μ_{pq} in Eq. (2), is invariant moment under the translation of coordinates[27,28]:

$$x' = x + \gamma \quad and \quad y' = y + \delta, \tag{4}$$

Where γ and δ are the constant values. Scale invariance acquired by normalization which the normalized central moments are shown in Eq. (5),[29]:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{(1+\frac{p+q}{2})}}, \tag{5}$$

5. Recognition Methodology

5.1. Training an Artificial Neural Network

It is necessary in supervised learning to train the network with hand drawn electrical symbols. To this end, 55 samples have been prepared for each symbol, numbers and some other alphabets.

5.2. Artificial Neural Network (ANN)

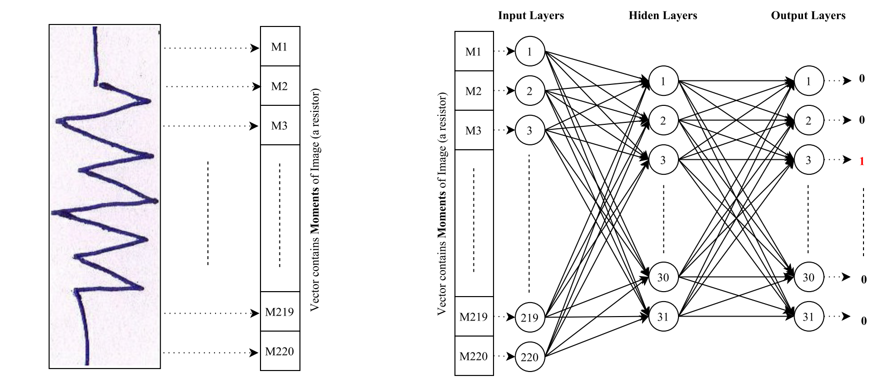
An Artificial Neural Network (ANN) is a powerful classifier that represents complex input and output correlations. It resembles the human brain in acquiring knowledge through learning and storing knowledge within inter-neuron connection strengths. Since, the ANN's synaptic weights are adjusted or trained, a particular input leads to a specific desired or target output[8].

There are basically two types of neural network structures; feed-forward network (acyclic) and recurrent network (cyclic). In this study we use feed-forward network which denotes a function of its current input to gradually teach the network system. Fig. 2(a), shows a block diagram in a supervised learning ANN, where the network is adjusted based on comparing the output of neural network and desired output until the output is matched with the desired output. When the network is trained for the first time, it can be used to test the new input data via the weights which were provided from the training session[2]. Fig. 2(b), indicates feed-forward neural networks and the simulation of neural networks topology (220×31×31). In this example of neural networks, there are 1705 training samples in input layers and 31 in hidden layers which lead to the creation of 31 outputs after classification (see Fig. (3)), which shows the identification of 31 characters from 1 to 31.

ANN in this study involves two *Training* and *Testing* parts: the former part is in charge of transform the training set and generating the model of a neural network[13]. Thereafter, it classifies the test set by the generated model in order to show the result. If the classification error is high, it can train the ANN with the new parameters in training set. This process should be repeated iteratively till the desired level of accuracy is achieved[6]. The implementation of later part (testing) is simple and straightforward such that the same procedures (in training set) have been applied to load and analyze the two parts (training and testing). In addition, the computer network parameters of input vectors in the training phase can be reused in the testing phase[2,10].

5.3. Using back propagation algorithm to reduce misclassification error

Back Propagation is a method for training an Artificial Neural Network. The network must be provided with both sample inputs and anticipated outputs when using a supervised training technique. The production of output layer is compared with the actual outputs and then the Back propagation algorithm calculates the error of comparison between desired outputs and the actual output. In Back propagation process, adjustment function by means of the achieved error modifies the weights of the various layers from the output layer to the input layer[4]. For example, a simple classification, where the input is an image of a fruit (e.g., an apple), and the correct output is the name of the fruit. Since single-layer perception cannot always learn some relatively simple patterns in which some input and output patterns are not linearly separable, we use a multi-layer artificial neural network to handle the problem[2].



(a) Converting moments of an image to vectors (b) Feed-forward neural network

Fig. 2: General schematic diagram of the system

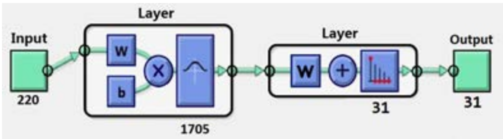


Fig. 3: Output network in MATLAB

6. Experimental results and analysis

In order to train the neural network, we collect all samples such as digits and electrical symbols (resistor, capacitor, inductor, voltage source, micro, ohm and alphabet associated with electrical circuits). In addition, for the testing part we use 20 different hand drawn images in each class.

6.1. Recognition Accuracy

For evaluation and calculating the accuracy of our proposed optical circuit recognition, we use the confusion matrix which is shown in Tables 3(a) and 3(b). As it mentioned above, we used 20 different hand drawn samples to test the procedure and obtaining the accuracy. Hence the confusion matrix has 31 rows and 31 columns, where in every row denotes one class. After feeding all the samples for testing purpose, all diagonal cells in the confusion matrix are correct classifications and other cells are mis-classifications. In addition, we used *Precision, Recall* and *F-measure* which are the basic measures in evaluating and calculating the search strategies in the following Equations and results:

$$Precision = \frac{TP}{(TP + FP)} = 0.8538, \tag{6}$$

$$Recall = \frac{TP}{(TP + FN)} = 1, \tag{7}$$

$$Fmeasure = 2 * \frac{(Precision * Recall)}{(Precision + Recall)} = 0.8364 \tag{8}$$

According to Table (4) the achieved results (Precision=0.8538, Recall=0.8338 and F-measure=0.8364) by the proposed system show that our method has high accuracy to recognize the electrical symbols from hand drawn electrical circuit.

Table 3: Confusion Matrices

(a) Confusion Matrix belong to class 1 to 15

		class1	class2	class3	class4	class5	class6	class7	class8	class9	class10	class11	class12	class13	class14	class15
		0	1	2	3	4	5	6	7	8	9	C	E	F	H	k
class1	0	17	0	0	0	1	0	0	0	0	0	1	0	0	0	0
class2	1	0	19	0	0	0	0	0	0	0	0	0	0	0	0	0
class3	2	0	0	18	0	1	0	0	0	0	1	0	0	0	0	0
class4	3	1	0	0	16	0	0	0	0	0	0	0	0	0	0	0
class5	4	0	0	2	0	16	1	0	0	0	0	0	0	0	1	0
class6	5	0	0	0	0	0	16	1	0	1	0	1	1	0	0	0
class7	6	0	0	0	0	0	1	15	1	1	0	0	1	0	0	0
class8	7	2	0	1	1	0	0	0	15	0	0	0	0	0	0	1
class9	8	0	0	1	0	0	0	1	1	17	0	0	0	0	0	0
class10	9	1	0	1	0	0	0	0	0	0	17	0	0	0	0	0
class11	C	1	0	0	0	0	0	0	0	0	0	19	0	0	0	0
class12	E	0	0	0	0	0	0	0	1	0	0	1	17	0	0	0
class13	F	0	0	0	0	0	0	0	0	0	0	1	0	18	0	0
class14	H	0	0	0	0	0	0	0	0	0	0	1	0	0	16	1
class15	k	0	0	0	0	0	0	0	0	0	0	1	1	0	0	15

(b) Confusion Matrix belong to class 16 to 31

		class16	class17	class18	class19	class20	class21	class22	class23	class24	class25	class26	class27	class28	class29	class30	class31
		L	M	R	V	m	Vl	C1	C2	In	μ	Ω	Re	AS	D1	D2	Mh
class16	L	19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
class17	M	0	18	0	0	0	0	0	0	0	0	0	0	0	0	0	0
class18	R	0	0	15	0	0	0	0	0	0	0	0	0	0	0	0	0
class19	V	0	0	0	18	0	0	0	0	0	0	0	0	0	0	0	0
class20	m	0	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0
class21	Vl	0	0	0	3	0	15	0	0	0	0	0	0	0	0	0	0
class22	C1	0	0	0	0	0	0	15	0	0	0	0	0	0	0	0	0
class23	C2	0	0	0	0	0	0	1	15	0	0	2	0	0	1	0	1
class24	In	0	0	0	0	0	0	0	0	13	1	2	0	0	2	0	1
class25	μ	0	0	0	0	0	0	0	0	0	17	1	0	0	2	0	0
class26	Ω	0	0	0	0	0	0	0	0	0	0	19	0	0	1	0	0
class27	Re	0	0	0	0	0	0	1	1	0	0	0	18	0	0	0	0
class28	AS	0	0	0	0	0	0	0	0	0	0	1	2	15	2	0	0
class29	D1	0	0	0	0	0	0	1	0	0	0	0	0	0	19	0	0
class30	D2	0	0	0	0	0	0	0	0	0	0	0	2	0	0	15	3
class31	Mh	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	18

Table 4: Performance measurements

	class1	class2	class3	class4	class5	class6	class7	class8	class9	class10	class11	class12	class13	class14	class15	class16	class17	class18	class19	class20	class21	class22	class23	class24	class25	class26	class27	class28	class29	class30	class31	Total
	0	1	2	3	4	5	6	7	8	9	C	E	F	H	k	L	M	R	V	m	Vl	C1	C2	In	μ	Ω	Re	AS	D1	D2	Mh	
True Positive (TP)	17	19	18	16	16	16	15	15	17	17	19	17	18	16	15	19	18	15	18	17	15	15	15	13	17	19	18	15	19	15	18	
True Negative (TN)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
False Positive (FP)	3	1	2	4	4	4	5	5	3	3	1	3	2	4	5	1	2	5	2	3	5	5	5	7	3	1	2	5	1	5	2	
False Negative (FN)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Precision	.77	1	.72	.88	.84	.84	.83	.75	.89	.94	.61	.85	1	.84	.88	.95	.94	.75	.72	1	.88	.83	.88	1	.89	.76	.81	1	.57	1	.78	0.8538
Recall	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
F-measure	.8	.97	.8	.84	.82	.82	.78	.75	.87	.89	.74	.85	.94	.82	.81	.95	.92	.75	.8	.91	.81	.78	.81	.78	.87	.84	.85	.85	.71	.85	.83	0.8364

7. Conclusion

Artificial neural networks are generally used to apply character recognition which have been proven to yield excellent results. Although, a poorly chosen set of features will yield poor classification rates. In this paper we used the artificial neural network capability in order to recognize and identify the electrical characters from an electrical image using image's moments. The experiment results show that the proposed method enables recognition and identification with high accuracy. At the current stage of development, the software performs well either in terms of speed or accuracy but not better[3]. Thus OCR can become a powerful tool for future data entry applications.

References

1. Donald Bailey, Andrew Norman and Giovanni Moretti. Electronic Schematic Recognition. *Massey University*, 1995.
2. Chris Calhoun, Thomas F. Stahovich, Tolga Kurtoglu and Levent Burak Kara. Recognizing multi-stroke symbols. *Proceedings of AAAI Spring Symposium on Sketch Understanding*, AAAI Press, Palo Alto; 2002, p. 15-23.
3. Deepayan Sarkar. Optical Character Recognition using Neural Networks. *Madison ECE 539 Project*, Fall 2003.
4. Nawrin Binte Nawab and M. M. Hassan. Optical Bangla character recognition using chain-code. *Informatics, Electronics Vision (ICIEV), 2012 International Conference*, May 2012, p. 622-627.
5. Marcus Liwicki1 and Lars Knipping. Recognizing and simulating sketched logic circuits. *Knowledge-Based Intelligent Information and Engineering Systems, Springer Berlin Heidelberg*. Vol. 3683; 2005, p. 588-594.
6. Dean Rubine. Specifying gestures by example. *Computer Graphics 25(4)*; 1991, p. 329-337.
7. Heloise Hse, Michael Shilman and Richard Newton. Robust sketched symbol fragmentation using templates. *Proceedings of International Conference on Intelligent User Interfaces (IUI)*; 2004, p. 156-160.

8. Liu Wenyin, et al. On-line graphics recognition. *International Workshop on Graphics Recognition: Recent Advances and Perspectives, Springer, Barcelona*, Vol. 3088; 2003. p. 289-302.
9. Tevfik Metin Sezgin and Randall Davis. Sketch interpretation using multi-scale models of temporal patterns. *Computer Graphics and Applications, IEEE 27.1*; 2007. p. 28-37.
10. Gennaro Costagliola and Vincenzo Deufemia. Visual language editors based on LR parsing techniques. *Proceedings of the Eighth International Workshop on Parsing Technologies, Nancy*; 2003, p. 78-90.
11. Levent Burak Kara and Thomas Stahovich. Hierarchical parsing and recognition of hand-sketched diagrams. *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology (UIST2004), ACM Press*, Santa Fe; 2004, p. 13-22.
12. Leslie M. Gennari, Levent Burak Kara and Thomas F. Stahovich. Combining geometry and domain knowledge to interpret hand-drawn diagrams. *Computer Graphics 29*; 2005, p. 547-562.
13. Manuel J. Fonseca and Cesar Pimentel and Joaquim A. Jorge. CALI: An on-line scribble recognizer for calligraphic interfaces. *Proceedings of AAAI Spring Symposium on Sketch Understanding, AAAI Press*, Palo Alto; 2002, p. 51-58.
14. Christine Alvarado and Randall Davis. Dynamically constructed bayes nets for multi-domain sketch understanding. *Proceedings of the International Joint Conference on Artificial Intelligence, AAAI Press*, Edinburgh; 2005, p. 1407-1412.
15. Tevfik Metin Sezgin and Randall Davis. Sketch recognition in interspersed drawings using time-based graphical models. *Computer Graphics* **32**; 2008, p. 500-510.
16. Tracy Hammond and Randall Davis. LADDER, a sketching language for user interface developers. *Computers and Graphics 29.4*; 2005. p. 518-532.
17. Eric Saund, et al. Perceptual organization as a foundation for intelligent sketch editing. *Proceedings of AAAI Spring Symposium on Sketch Understanding, AAAI Press*, Palo Alto; 2002, p. 118-125.
18. N.Zlatoff, B.Tellez and A.Baskurt. Image understanding and scene models: a generic framework integrating domain knowledge and gestalt theory. *Proceedings of the International Conference on Image Processing, IEEE Press, Singapore*, Vol. 4, 2004. p. 2355-2358.
19. Xie, Lexing and et al. Structure analysis of soccer video with domain knowledge and hidden Markov models. *Pattern Recognition Letters 25.7*; 2004. p. 767-775.
20. Michie Donald, et al. Machine learning, neural and statistical classification; February 17 1994.
21. Parvez Faruki, et al. Android Security: A Survey of Issues, Malware Penetration, and Defenses. *Communications Surveys and Tutorials, IEEE, Secondquarter 2015*, Vol.17, no.2, p. 998-1022.
22. Smita Naval, Vijay Laxmi, Manoj Gaur, Sachin Raja, Muttukrishnan Rajarajan and Mauro Conti. Environment-Reactive Malware Behavior: Detection and Categorization. *In Proceedings of the 3rd International Workshop on Quantitative Aspects in Security Assurance (ESORICS 2014 workshop: QASA 2014), in press, Wroclaw, Poland, September 10-11, 2014*.
23. Reza Khoshkangini, Syroos Zaboli, and Mauro Conti. Efficient Routing Protocol via Ant Colony Optimization (ACO) and Breadth First Search (BFS). *Proceedings of the IEEE International Conference on Cyber, Physical and Social Computing (IEEE CPSCom 2014)*, in press, Taipei, Taiwan; 2014.
24. Zafer Iscan, Ayhan Yksel, Zmray Dokur, Mehmet Korrek and Tamer olmez. Medical image segmentation with transform and moment based features and incremental supervised neural network. *Digital Signal Processing*; 2009, Vol. 19. p. 890-901.
25. Yaser Daanial Khan, Sher Afzal Khan, Farooq Ahmad and Saeed Islam. Iris Recognition Using Image Moments and k-Means Algorithm. *The Scientific World Journal*; 2004.
26. H. Tian, S. K. Lam, T. Srikanthan. Implementing Otsu's thresholding process using area-time efficient logarithmic approximation unit. *Circuits and Systems, 2003. ISCAS'03. Proceedings of the 2003 International Symposium on. Vol. 4. IEEE*; 2003.
27. Laptev and Ivan. *Visual Recognition: Objects, Actions and Scenes*; 2014.
28. Milan Sonka, Vaclav Hlavac and Roger Boyle. Image processing, analysis, and machine vision. *Cengage Learning*; 2014.
29. Zhihu Huang and Jinsong Leng. Analysis of Hu's moment invariants on image scaling and rotation. *Computer Engineering and Technology (ICCET), 2010 2nd International Conference IEEE*, Vol. 7; 2010.