Global Journal of Advanced Computer Science and Technology

Volume No. 13
Issue No. 1
January - April 2025

ENRICHED PUBLICATIONS

ENRICHED PUBLICATIONS PVT.LTD

JE - 18,Gupta Colony, Khirki Extn,
Malviya Nagar, New Delhi - 110017.
E- Mail: info@enrichedpublication.com
Phone :- +91-8877340707

Global Journal of Advanced Computer Science and Technology

Aims and Scope

Global Journal of Advanced Computer Science & Technology deals with information technology, its evolution and future prospects and its relationship with the Business Management. It addresses technological, managerial, political, economic and organizational aspects of the application of IT in relationship with Business Management. The journal will serve as a comprehensiveresource for policy makers, government officials, academicians, and practitioners. GJACST promotes and coordinates the developments in the IT based applications of business management and presents thestrategic roles of IT and management towards sustainable development.

Global Journal of Advanced Computer Science and Technology

Global Journal of Advanced Computer Science and Technology

(Volume No. 13, Issue No. 1, January - April 2025)

Contents

An Efficient Automated Attendance Entering System by Eliminating Counterfeit Signatures using Kolmogorov Smirnov Test

K.M.L.P. Weerasinghe α & B.H. Sudantha σ
Department of Information Technology Sri Lanka Institute of Information Technology, Malabe, Sri Lanka.
Department of Information Technology University of Moratuwa, Sri Lanka.

ABSTRACT

Maintaining the attendance database of thousands of students has become a tedious task in the universities in Sri Lanka. This paper comprises of 3 phases: signature extraction, signature recognition, and signature verification to automate the process. We applied necessary image processing techniques, and extracted useful features from each signature. Support Vector Machine (SVM), multiclass Support Vector Machine and Kolmogorov Smirnov test is used to signature classification, recognition, and verification respectively. The described method in this report represents an effective and accurate approach to automatic signature recognition and verification. It is capable of matching, classifying, and verifying the test signatures with the database of 83.33%, 100%, and 100% accuracy respectively.

Keywords: image processing, kolmogorov smirnov test, machine learning, support vector machine.

I. INTRODUCTION:

A ttendance records are very essential in the academic activities of universities. Almost all the universities in Sri Lanka, signatures of candidates are taken in lectures, practical sessions, during examinations, etc. to verify the presence of the real candidate.

The paper-based attendance sheet is passed in each session to take the signature of each student. For later evaluations, attendance records should enter to the excel sheets and entering student's attendance into the excel sheets for each of the subjects which are very crucial, time-consuming process.

Automated student's attendance entering system can be used to simplify this task. Many attempts are made to automate this process with success to a certain extent. Many of these systems make use of sophisticated biometric equipment while some others use Barcodes and Radio Frequency Identity Cards [5]. Handwritten signatures considered the most natural method of authenticating a person's identity. However they are handled as images and recognized using computer vision and machine learning techniques. With modern computers, it is needed to develop fast algorithms for signature extraction, recognition, and verification. But even today the majorly used system is to take the signature of present candidates and then manually enter these records into the computer. In this study, the process has automated by developing a system which uses image processing techniques to update the attendance records automatically. To build up such a system signature extraction, recognition and verification are essential. Another task is the identification of counterfeit signatures. If we count the number of signatures and the number of heads during a lecture, practical session or in examinations, they should be the same. But sometimes some students sign for their colleagues or replaced by other students.

Therefore, identification of counterfeit signatures is very much essential in this type of situations. The handwritten signature is a prevalent way of authenticity. Despite its known weaknesses, and development of cryptographic and biometric techniques, it is still the most commonly used way of authentication when dealing with paper documents and forms. In this thesis, we focus on the application of biometric recognition for automatic student authentication, in particular making use of handwritten signatures, which are one of the most socially accepted biometric traits.

In education, signatures are used for attendance control, either to lectures or exams, but not for authentication. With the rapid deployment of dynamic signature extraction, recognition and verification automated students attendance entering system has been used for student authentication. Also, the use of this technology can be extended to different administrative services within the education system to add a higher security level to the traditional procedures of authentication.

Signature matching has been used in areas such as extraction [4], recognition [16] and verification [7]. While signature extraction aims to find document images that contain signatures [4], and signature recognition tries to find the corresponding signer of a test sample given a database of signature exemplars from different signers [2], signature verification deals with confirming the authenticity of a signature, i.e., decides whether a sample signature is genuine or forgery by comparing it with stored reference signatures. From the viewpoint of automating the attendance entering system, it involves machine learning from a population of signatures. In this study person, dependent learning is used in signature verification phase, so that there are only genuine signatures in the database. This technique is called special-learning. special-learning, a person’s signature is learnt from multiple samples of only that person’s signature, where within-person similarities are learnt to identify the signature is genuine or counterfeit.

The rest of this article is organized as follows. Section II mentioned the current achievements in this domain. Section III gives a general description of the proposed method. In section IV, discuss the results and discussions. In section V, provides the conclusions.

II. Literature Review

Ritesh Banka [1] has presented a new approach for the extraction of signature and handwritten regions from official binary document images. He proposed a new two-level scale invariant classification technique to extract the gray-scale handwritten area from the scanned document.

Ogul and coworkers [4] described a discriminative framework to extract the signature from a bank service application document. This is based on the classification of segmented image regions using a set of representative features. The segmentation is done using a two-phase connected component labeling approach.

Then evaluate solely and combined effects of several feature representation schemes in distinguishing signature and non-signature segments over a Support Vector Machine classifier.

Gupta [6] has done a cursive signature extraction and verification. In his research, he presented a new approach, based on connected component analysis and geometric properties of labeled regions. Manesh [15] proposed a method to automatically identify the signature in the scanned document images using a simple region growing algorithm. Offline handwritten signature verification using ANN [10][13][17] was another concern on this research paper. Sisodia [17] implemented a Static Signature Verification System with four stages such as image preprocessing, feature extraction, classification and decision making. Classifier used an ANN with Error Back Propagation algorithm to attain the result. The relevant features used by the classification are centroid, length and width of the signature in the 200×100 pixels' image box, quadrant areas, one dimensional first and second derivatives of the image and global slant angle. Menu Bhatia [15] was used maximum horizontal and vertical histogram, the center of mass, normalized are of signature, aspect ratio, tri surface feature, six-fold surface feature and transition feature as the extracted features from the candidate signature.

In contrast to the previous research, some have also used HMM and Graphometric features [8][9] and conjunction with neural network and support vector machines [12]. Abdullah [3] proposes a new method for signature recognition using Delaunay triangulation. Rupali Mehra and coworkers [13] present Surf features and neural-fuzzy techniques based recognition of offline signatures system that is trained with lowresolution scanned signature images. Gautam [17] has used SIFT and Delaunay triangulation for image matching in their research.

Woods [5] considered image area, vertical center, and the horizontal center of the signature, maximum vertical projection, maximum horizontal projection, vertical projection peaks, horizontal projection peaks, number of edge points, number of cross-points and Hough transform for feature extraction of each signature. Extracted values of each signature images from the database of 150 are given to the feed forward neural network (trained using back propagation gradient descent learning).

Gulzar and coworkers [10] present neural network based recognition of offline handwritten signature system that is trained with low- resolution scanned signature images. And also Prashanth C.R. [21] presents DWT based offline signature verification using angular features (DOSVAF). The signature is resized, and Discrete Wavelet Transform (DWT) is applied on the blocks to extract the features.

Vahid Kiani [11] proposes a new method for signature verification using local Radon Transform. The proposed method uses Radon Transform locally as feature extractor and Support Vector Machine (SVM) as the classifier. The main idea is using Radon Transform locally for line segments detection and feature extraction, against using it globally. The advantages of the proposed method are robustness to noise, size invariance and shift invariance. Having used a dataset of 600 signatures from 20 Persian writers, and another dataset of 924 signatures from 22 English writers, their system achieved good results.

In paper [8] a system is introduced that uses only global features. A discrete random transform which is a sinograph is calculated for each binary signature image at the range of $0-360$, which is a function of the total pixel in the image and the intensity per given pixel calculated using non-overlapping beams per angle for X number of angles. Due to this periodicity, it is a shift, rotation, and scale invariant. An HMM is used to model each writer signature. The method achieves an AER of 18.4% for a set of 440 genuine signatures from 32 writers with 132 skilled forgeries. Support Vector Machines (SVMs) are machine learning algorithms that use a high dimensional feature space and estimate differences between classes of given data to generalize unseen data. The system in [15] uses global, directional and grid features of the signature and SVM for classification and verification. The database of 1320 signatures is used from 70 writers. 40 writers are used for training with each signing eight signatures thus a total of 320 signatures for training. For initial testing, the approach uses eight original signatures and eight forgeries and achieves FRR 2% and FAR 11% [15].

III. Methodology

The main steps of this research consist of

- Signature extraction from attendance sheets based on morphological operations
- Separate signature, non- signature area using binary Support Vector Machine
- Signature recognition by training Error-Correcting Output Codes multiclass model using SVM
- Signature verification using Kolmogorov Smirnov test

A software package, Matlab2016b is used for this procedure.

a) Signature Extraction from the Scanned Attendance Sheets

An important task in the automated processing of scanned attendance sheets is to extract the signatures. Here both signature and non-signature area are extracted and classify them to separate signature area. The main steps in the signature extraction process are represented by Fig.1.

The main steps are:

- Image Pre-processing
- Classification using binary SVM

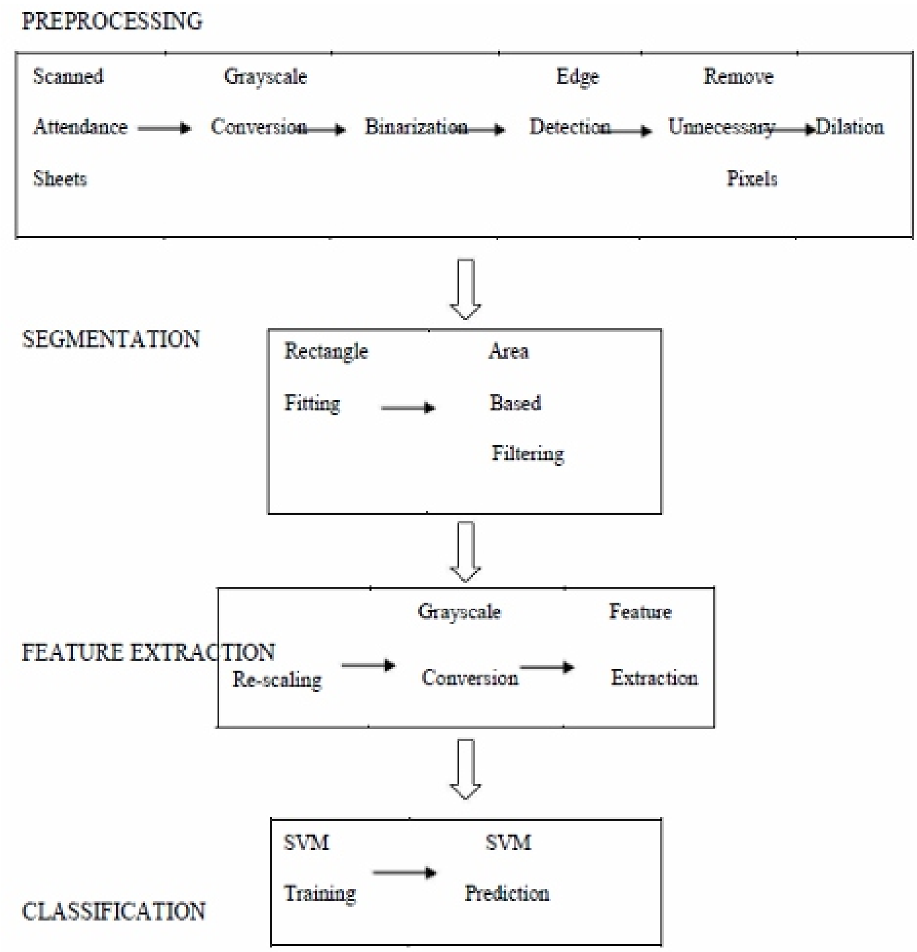


Fig.1:A brief outline of signature extraction

b) Image Pre-processing

In the pre-processing phase gray conversion, image binarization, edge detection, remove unnecessary pixels, morphological dilation (close, thicken, bridge), image segmentation and cropping are performed.

Fig. 2: Apply Pre-processing Techniques for Scanned Attendance Sheets

c) Classification using Binary SVM

After extracting the images from attendance sheets, the segmented images should be classified using a set of representative features. Here used features of segmented images in distinguishing signature and non-signature segments over a binary SVM classifier.

d) Signature Recognition

Signature recognition is a writer identification problem, whose objective is to find the author of a test signature given a database of signature exemplars from different signers. They are composed of special characters and flourishes, and therefore most of the time they can be unreadable. Also, intrapersonal variations and interpersonal differences make it necessary to analyze them as complete images and not as letters and words put together [15]. The signature acquisition, Pre-processing of signatures, Feature extraction, Train and test an Error Correcting Output Codes (ECOC) multiclass model using SVM are the main steps of signature recognition.

e) Signature Acquisition

Handwritten signatures are taken from 103 students who followed Statistical Inference-I course module in semester-I of academic year 2014/2015, in Faculty of Applied Sciences, Wayamba University of Sri Lanka. Seven signatures are taken from each student so that 721 are used in signature recognition phase.

Fig. 3: Sample Signature Database

f) Pre-processing of Signatures

Gray conversion, Image binarization, Remove unnecessary pixels, Thinning and Data area cropping are performed in the pre-processing phase:

Fig. 4: Apply Pre-processing Techniques for Signatures

g) Feature Extraction of Signatures

Before the feature extraction process to increase the accuracy of the system signature image is partitioned into four equal parts and extract features from each part. So, that the number of features which can be used to train the model has been increased.

Fig. 5: Partitioned Signature into 4 Parts

The choice of the features that provided to the classifiers of the system is essential. In this work, global and local features are used. Pure width, pure height, baseline shift, kurtosis, skewness, maximum vertical projection, maximum horizontal projection, vertical center of mass and horizontal center of mass, Hough transform, etc. are used as global features.

To increase the accuracy of the system grid based features are also extracted from the handwritten signatures as local features. Here Histogram Orient Gradient (HOG) features are extracted as grid features and combine them with global features in the recognition process. The total number of extracted HOG features is 2592.

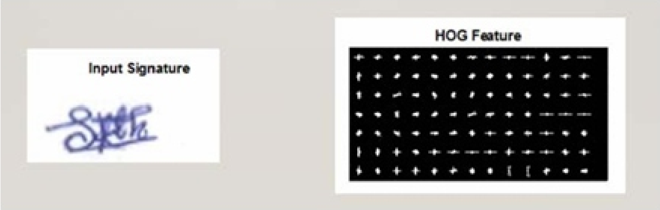


Fig. 6: HOG features extracted from one signature

Local and global features of images are given to error-correcting output code multiclass model (ECOC) to train it, so that particular signature image is recognized. ECOC classification requires a coding design, which determines the classes that the binary learners train on, and a decoding scheme, which ensures how the results (predictions) of the binary classifiers are aggregated. Suppose that there are three classes, the coding design is one vs. one, the decoding scheme uses loss g, and the learners are SVMs. To build the classification model, ECOC follows following steps.

A one vs. one coding design is:

	Class 1	Class 2	Class 3
Learner 1	1	1	0
Learner 2	-1	0	1
Learner 3	0	-1	-1

A new observation is assigned to the class () that minimizes the aggregation of the losses for the L binary learners. That is,

$$\hat{k} = \underset{k}{\operatorname{argmin}} \frac{\sum_{l=1}^{L} |m_{kl}|\, g(m_{kl}, s_l)}{\sum_{l=1}^{L} |m_{kl}|}$$

ECOC models can be used to improve the classification accuracy, even compared to other multiclass models.

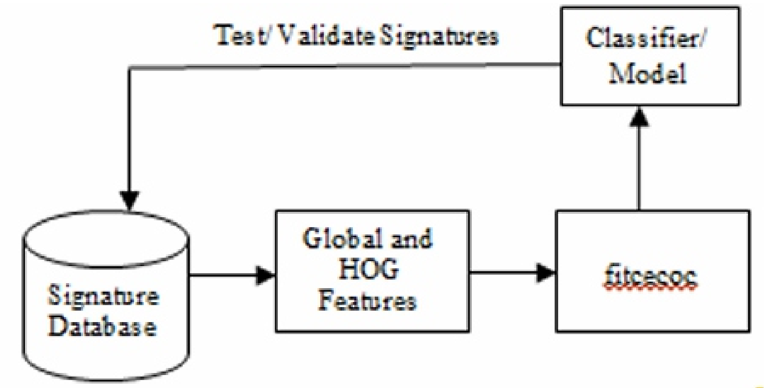


Fig. 7: Signature Recognition using Multiclass SVM

h) Signature Verification

The performance task of signature verification process is one of determining whether a questioned signature is genuine or not. The image of a questioned signature is matched against multiple images of known signatures. Visual signature verification is naturally formulated as a machine learning task. The machine learning tasks can be stated as general learning (which is person-independent) or special learning (which is person-dependent), paralleling the learning tasks of the human questioned document examiner. In the case of general learning the goal is to learn from a large population of genuine and forged signature samples. The focus is on differentiating between genuine-genuine differences and genuine-forgery differences. Special learning focuses on learning from genuine samples of a particular person. The focus is on learning the differences between members of the class of genuine. The verification task is a one-class problem of determining whether the questioned signature belongs to that class or not. Using Kolmogorov Smirnov test correctly classified signatures are used to confirm the genuineness.

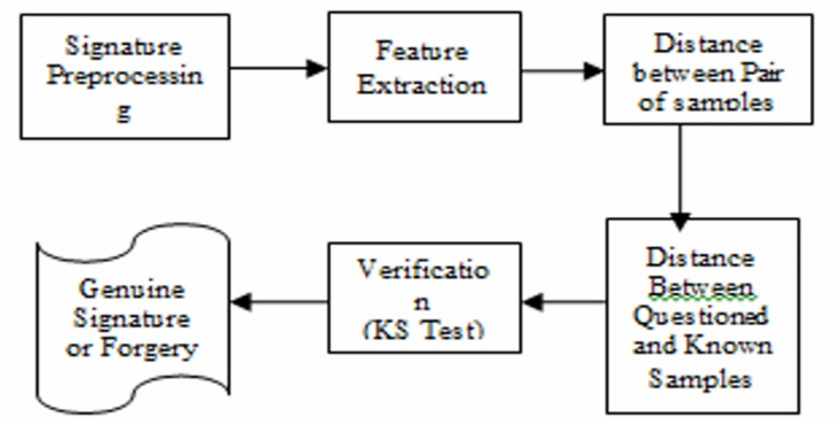


Fig. 8: Signature Verification using Kolmogorov Smirnov Test

If a given person has N samples, 2 �defined as N! /N! (N−r)! pairs of samples can be compared as shown in Fig.8. Let N be the total number of samples and NWD = 2 � be the total number of comparisons that can be made which also equals the length of the withinperson distribution vector. The within-person distribution can be written as

$$D_W = (d_1, d_2, \ldots, d_{NWD})^T$$

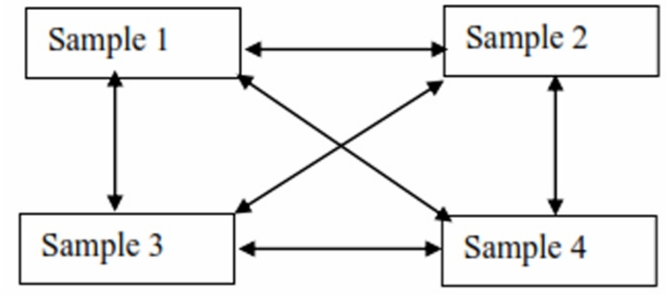


Fig. 9: Comparing All Possible Genuine-Genuine Pairs

Analogous to this, the questioned sample signatures can be compared with every one of the N knowns similarly to obtain the questioned vs. known distribution. The questioned vs. the known distribution is given by

$$D_{QK} = (d_1, d_2, \ldots, d_N)^T$$

where dj is the distance between the questioned sample and the jth known sample, j ∈ {1,..., N}. I) Performance Evaluation For unmatched signatures distance statistics is large and for matched signatures distance statistic is small. Since .004 is less than .20, the null hypothesis has been accepted. That is distributions are approximately same for matched signatures. In this study 0.01 has taken as the significance level.

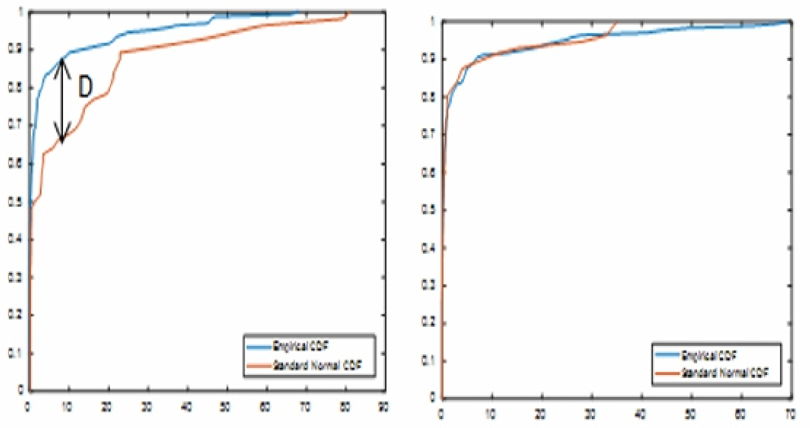


Fig. 10: Test Statistic for Unmatched and Matched Signatures

ALGORITHM:

Input = Signature image

Output = Conformation from system whether the signature is genuine or counterfeit.

Step 1: Acquire matched signature images from the signature recognition process

Step 2: Enhanced the signature images by preprocessing

Step 3: Create a feature vector by combining extracted features from the pre-processed signature images.

Step 4: Obtain the distances of features between every seven samples of the known signature in the database. (Results gave 21 × 8 matrices)

Step 5: Same has been done between the known sample and questioned sample. (Results gave 7 × 8)

Step 6: Apply KS test for two distributions and obtain the probability of similarity.

Step 7: Repeat step 1-7 to test all the signatures recognized by the system.

Step 8: If the probability is less than 0.01 the signature is identified as “Forge”, otherwise as “Genuine”.

Table 1: Sample Distance Distribution of Known Signature

1	2	3	4	5	6	7
0.44	0.68	0.52	0.87	1.01	0.81	0.93
1.67	2.44	1.93	2.95	3.27	2.80	3.09
122	98	139	67	38	69	43
0.27	0.54	0.67	0.08	0.30	0.40	0.28
0.048	0.07	0.05	0.1	0.12	0.09	0.10
0.10	0.16	0.12	0.21	0.24	0.19	0.22
5	7	9	7	8	5	5
0.03	0.06	0.04	0.08	0.09	0.07	0.08

Table 2: Sample Distance Distribution of Known vs. Questioned

1	2	3	4		19	20	21
0.24	0.08	0.43	0.57	…	0.19	0.07	0.11
0.77	0.26	1.27	1.59	…	0.46	0.18	0.28
24	17	55	84	…	31	5	26
0.26	0.39	0.18	0.03	…	0.09	0.01	0.11
0.03	0.00	0.05	0.07	…	0.03	0.01	0.01
0.06	0.02	0.11	0.14	…	0.05	0.02	0.03
2	4	2	3	…	3	3	0
0.02	0.00	0.04	0.05	…	0.01	0.00	0.01

j) Enter Data to Excel Sheets

After identifying whether a particular signature is genuine or forge the attendance records has been entered to the excel sheets. If the KS test identify the signature as genuine in verification process, '1' has been entered in front of the relevant student index number in the excel sheet.

IV. Results & Discussion

When extract the signatures from scanned attendance sheets in some situations there are some discontinuities in the signatures. In those situations, whole signature is not including in the bounded region as following figure (one signature is separated into parts).

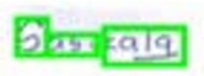


Fig. 11: Signature Image with Multiple Bounded Regions due to Discontinuity

To overcome that problem edge detection, morphological dilation, thicken and bridge has been used in pre-processing stage.

Fig. 12: Signature Image after Removing the Discontinuity

The errors we are trying to minimize in recognition and verification are: classifying one person's signature as belonging to another one and acceptance of a fake signature. Some signatures are misclassified by another student's signature due to some similarities between two signatures in the recognition process.

Fig. 13: Misclassification of a Signature

V. Conclusion

Today information technology has proved that there is a need to retrieve, search, query and store large amount of electronic information efficiently and accurately. In signature extraction phase, the whole part of the signature not extracted due to some reasons. Those are: excessive dusty noise, logos, figures, printed and handwritten text etc., large ink- blobs joining disjoint characters or components, degradation of printed text due to poor quality of paper and ink, text overlapping the signature. By increasing the space between text and signatures, it could be avoided the overlapping of signatures with text. Proposed system is extracted signatures with 100% accuracy.

In recognition process the combination of global and local features are used to train the ECOC multiclass model using SVM. The accuracy is 83.33% and it suggest that the use of gradient-based feature sets with global features can serve the most reliable way of detecting signatures in signature recognition process.

A machine learning approach is used in signature verification process, because only the genuine signatures are in the registered student database. All signatures are identified as genuine with 100% accuracy. Finally, we can conclude that this system can be used in a university educational environment for automatic student authentication. Eventually, based on the methodologies employed in this report, it provides a promising stage for the development of an automated attendance entering system.

References Références Referencias

1. *Banka, R., Nourbakhsh, F., 2010. Extraction of Signature and Handwritten Regions from Official Binary document Images.*

2. *Bhattacharyya, D., Kim, T., 2010. Design of artificial neural network for handwritten signature recognition. Int. J. Comput. Commun. 4, 59–66.*

3. *Coetzer, J., 2005. Off-line signature verification. Stellenbosch: University of Stellenbosch.*

4. *Cüceloğlu, İ., Oğul, H., 2014. Detecting handwritten signatures in scanned documents, in: Proceedings of the 19th Computer Vision Winter Workshop. pp. 89–94.*

5. *Gonzalez, R.C., Woods, R.E., 2002. Digital image processing. Pearson Education, Delhi, India.*

6. *Gupta, C.S., Dixit, U.D., 2015. A Review on Signature Detection and Signature Based Document Image Retrieval 4.*

7. *Hafemann, L.G., Sabourin, R., Oliveira, L.S., 2015a. Offline handwritten signature verification-literature review.*

8. *Kalera, M.K., Srihari, S., Xu, A., 2004a. Offline signature verification and identification using distance statistics. Int. J. Pattern Recognit. Artif. Intell. 18, 1339–1360.*

9. *Karouni, A., Daya, B., Bahlak, S., 2011. Offline signature recognition using neural networks approach. Procedia Comput. Sci. 3, 155–161. 2010.12.027*

10. *Khuwaja, G.A., Laghari, M.S., 2011. Offline handwritten signature recognition. World Acad. Sci. Eng. Technol. 59, 1300–1303.*

11. *Kiani, V., Pourreza, R., Pourreza, H.R., 2009. Offline signature verification using local radon transform and support vector machines. Int. J. Image Process. 3, 184–194.*

12. *Kumar, P., Singh, S., Garg, A., Prabhat, N., 2013a. Hand written signature recognition & verification using neural network. Int. J. Adv. Res. Comput. Sci. Softw. Eng. 3.*

13. *Mehra, R., Gangwar, R.C., 2014. A Survey: Enhanced Offline Signature Recognition Using Neuro-Fuzzy and SURF Features Techniques Vol 5(3), 4350–4353.*

14. *Nguyen, V., Blumenstein, M., Muthukkumarasamy, V., Leedham, G., 2007. Off-line signature verification using enhanced modified direction features in conjunction with neural classifiers and support vector machines, in: Ninth International Conference on Document Analysis and Recognition (ICDAR 2007). IEEE, pp. 734–738.*

15. *Özgündüz, E., Şentürk, T., Karslıgil, M.E., 2005. Offline signature verification and recognition by support vector machine, in: Signal Processing Conference, 2005 13th European. IEEE, pp. 1–4.*

16. *Piyush Shanker, A., Rajagopalan, A.N., 2007. Offline signature verification using DTW. Pattern Recognit. Lett. 28, 1407–1414.*

17. *Pushpalatha, N., Gautam, A., 2014. Offline signature Verification using spatial domain feature sets and support vector machine. Int. J. Emerg. Technol. Adv. Eng. 4, 544.*

18. Shirdhonkar, M.S., Kokare, M.B., 2010. Discrimination between printed and handwritten text in documents. IJCA Spec. Issue on 2010.

19. Singh, N., Kaushal, S., 2015. International Journal of Engineering Sciences & Research Technology OffLine Signature Recognition Using Modified Neural Networks Approach.

20. Sisodia, K., Anand, S.M., 2009. Off-line handwritten signature verification using artificial neural network classifier. Int. J. Recent Trends Eng. 2, 205–207.

Computer Vision based Traffic Monitoring and Analyzing from On-Road Videos

T.M. Amir-Ul-Haque Bhuiyan α, Mrinmoy Das σ & Md. Shamim Reza Sajib ρ
Department of Computer Science & Engineering
Bangladesh University of Business & Technology (BUBT).

ABSTRACT

Traffic monitoring and traffic analysis is much needed to ensure a modern and convenient traffic system. However, it is a very challenging task as the traffic condition is dynamic which makes it quite impossible to maintain the traffic through traditional way. Designing a smart traffic system is also inevitable for the big and busy cities. In this paper, we propose a vision based traffic monitoring system that will help to maintain the traffic system smartly. We also generate an analysis of the traffic for a certain period, which will be helpful to design a smart and feasible traffic system for a busy city. In the proposed method, we use Haar feature based Adaboost classifier to detect vehicles from a video. We also count the number of vehicles appeared in the video utilizing two virtual detection lines (VDL). Detecting and counting vehicles by proposed method will provide an easy and cost effective solution for fruitful and operative traffic monitoring system along with information to design an efficient traffic model.

Index Terms: video based traffic monitoring, traffic surveillance, counting vehicles,

I. INTRODUCTION:

Vision based traffic surveillance system has become an active area of research interest over the past decade in the developed countries. This is a potential area of research as it has some significant applications. In Bangladesh traffic jam is very common an unpredictable. Due to limitation of work force, authorities are unable to find the reason and control this. It is of interest to digitally process and analyze these videos in real-time in order to extract reliable data on traffic flow and to detect traffic events. For example, because of such video analysis, traffic density in major arteries can be estimated and the least congested routes and travel time estimates can be computed. This information can be achieved by counting vehicles passing through the roads. By implementing this, the authority can have the information of traffic flow on a particular road on a particular time, measure rush and reduce the problem. This type of traffic surveillance system is becoming popular in many countries. The proposed method needs two major things to be implemented: i) Detecting and counting vehicles,

ii) Generate necessary information to be used to maintain the traffic. Many approaches have been introduced for vehicle detection. Some available methods uses lidar, radar and computer vision. As camera is cheaper than radar or lidar, vision based vehicle detection and classification has become more popular than lidar or radar based detection system. Though computational power has increased dramatically, vehicle detection and classification is not an easy task. The problem is the dynamic environment of the road. The condition of the road cannot be predicted. There can be many human made infrastructures; pedestrians, which makes this, task a difficult one. In addition, there are change in background, illusion and heterogeneity of vehicles.

II. Related Works

Background subtraction based method is used by [1] for detection. Deep neural network based detection and classification model [2] is very expensive in terms of computational resources and time and is not suitable in real time. Niluthpalet. al. [3] generates time spatial images from video frames and gain a very good speed and accuracy in detection. Vehicle detection approaches can be divided into two broad categories: appearance based and motion based methods.

Camera placement plays a significant role in video-based vehicle detection. Camera can be moving or static. As occlusion is the main problem in vision based detection system, camera should be placed in some position that minimizes the probability of occlusion. Camera placement depends on the appearance of the vehicles. In [3] camera is placed in an over bridge for taking both incoming and outgoing vehicles. It takes both front and rear view of vehicles. Broggi et al. [4] placed a camera to capture the side view of vehicles. Sivram and Trevedi [5] mounted camera in front of the moving vehicles that capture the rear view of front side vehicles. For static camera, good choice is some higher position than the level of vehicles that reduces the chances of occlusion. This decreases the chance of partial occlusion caused by vehicles but vehicles appearance changes from first lane to third len or fourth len. Yong Tang [6] placed the camera in a high position like over bridge and capture the front view of the vehicles To detect vehicles from a frame, features are extracted from the frames. Many types of features have been introduced for vehicle detection so far. Sivram and Trevedi [7] used edge features to highlight the side of a vehicle and cast shadow. In recent years, some strong features due to the robustness and reliability replace simple features like edges or corners. These features are commonly used in computer vision for both detection and classification. Histogram of oriented gradient (HOG) was extremely well represented for vehicle detection as well as object detection. Inspired by human detection method [8] of Dalal and Triggs [9] has used HOG features nicely to detect vehicles. At first they compute gradients from the images and extract HOG features. A linear SVM classifier trains the extracted features. Though HOG features provide a very good detection rate, the main drawback is its calculation speed. As classification will be done after detection, its speed should be good. Haar-like features have been used nicely for face detection. Haar-like features are calculated with the help of integral image. Integral image can be calculated at a very fast speed. Due to its calculation speed and successful use in face detection, it has been also used for vehicle detection in [10] and [11] successfully. Scale invariant feature transform (SIFT) [12] was used in [13] to detect rear faces of vehicles. Though this feature cannot provide better performance than HOG or Haar, it is considerably good in case of occlusion. Lin et al. [14] used a combination of SURF [15] and edge features to detect vehicles in the blind spot.

Support vector machine (SVM) is a strong binary classifier. It has been widely used for vehicle detection. The combination of HOG features and SVM classifier have been used for vehicle detection in [9] and [16].

AdaBoost [17] is also widely used in real time vehicle detection. As the classification speed of AdaBoost classifiers is high, it has become popular in real time classification applications. A combination of Haar-like features and AdaBoost classification is used in [18] and [19] for vehicle's rear face detection and perform very good in real time. The purpose of the AdaBoost algorithm is to use the feature to discover the best weak classifiers to form a strong classifier, and has shown its capability to improve the performance of various detection and classification applications.

Actually the strong classifier is an ensemble classifier composed of many weak classifiers that just better than a random guess. Tang et al. [6] also used this method successfully. This method is very fast and provide high accuracy. However, the main drawback of this method is high false positive rate. The method proposed in [18] and [19] achieve a accuracy of 98% with 3%-5% false positive rate which is not tolerable in these type of applications. Yong Tang [6] used this method and achieved good accuracy but had a false positive rate of 3%. This method is very fast and applicable in real time.

But false positive rate can cause inapplicable in some sectors. In recent year deep neural network and model, based classification is being used to detect 3D vehicles.

Both need high computational resources and execution time. Researchers have done some motion-based approaches. In [4], an adaptive background model was constructed, with vehicles detected based on motion that differentiated them from the background. Adaptive background modeling was also used in [20], especially to model the area where overtaking vehicles tend to appear in the cameras end of view. In [3] they used three virtual detection lines and generate time spatial image (TSI) from three frames. The vehicles present in a time spatial image is called TSI object blobs (TOBs). Then canny edges of TOB is generated. After that binary masks of the TOBs are obtained. Then vehicles are detected from multiple TOBs. This method generates a very good result with a good calculation speed and applicable in real time application. However, it is not suitable in conditions where there is heavy rush on the road and vehicles are moving in a low speed.

After detecting the vehicles, we count the total number of vehicles. Bas, Erhan and Tekalp [21] proposed adaptive background subtraction and Kalman filtering based model to count vehicle. However, their method provides a very good accuracy but it is comparatively slow to compute.

Unzueta, Luis, et al. [22] propose a robust adaptive multi cue segmentation strategy that detects foreground pixels corresponding to moving and stopped vehicles, even with noisy images due to compression. This method is also reliable and provides satisfactory result in various conditions.

III. Proposed Method

Our method is proposed targeting real time applications. Therefore, the execution time must be faster in both detection, counting and analysis. We are proposing a cost effective and faster model that can easily be implemented for real time traffic surveillance. In the next sub sections, we will discuss about the approaches and methods we use for detection and counting vehicles from a video and then generate the information about traffic.

a) Detection

i. Camera placement: At the very first stage of detection, we have to select a suitable place to set camera. We propose our method targeting to run on videos taken by some static camera.

ii. Feature Extraction: As we propose our method for real time application, we need so select a feature that is fast to compute. In proposed method, haar-like features are chosen for vehicle detection.

The detection is done using Viola-Jones object detection framework. A window of the target size is moved over the input image, and for each subsection of the image, the Haarlike feature is calculated.

The detection is done using Viola-Jones object detection framework. A window of the target size is moved over the input image, and for each subsection of the image, the Haarlike feature is calculated.

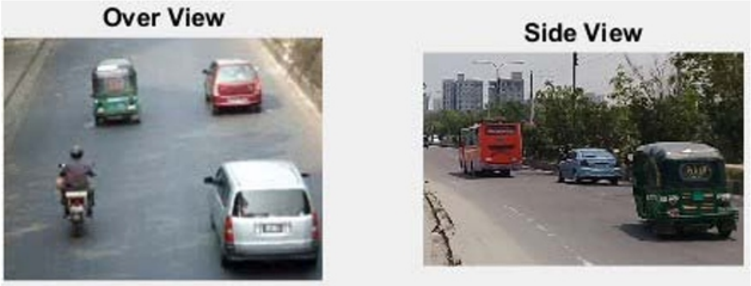


Fig. 1: Left is the view from over bridge and right side shows the view from side

iii. Classifier: SVM is faster in training stage than AdaBoost classifiers. SVM becomes quite slower in test stage. We need our method to perform faster in testing. Some weak classifiers are trained in AdaBoost learning algorithm and then combine them to make a strong classifier. A cascade of 25 classifier is made in proposed method. The first two classifier

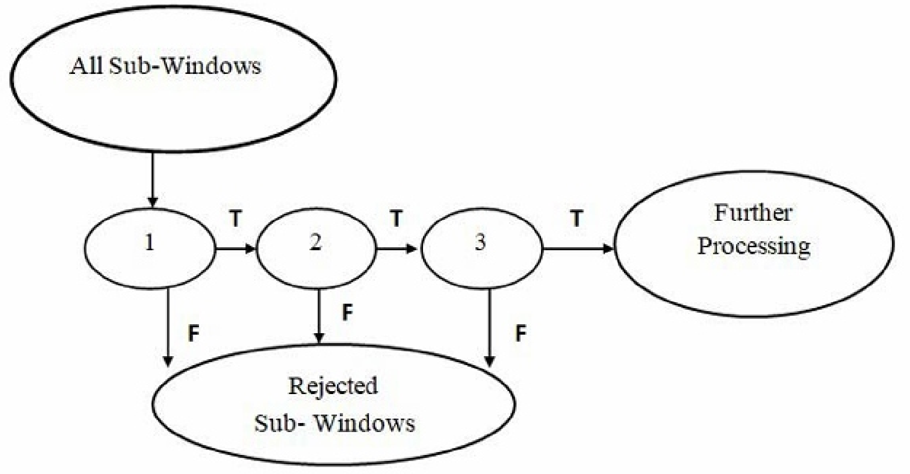


Fig. 2: Testing process of AdaBoost Classifiers

is the strongest one because it is made from the strongest feature selected in AdaBoost feature selection process. These two classifier can produce a 100% detection rate with a 50% false positive rate. As the number of classifier increases, the false positive rate becomes lower. For training the classifiers, we take 5000 images as positive that contains vehicles and 2000 negative images. After training, we get a cascade of 25 classifiers. In testing stage, a sub window of size 40x40 is moved over the input image of size 640x360. The sub windows that become positive after classifying by the first classifier is set as input for the second classifier.

The negative sub-windows are discarded in each step. Sub-window that remains positive after going through some steps are said to be positive and the area covering by some joined sub-window is detected as vehicles.

b) Counting

As a vehicle can appear in multiple frames, it will be detected multiple times also. However, we have to count only once. Therefore, we cannot count each vehicle of all the frames. To count a vehicle only once, we use two virtual detection lines (VDL). The classifier only computes the region of the virtual detection line. So when vehicles appear on the region within virtual detection lines, only then they are detected. When a vehicle appears in the VDL its full rear view must be seen

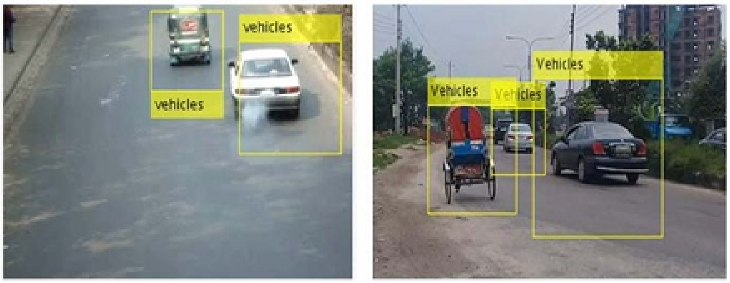


Fig. 3: Left: Detected vehicles from video taken from over bridge, Right: Detected vehicles from a video taken from the side of the road

Fig. 4: Two virtual detection lines of a frame. Arrow indicates the flow of traffic and

for this no chance of false detection. As a vehicle must appear within the region of VDL, there will be no possibility to miss any vehicle. Still there is possibility of counting a vehicle more than once. Therefore, we count vehicle after each three frames. The length of the VDL is set in such a way that no vehicle can go past the VDL within these three frames.

The difference of the two VDL is the length a vehicle can cover within five frames at an average of 40km/h. Therefore, the faster vehicle like bus or car cannot disappear within five frames and the slower vehicle like rickshaw won't appear twice within the frame as we count after each three frames. This provides a very good accuracy in counting.

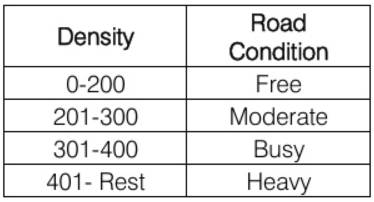
c) Classifying Road Condition

After counting the vehicles passing through a road, we generate some useful information that can be used to operate traffic smartly and design an efficient traffic model. We experiment the proposed method in the videos taken from a road then we count the number of vehicles passing through that road in a certain amount of time. Suppose there is a busy road where a large number of vehicles go through within 9.00 am to 9.30 am. However, this road can be free at another time.

$$Density = \frac{NumberofVehicles}{NumberofLaneoftheRoad}$$

The value number of vehicles is found by counting vehicles in every 20 minutes. If the density is less than or equal 200, then we say the condition of the road is free. If it is between 200 and 300 then the condition of the road is moderate. If it is between 300 and 400 then the road is busy and after that, the condition of the traffic of the road is heavy.

Table I: Classification of Road Condition by Density



Density	Road Condition
0-200	Free
201-300	Moderate
301-400	Busy
401- Rest	Heavy

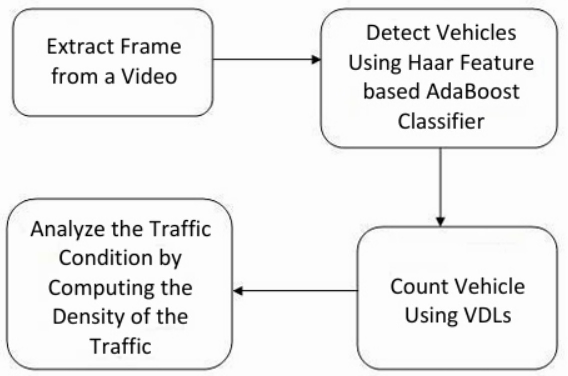


Fig. 5: System Architecture of the Proposed Method

IV. Experimental Result

Multiple experiments have been done to generate the result of the method. All the experiments have been done on different videos.

a) Data Set Collection

There are 15 videos taken from Kalshi Road, Mirpur, Dhaka, Bangladesh from the side of the roads. Videos are taken on different environments including sunny, cloudy weather in different times. Some videos are taken in and opposite direction of sunlight. Another 10 videos taken by MVDL [3] authors is used to test the proposed method.

For training we take two instances of each image thus the positive image set for detection contains almost 5000 images of vehicles. For negative image set, we use 1300 negative images of Caltech car data set and 700 images of local roads. The negative image set contains almost 2000 images. Those images are resized and used for training.

b) Result

Extensive experiments have been carried out to generate the result of the proposed method. Without considering occlusion, the average detection rate is 97.81%. Average false positive rate is reduced to 1.8%. The accuracy of counting is also satisfactory. The proposed method provides 93.11% counting accuracy. The Roads where

Table II: Collected Data Set Description

Place	No. of Videos	Position	Time	Duration (Mins)	Environment
Kalshi Road, Mirpur, Dhaka	4	Side	9.00 AM	6.36	Sunny
Kalshi Road, Mirpur, Dhaka	4	Side	12.00 PM	6.23	Cloudy
Kalshi Road, Mirpur, Dhaka	5	Side	4.00 PM	9.25	Sunny
Kalshi Road, Mirpur, Dhaka	2	Side	2.00 PM	5.17	Sunny
Cantonment Fly Over Road, Dhaka	3	Side	11.00 AM	3.11	Against Sun-light
Shahbagh, Dhaka	5	Over Bridge	10.00 AM	22.54	Sunny
Shahbagh, Dhaka	3	Over Bridge	2.00 PM	12.21	Cloudy
Shahbagh, Dhaka	2	Over Bridge	4.00 PM	9.46	Partially Sunny
BUET, Dhaka	2	Over Bridge	10.00 PM	8.35	Sunny
BUET, Dhaka	2	Over Bridge	12.00 PM	4.49	Cloudy

Table III: Detection Accuracy of Proposed Method for Different Data Set

Data Set	Detection Accuracy	False Positive Rate
Data Set 1	97.81%	1.25%
MVDL [3](Data Set 2)	97.23%	1.67%
Zhang [1]	97.37%	2.5%

rickshaws are plenty, there the accuracy of the method decreases. The rickshaws are very slow compared to others vehicle and for this one rickshaw may appear twice in the VDL region. We can see the result of the video taken from BUET. Most of the vehicles there are rickshaws and the total count is greater than the actual count.

After counting the vehicles for a particular time (20 minutes), the proposed method provides the information about the condition of the roads. The results are generated by calculating the density. We have carried out experiments in our collected videos taken from different place of Dhaka city. We find that the method provides accurate result about the condition of the roads.

V. Conclusion

Traffic monitoring by analysing videos of the road is a very challenging task. It is more difficult on the busy roads like

Table IV: Counting Accuracy by Proposed Method

Road Name	Duration of Video (Mins)	Actual Count	Count by Proposed Method	Accuracy
Kalshi Road	6.23	288	270	93.75%
Kalshi Road	9.25	396	375	94.70%
Matikata Overbridge	3.23	114	105	92.10%
Shahbagh	5.52	244	231	94.64%
BUET	4.49	87	96	89.65%

Table V: Traffic Condition of Some Roads in Different Time

Road Name	Duration of Video (Mins)	Time	Traffic Condition
Kalshi Road	6.23	12.00 PM	Moderate
Kalshi Road	20.21	9.00 AM	Heavy
Matikata Over-bridge	3.23	12.00 PM	Free
Shahbagh	5.52	2.00 PM	Busy
Shahbagh	5.52	4.00 PM	Heavy

the roads of Dhaka city as the density of the traffic is very high in most of the time. The proposed method can provide a solution to this problem as it is cheap and easy to implement. The proposed method uses Haarlike feature based Adaboost classifier that is faster to compute and provide a very good accuracy in detection. Two virtual detection line (VDL) is used to count the vehicle. The difference of the two VDL is set in such a way that minimizes the chance of missing or counting twice. Although the obtained results are promising, the algorithm still needs further modifications. Because the accuracy of the method decreases when there is so much rush on the roads. it also provides a dissatisfactory result on the roads where rickshaws are plenty. The reason behind it is occlusion of the traffic. As the rickshaw pullers do not develop a habit of maintaining lanes, it causes occlusion. Heavy traffic condition also causes occlusion. A sufficient number of experiments have been carried out on different data set. Experimental results demonstrate that the proposed method provides an acceptable and satisfactory result in counting vehicles and classifying road condition in terms of accuracy, robustness and execution time. References

References Références Referencias

1. GuohuiZhang, Ryan P. Avery, Yinhai Wang, "A Video-based Vehicle Detection and Classification System for Real-time Traffic Data Collection Using Uncalibrated Video Cameras," TRB 2007 annual meeting.

2. G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955.

3. Niluthpol Chowdhury Mithun, Nafi Ur Rashid, S. M. Mahbubur Rahman "Detection and Classication of Vehicles from Video Using Multiple Time-Spatial Images," IEEE transactions on intelligent trasportation systems, vol. 13, NO. 3, September 2012.

4. A. Broggi, A. Cappalunga, S. Cattani, and P. Zani, "Lateral vehicles detection using monocular high resolution cameras on TerraMax," in Proc. IEEE Intell. Veh. Symp., Jun. 2008, pp. 11431148.

5. SayananSivarama, Mohan Manubhai Trivedi, "Integrated Lane and Vehicle Detection, Localization, and Tracking: A Synergistic Approach," IEEE transactions on intelligent trasportation systems, vol. 14, no. 2, June 2013.

6. Yong Tang, Congzhe Zhang, RenshuGu, Peng Li, and Bin Yang. Vehicle detection and recognition for intelligent traffic surveillance system. Multimedia tools and applications, 76(4):5817-5832, 2017

7. SayananSivarama, Mohan Manubhai Trivedi, "Deep Nural Network," IEEE transactions on intelligent trasportation systems, vol. 14, no. 2, June 2013.

8. NavneetDalal and Bill Triggs. Histograms of oriented gradients for human detection. In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, volume 1, pages 886-893. IEEE, 2005.

9. S. Teoh and T. Brunl, "Symmetry-based monocular vehicle detection system. Mach," Vis. Appl. [Online]. 23(5), pp. 831842.

10. W.-C. Chang and C.-W. Cho "Real-time side vehicle tracking using parts-based boosting," IEEE Int. Conf. SMC, 2008, pp. 33703375. Year 2 01 24

11. W.Liu, X.Wen, B.Duan, H.Yuan, and N.Wang, "Rear vehicle detection and tracking for lane change assist," in Proc. IEEE Intell. Veh. Symp., Jun. 2007, pp. 252257.

12. D. Lowe "Object recognition from local scaleinvariant features," in Proc. Int. Conf. Comput. Vis., 1999, pp. 11501157.

13. X. Zhang, N. Zheng, Y. He, and F. Wang, "Vehicle detection using an extended hidden random eld model," in Proc. 14th Int. IEEE Conf. ITSC, Oct. 2011, pp. 15551559.

14. B.-F.Lin, Y.-M.Chan, L.-C.Fu, P.-Y.Hsiao, L.-A.Chuang, S.-S.Huang, and M.-F. Lo, "Integrating appearance and edge features for sedan vehicle detection in the blind-spot area," IEEE Trans. Intell. Transp. Syst., vol. 13, no. 2, pp. 737747, Jun. 2012.

15. Z.Chen, N.Pears, M.Freeman, and J.Austin, "Road vehicle classication using support vector

machines," in Proc. IEEE Int. Conf. Intell. Comput. Intell. Syst., Shanghai, China, 2009, pp. 214218.

16. SayananSivarama, Mohan Manubhai Trivedi, "Active learning for on road vehicle detection: A comparative study," Mach. Vis. Appl. Special Issue Car Navigation and Vehicle Systems, pp. 113, Dec. 2011.

17. Yoav Freund, Robert Schapire, and Naoki Abe. A short introduction to boosting. Journal-Japanese Society for Artificial Intelligence, 14(771780): 1612, 1999.

18. J. Cui, F. Liu, Z. Li, and Z. Jia, "Vehicle localization using a single camera," in Proc. IEEE IV, Jun. 2010, pp. 871876.

19. D. Withopf and B. Jahne, "Learning algorithm for real-time vehicle tracking," in Proc. IEEE ITSC, Sep. 2006, pp. 516521.

20. J. Wang, G. Bebis, and R. Miller, "Overtaking vehicle detection using dynamic and quasi-static background modeling" in Proc. IEEE CVPR, Jun. 2005, p. 64.

21. Bas, Erhan, A. Murat Tekalp, and F. Sibel Salman. "Automatic vehicle counting from video for traffic flow analysis." Intelligent Vehicles Symposium, 2007 IEEE. Ieee, 2007.

22. Unzueta, Luis, et al. Adaptive multicue background subtraction for robust vehicle counting and classification." IEEE Transactions on Intelligent Transportation Systems 13.2 (2012): 527-540.

Membrane Computing Aggregation (MCA): An Upgraded Framework for Transition P-Systems

Alberto Arteta α, Luis Fernando Mingo σ, Nuria Gomez ρ & Yanjun Zhao Ꮐ
Troy University, Computer Science.
Polytechnic University, Computer Science.

ABSTRACT

MCA (Membrane computing aggregation is experimental computational frame. It is inspired by the inner properties of membrane cells (Bio-inspired system). It is capable of problem solving activities by maintaining a special, "meaningful" relationship with the internal/external environment, integrating its self-reproduction processes within the information flow of incoming and outgoing signals. Because these problem solving capabilities, MCA admits a crucial evolutionary tuning by mutations and recombination of theoretical genetic "bridges in a so called "aggregation" process ruled by a hierarchical factor that enclosed those capabilities. Throughout the epigenetic capabilities and the cytoskeleton and cell adhesion functionalities, MCA model gain a complex population dynamics specifics and high scalability. Along its developmental process, it can differentiate into meaningful computational tissues and organs that respond to the conditions of the environment and therefore "solve" the morphogenetic/configurational problem. MCA, above all, represents the potential for a new computational paradigm inspired in the higher level processes of membrane cells, endowed with quasi universal processing capabilities beyond the possibilities of cellular automata of and agent processing models.

I. INTRODUCTION:

In spite of all the recent emphasis and advancements in systems biology, synthetic biology, and network science about modelling of gene networks, protein networks, metabolic and signaling networks, etc. some of the most important computational properties of membrane cells have not been grappled and "abstracted" et: scalability, tissular differentiation, and morphogenesis -i.e., the capability to informationally transcend the cellular level and organize higher level information processes by means of heterogeneous populations of membrane cells organized as "computational tissues and organs". Synthetic biology has become extraordinarily active in the manufacture of very simple and robust models and simulations tailored to the realization problems of circuits and modules in vivo, mostly addressed to prokaryotic systems. In the first wave of these studies, very basic elements such as promoters, transcription factors, and repressors were combined to form small modules with specified behaviors. Currently modules include switches, cascades, pulse generators oscillators, spatial patterns, and logic formulas (Purnick & Weiss, 2009). The second wave of synthetic biology is integrating basic parts and modules to create systemslevel circuitry. genomes and synthetic life organisms are envisioned, and application-oriented systems are contemplated. Different computational tools and programming abstractions are actively developed (the Registry of Standard Biological Parts; the Growing Point Language GLP; the Origami Shape Language OSL, the PROTO bio programming language, etc. See details at the Open Wetware site). Evolving cell models of prokaryotes have also been addressed (Cao et al., 2010). (Bashor et al., 2010). As some have put, "systems broaden the scope of synthetic biology designing synthetic circuits to operate in reliably in the context of differentiating and morphologically complex membrane cells present unique challenges and opportunities for progress in the field" (Haynes & Silver, 2009). However, very few synthetic biology researchers do contemplate using systems.

In systems biology, a plethora of modelling developments have been built around signaling pathways, cell cycle control, topologies of protein networks, transcriptional networks, etc. There is a relatively well consolidated thinking, in part due to traditional physiology and to systems science and control theory which were at the origins of this new field, of going "from genes to membrane cells to the whole organ" as D. Noble has done for heart models (Noble, 2002). The integration of proteins to organs has also been promoted by bioinformatic-related projects such as the "Physiome Project" (Hunter et al., 2002).

Important works have been done in the vicinity of "network science" in order to make sense of gene networks, protein networks, transcription networks, complexes formation, etc. For instance, about how is dynamically or2anized modularity in the yeast proteinprotein interaction network (Han et al., 2004), it was uncovered that two types of "hub" contribute to the organized modularity of the proteome: "party” hubs which interact with their partners simultaneously, and "date" hubs, which bind their different partners at different times and locations (we will see later on the importance of the discussion on "modularity" in the evodevo field). Predictive models of mammalian membrane cells have been described using graph theory, assembling networks and integrative procedures (Ma'yan et al., 2005). Important systems biology compilations and far-reaching cellular models have been made by Balazsi et al. (2005), Kitano (see in Oda et al., 2004), Luscombe et al. (2004), Huh et al., (2010) ...It has to be emphasized that concerning the views advocated in this proposal, most of systems biology works depart from the goal of "abstracting computational power out from systems" and focus instead on "applying computational power to analyze the organization of systems." Notwithstanding the foregoing, studies such as A. Dan chin (2009) on bacteria as computers making computers, and by Ray et al. (2010) on the operating system of bacteria could be considered as forerunners in the former direction.

In the science of development (the "evo-devo" discipline) most of the emphasis has been on modularity. What it exactly means in developmental terms is still a matter of controversy (Schlosser & Wagner, 2004; Carroll, 2005; Sprinzak, 2010); but undoubtedly modularity refers to the capability of cellular networks to dissociate networked processes at a lower level and to recombine or redeploy them at the higher level of the multicellular organism. Thanks to the cellular signaling system, the genetic switches, the cytoskeleton, and some other topobiological mechanisms (Edelman, 1988; Szathmary, 2001), the unitary network of cellular processes integrated into the cell-cycle may be broken down into coherent modules and be performed separately in different membrane cells within differently specialized tissues (Palmer, 2004). This implies a flexible organization for the deployment of biomolecular processing modules, which actually are "cut" differently in each tissue along the developmental process, due also to chromatin remodelling during development (Ho & Crabtree, 2010). Interestingly, not only differentiation but also morphology becomes an instance of the scalable "modular" processing, throughout the "tensegrity" emergent property and the ontogenetic arrangement of symmetry breakings in a force field. The emergence of cellular bauplans where signaling, force fields, and cytoskeletal mechanical modes conspire together to create but a few basic morphologies for membrane cells, depending also on the populations present, seems to be another important consequence (Mojica et al., 2009). Interestingly, complex morphologies obtained out from Turing diffusion model have been cogently discussed as a result of cell-to-cell developmental interactions (Kondo & Miura, 2010) .Currently, the evo-devo field accumulates a considerable mass of biomolecular-or2anizationfacts, poorly conceptualized yet, to be computationally "abstracted" in the perspective of MCA advancement. In the fields closer to computer science and Biocomputing, it has been important the introduction of the agent based approach (as pioneered by W. Fontana and others), which uses sets of rules to define relationships between cellular components substituting for the simple Boolean networks and differential equations used up to now. Proteins and other biomolecules become molecular "automata" and the aggregate behavior that emerges out from these models is the combinatorial expression of all those automata doing their specific micro-functions (Blow, 2009). This approach shows promise for "evolvable" advancement of network models endowed with the flexible modularity property. It is somehow close to the already mentioned predictive models of mammalian membrane cells that are using graph theory, assembling networks and integrative procedures (Mayan et al., 2005).

New generations of cellular models (of "automata") have been developed too, with powerful data content and with potential for modelling multi-cellular systems in a general way, supporting userfriendly in silicon experimentation and discovery of emergent properties (Amir-Kroll et al., 2008). Under the approach of Artificial Embryology, a developmental system has been obtained by means of cellular automata systems capable of following "rewriting rules" procedures, emulating elementary morphologies and multicellular distributions (Federici & Downing, 2006).

As for the developments in molecular Biocomputing, the idea that bio-molecules (DNA, RNA, proteins) might be used for computing already emerged in the fifties and was reconsidered periodically with more and more arguments which made it more viable. But the definitive confirmation came in 1994 (Adleman, 1994) when L. Adleman successfully accomplished the first experimental close connection between molecular biology and computer science. He described how a small instance of a computationally intractable problem might be solved via a massively parallel random search using molecular biology methods. An important part of this project is focusing on bio-inspired models of computation abstracted from the very complex networks in living systems. Its goal is to investigate several aspects of these models particularly focused on connections between theoretical models and natural (biological) networks. The main topics are: Computational aspects (computational power, structural and description complexity).

Application aspects (simulation, physical implementation, experimental results, training issues). This part is intended to be a contribution to both Global Computing (which includes neural networks, cellular automata, etc.) and Bio-inspired Computing (as a part of Natural Computing) a new and interdisciplinary field which lies at the crossroads of mathematics, computer science, molecular biology and linguistics. There are research groups working in similar or connected topics in Europe (Germany, France, Spain, Holland, Hungary, Romania, Moldavia, Finland, Poland, Austria, Italy), USA, Japan, India, China.

Several new directions of research have been initiated in the last decade: computing devices inspired from the genome evolution (Dassow & Mitrana, 1997; Dassow et al., 1997; Dassow et al., 2002), membrane systems (Nun, 2002) with an explosive development, evolutionary systems based on the behavior of cell populations (Ardelean et al.,2004) computing models simulating the process of gene assembly in ciliates (Ehrenfeucht et al., 2003), (Freund et al., 2002), (Istrail et al., 2007), networks of evolutionary processors (Manea et al., 2010), etc. The joint efforts of biologists and computer scientists led to a new concept, namely the template-guided recombination which seems to offer a "bioware" implementation of the process of gene assembly (Angeleska et al., 2007), (Presscot et al., 2003).

Swarm computation is mainly based on the same idea: a swarm is a group of mobile biological organisms wherein each individual communicates with others by acting on its local environment (Engelbrecht, 2005). A computational model based on multiset rewriting is used to simulate the emergence of autocatalytic cycles which are often found in living systems is proposed in (Suzuki&Tanaka, 1997). The use of X-machines, a variant of finite state machines with much more computational power, is used to model immunological pathways (Holcombe&Be11,1998). Moreover, (Istrail et al., 2007) proposes a new paradigm, "genomic computer", where the entire genomic computer", where the entire genomic regulatory system is viewed as a computational system and not only the immune system as it was considered in (Dasgupta,1998).

Many works were devoted to the study of a wide range of operations on biological sequences in vivo and in vitro (bio-operations): PA-matching, annealing, Watson-Crick superposition, transposition, inversion, duplication, translocation, etc. (Karp,2002) gives an overview of the most important and attractive problems for mathematicians coming from genomics and molecular biology. Last but not least, the molecular computing contributed to the understanding of selfassembly which is one of the key concepts in nanoscience (Reif&LaBean,2007).

The new sub-area of Computation Theory called Bio-Inspired Computing is very dynamic. After approximately 12 years the bibliography about BioInspired Computing counts nearly 1000 papers and several books and grows rapidly each year. These papers were published in either computer science forums or biological ones. Many prestigious international journals hosted special issues but new journals were also created: a permanent column in the Bulletin of the European Association for Theoretical Computer Science, Natural Computing, Journal of Unconventional Computing, Theoretical Computer Science-Track C, Theory of Natural Computing, etc. Each year several conferences are devoted mainly to this area: DNA Based Computers (15 editions so far), Unconventional Models of Computation (8 editions so far), Workshop in Membrane Computing (11 editions so far), International Work-Conference on Artificial and Natural Networks (9 editions so far), International WorkConference on the Interplay between Natural and Artificial Computation (5 editions so far), Pacific Symposium on Biocomputing (first edition in 1995). Regarding applicative models there are many attempts to update Cells computing paradigm in Arteta (2009) Arteta (2010) Arteta (2011) Arteta (2012) Arteta (2013) Arteta (2014), Frutos (2009) and Frutos (2013) among others.

II. Membrane Computing

A Transition P System of degree n 1 > n is a construct

$$\Pi = \left(V, \mu, \omega_1, \quad \omega_n, (R_1, \rho_1) \quad R_n, \rho_n\right) i_0)$$

Where:

V is an alphabet; its elements are called objects; μ is a membrane structure of degree n, with the membranes and the regions labeled in a one-to-one manner with elements in a given set; in this section we always use the labels 1, 2, n; n i i ≤ ≤ ω 1 , are strings from * V representing multisets over V associated with the regions 1,2,..,n of μ n i Ri ≤ ≤ 1 , are finite set of evolution rules over V associated with the regions 1,2,..,n of μ; i ρ is a partial order over n I Ri ≤ ≤ 1 , specifying a priority relation among rules of i R . An evolution rule is a pair (u,v) which we will usually write in the form v u→ where u is a string over V and v=v' or v=v'δwhere v' is a string over

$$(V \times \{here\ , out\ \}) \cup (V \times \{in_j\ 1 \leq\ j \leq\ n\}),$$

and δis a special symbol not in. The length of u is
called the radius of the rule v u→
io i is a number between 1 and n which specifies the output membrane of ∏

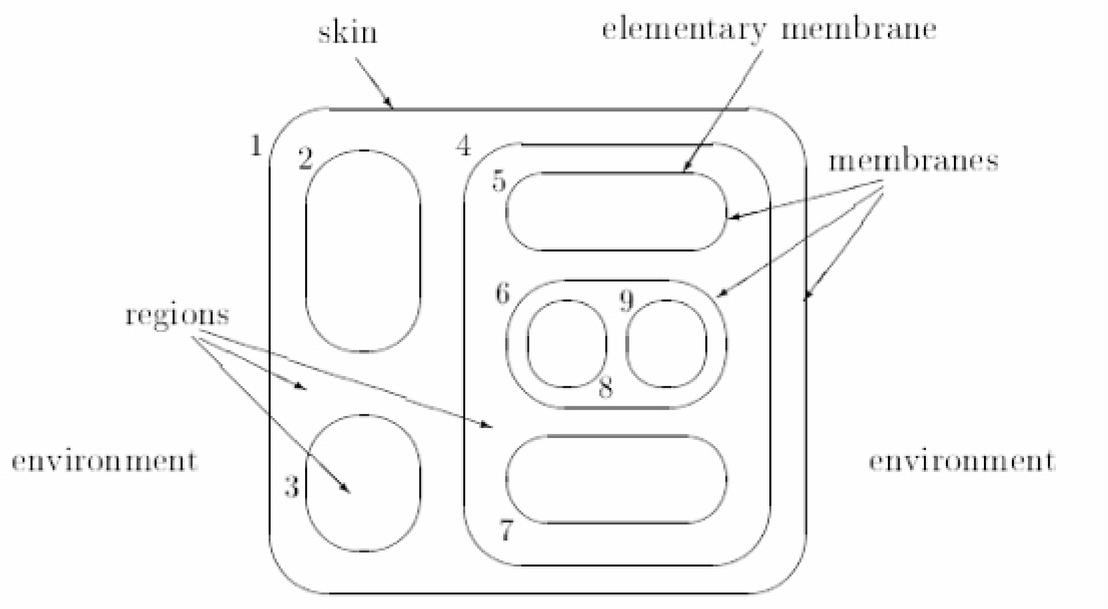


Fig. 1: P-system structure

Definition Multiset of objects Let U be a finite and not empty set of objects and N the set of natural numbers. A multiset of objects is defined as a mapping:

$$M : U \rightarrow N$$
$$a_i \rightarrow u_1$$

Where i a is an object and i u its multiplicity

$$M = \{(a_1, u_1), (a_2, u_2), (a_3, u_3). \ \} = a_1^{u1} \cdot a_2^{u2} \cdot a_n^{un} . \quad .$$

As it is well known, there are several representations for multisets of objects.

Note: Initial Multiset is the multiset existing within a given region in where no application of evolution rules has occurred yet.

Definition Evolution rule with objects in U and targets in T

Evolution rule with objects in U and targets in T is defined by $r = (m, c, \delta)$ where

$$m \in M(U), c \in M(UxT) \text{ and } \delta \in \{to\ dissolve, not\ to\ dissolve\}$$

From now on 'c' will be referred a s the consequent of the evolution rule 'r'. Note: The set of evolution rules with objects in Uand targets in T is represented by R (U, T). Definition Multiplicity of an object in a multiset of objects M(U) Let U ai ∈ be an object and let) U (M m∈ be a multiset of objects. The multiplicity of an object is defined over a multiset of objects such as:

$$
\begin{aligned}
|\ |_{a_i} &: U \times M(U) \rightarrow N \\
(a_i, m) &\rightarrow |m|_{a_i} = n \mid (a_i, n) \in m
\end{aligned}
$$

Definition Multiplicity of an object in an evolution rule r

Let U ai∈ be an object and let) T , U (R be a multiset of evolution rules. Let ( )) T , U (R , c , m r ∈ δ = where

$$m \in M(U), c \in M(UxT) \text{ and } \delta \in \{\text{to dissolve, not to dissolve}\}$$

The multiplicity of an object is defined over an evolution rules such as:

$$
\begin{aligned}
|\ |_{a_i} &: U \times R(U,T) \rightarrow N \\
(a_i, r) &\rightarrow |m|_{a_i} = n \mid (a_i, n) \in m
\end{aligned}
$$

P-system evolution

Let i C be the consequent of the evolution rule i r . Thus,

The representation of the evolution rules is:

$$
\begin{aligned}
&r_1 : a_1^{u1}\ \acute{a}_2^{u1}\ ?a_n^{u1n} \rightarrow C_1 \\
&r_2 : a_1^{u2}\ \acute{a}_2^{u2}\ ?a_n^{u2n} \rightarrow C_2 \\
&\dots\dots\dots\dots\dots\dots \rightarrow \dots\dots\dots\dots\dots \\
&r_m : a_1^{um1} a_2^{um2} . a_n^{um\quad n} \rightarrow C_m
\end{aligned}
$$

P-systems evolve, which makes it change upon time; therefore, it is a dynamic system. Every time that there is a change on the p-system we will say that the psystem is in a new transition. The step from one transition to another one will be referred to as an evolutionary step, and the set of all evolutionary steps will be named computation. Processes within the psystem will be acting in a massively parallel and nondeterministic manner. (Similar to the way the living cells process and combine information). We will say that the computation has been successful if:

III. The Upgrade

The proposal is a new computational paradigm based on Membrane cells, scalable ones which are capable to produce "computational tissues and organs". The organization of such computational tissues and organs is inspired by the emerging informational properties of biomolecular networks and will be based on scalable "membrane cells" guided by functional rules similar to the biological ones (molecular recognition, self-assembly and topo biology-theory rules).

The direct inspiration from the membrane cells is precisely the breakthrough of the MCA project. By building computational tissues our proposal makes an evolutionary jump with respect of today research in this field, mainly focused on aggregates of unicellular organisms (e.g. bacteria). Far from modelling and simulating the cellular processes, our computational paradigm will be a clear abstraction of the basic mechanisms and computational capabilities of the membrane cells and tissues, in order to solve complex problems in a new (bioinspired) way.

Real tissues display far more complex properties (emergent properties) than the sum of the properties of the individual membrane cells they are made from. In the same way, the emergent properties and functions of our membrane cells and computational tissues will be used for the resolution of real problems, impossible to be appropriately solved by conventional methods: not only biological morphogenesis, but also evolution of economic systems and prediction of crisis, optimization of "industrial ecologies", analysis of the dynamics of social interactions and conflicts, ecosystem disturbances, etc., that are more complex than combinatorial optimization, as well as other classical NP Complete ones.

Our "membrane cells" will be a species of "proto-membrane cells" and a far objective of the project is also the ex-novosynthesis of " membrane cells" and tissues performing as living computational biomolecular networks. The lon2-term vision that motivates this breakthrough is to build new information processing devices with evolving capabilities, which will adapt themselves to the complexity of the problems.

In particular, we foresee a synthetic approach to build computational membrane cells and tissues, and to create computational bio-inspired devices of higher complexity (tissues-organs). A far future objective of the project goes beyond the mathematical, software and hardware tools. It is to obtain in lab synthesized "living" information processing systems based on artificial “membrane cells" and hybrid systems combining living components (our "synthesized membrane cells") and non-living elements (e.g. silicon-based).

MCA approach is the most appropriate to deal with extremely complex problems that will be crucial in the future. It shows potential to go beyond classical Biocomputing strategies such as self-reproducing machines, cellular automata, perceptron’s & neural networks, genetic algorithms, adaptive computing, bacteria-based computation, artificial membrane cells, etc. Specifically, a new generation of natural computing could be built, based upon the scalable " membrane cells" with problem solving capacity in very different realms: biomaterials and bioengineering, non-linear parallel processing, design of bioinspired systems, modelling of economic, industrial and financial systems, optimization strategies in social settings, etc. For the achievement of our long-term objectives we need to: analyze the wide amount of existing knowledge regarding one of the deepest sources of biocomputational power, the topological and flexible networking properties of biomolecular scalable modules in membrane cells, realize an abstraction of the basic mechanisms and computational capabilities of the membrane cells both at sub cellular and networking level, and develop formal models to be used in new information processing technologies, basically based on combinatory processes of protein domains and genetic switches, together with cytoskeleton dynamics and topobiology-theory, use the above proposed models to create scalable "/proto membrane cells" and abstract-formal "evolvable" cellular networks and computational tissues & organs endowed with these flexible modularity properties.

For our far final objective we need to obtain in lab proof that synthesis of new forms of living" membrane cells" in an inverse process: "membrane cells and tissues" => "theoretical abstract/formal models" => "artificial membrane cells and tissues" => "in lab synthesized living membrane cells" is possible. MCA breakthrou2h is an essential step towards the achievement of our lon2-term vision because it will set the theoretical basis and develop the experimental tools for the creation of the scalable membrane cells, computational tissues and organs (both abstract and living ones).

IV. MCA System

A MCA is a set Ω={ℿ0,ℿ1,ℿ2,…, −1,) and a set ℵ of aggregation rules among membranes. The set of aggregation rules are not fully integrated with the evolution rules of a given p-System but establishes the correlation between 2 given membrane models by deciding the way 2 or mere P-systems are being aggregated. The rules can be defined as a Matrix relation

$$\varphi_1(k_1, k_2, \;. k_{m'}) \equiv \begin{pmatrix} u_1 \;\; {}_1 u_2 \;\; {}_1 \cdot u_{m1} \\ u_1 \;\; {}_2 u_2 \;\; {}_2 \cdot u_{m2} \\ \cdot\;\cdot\;\cdot\;\cdot\;\;\cdot\;\cdot\;\cdot\;\cdot\;\;\cdot\;\cdot\;\cdot \\ u_{1n} \quad u_{2n} \quad \cdot\; u_m \end{pmatrix}_n \begin{pmatrix} k_1 \\ k_2 \\ \cdot\;\cdot\;\cdot \\ k_m \end{pmatrix} = \begin{pmatrix} u_1 \\ u_2 \\ \cdot\;\cdot \\ u_n \end{pmatrix}.$$

Where φ1(k) is the aggregation relation and is defined by the association of n P-systems, k determines the aggregation rules of each component in every psystem Iand Uare the component (objects). Evolution rule application phase.

This phase is the one that has been implemented following different techniques. In every region within a p-system, the evolution rules application phase is described as follows: Rules application to a multiset of object in a region is a transforming process of information which has input, output and conditions for making the transformation. Given a region within a p-system, let U= } n i | a { i ≤≤ 1 be the alphabet of objects, m a multiset of objects over U and R(U,T) a multiset of evolution rules with antecedents in U and targets in T. The input in the region is the initial multiset m. The output is a maximal multiset m'. The transformations have been made based on the application of the evolution rules over m until m' is obtained.

Application of evolution rules in each region of P systems involves subtracting objects from the initial multiset by using rules antecedents. Rules used are chosen in a non-deterministic manner. This phase ends when no rule is applicable anymore. The transformation only needs rules antecedents as the consequents are part of the communication phase.

Observation

Let N ki∈ be the number of times that the rule i r is applied. Therefore, the number of symbols j a which have been consumed after applying the evolution rules a specific number of times will be:

$$\sum_{i=1}^{m} k_i \cdot u_{ij}$$

Definition

Given a region R and alphabet of objects U, and R (U, T) set of evolution rules over U and targets in T.

$$\begin{aligned}
r_1 &: a_1^{u_{1\,1}} a_2^{u_{1\,2}} . a_n^{u_{1n}} \rightarrow C_1 \\
r_2 &: a_1^{u_{2\,1}} a_2^{u_{2\,2}} . a_n^{u_{2n}} \rightarrow C_2 \\
&\quad \ldots \qquad \rightarrow \ldots . \\
r_m &: a_1^{u_{m1}} a_2^{u_{m2}} . a_n^{u_{m\,n}} \rightarrow C_m
\end{aligned}$$

Maximal multiset is that one that complies with:

$$\bigcap_{l=1}^{m} \left[\bigcup_{i=1}^{n} \left(u_i - \sum_{j=1}^{m} (k_j \cdot u_{j\,i}) \le u_{l\,i} \right) \right]$$ [1] [Arteta,2010]

V. Correction

The correction of the system fully relies in the correction of the internal P-system of the MCA. In order to prove the aggregation system is distributed then 2 processes need to be proven.

1. Correction of the formal definition of Transition PSystem (Paun , 1998)
2. Correction of the aggregation rules applying to 2 given P-systems.

The correction of the second point gets reduced to a deductive demonstration where the aggregation of 2 given P-systems is base case and the generic case of n-P-systems can be seen as the aggregation of n-1 Psystems (inductive case) with a correct aggregation to the last one.

Thus, the key is to prove that aggregation of 2 given P-system is a correct process and indeed reinforce the idea of full inherent parallelism and nondeterministic modelling that membrane models are after.

Aggregation rule. Let us use a short definition of a given P-System ( ) 0 1 1 1 i) , R) , . . (R (, , . . . , , , V n , n , n ρ ρ ω ω μ = ∏ Base case. Given 2 Transition P-system � 1={ℿ1 1} 2={ℿ2 2} Aggregation 1 2) 12={ℿ12 1 12 12} where � 1 2 are 2 given P-Systems, 12 is the aggregated Psystem where ℿ12 is the aggregated alphabet of both P-systems , 12 is the set of regions in the aggregated P-system and 12 12 are the multiset of objects and set of evolution rules of the aggregated P-system.

- Building the aggregated alphabet ℿ12 is obvious.

The result is the Union of both. Correctness for this operation is also obvious.

- The aggregation of the 2set of multisets is obvious.

The result is the Unionof both. Correctness for this operation is also obvious.

- The aggregation of the 2set of the set of the evolution rules 12 is obvious. The result is the Union of both. Correctness for this operation is also obvious.

There are 2 factors in the aggregation that are not obvious which are the aggregated Set of regions � 12. This set of regions is constructed in our proposal as supervised and directed by the factor λ that defines the capabilities previously mentioned. This λ is defined dynamically by the nature of problem the MCA is about to fix. i.e.in a problem of sum of squares is not necessary aggregation as 2 independent P-system could calculate their squares [Paun,2001] and send those outputs to a third (obvious) one that calculates the sum of both results. However, for didactic purposes and aggregated solution could be provided in where a MCA is created with 2 Input P-systems. The aggregated would assign equal λ (priority) to both of them, and then either of them could contain the other one. The container P-system process the output of the contained P-system by adding it to an another square number.

Other problems, especially those that requires sub solutions that are part of optimization techniques would be required to establish a clear hierarchy in the aggregation of MCA. Thus:

- The aggregation of the regions of 2 P-systems would be determined by a priority or hierarchy described by λ. This is a dynamic factor that must be configured right before the problem is dealt with.
- The aggregated P-system will have to work the communication phase after every evolutionary step. This communication phase also fully relies on the hierarchy establish by λ and will operate as normal when the aggregation is complete and the MCA is finished.

a) Inductive case

Given a successful aggregation (MCA) of n P-systems MCA (n), is it correct to aggregate n+1 P-systems?

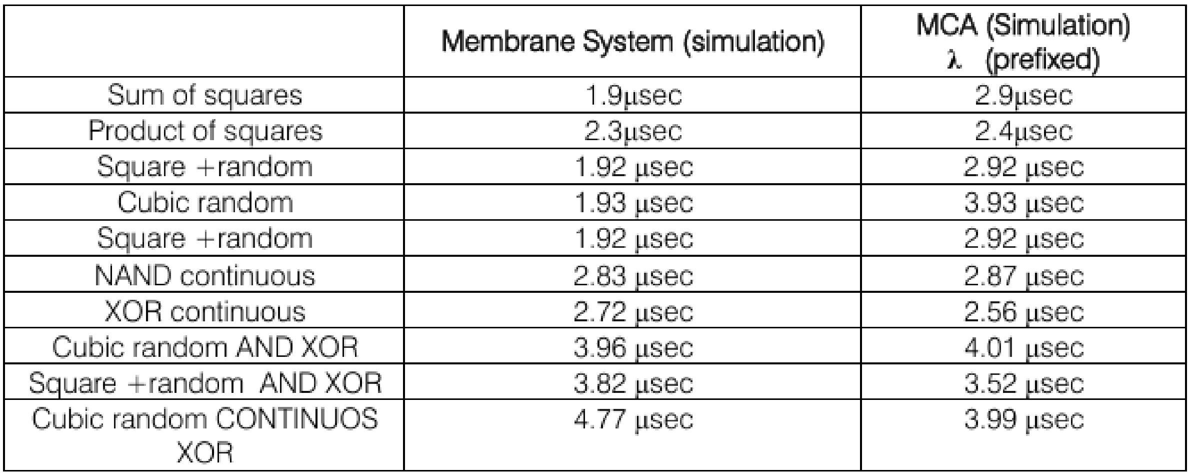
The inductive case is a direct consequence of the aggregated property.

MCA (n) system becomes a complex P-System with an aggregation of regions according to the λ factor .MCA (n)= let’s call the aggregated P-system as � ={ℿ1 1}. Once the aggregation is seen as a

P-system, aggregating it with another 1 is obvious by applying the base case.

b) Simulations and results

We have been performing some simulations in simple problem solving in same traditional computing paradigm for small problems clearly aggregation is not necessary, although the advantage of this proposal shows up, when the complexity of the problem increases. Theoretically a fully and corrected aggregated Solution (A whole MCS) would overweight the cost of the calculation of λ and he redesign of the membrane system that can always occur during compiling time anyways.



	Membrane System (simulation)	MCA (Simulation) λ (prefixed)
Sum of squares	1.9µsec	2.9µsec
Product of squares	2.3µsec	2.4µsec
Square +random	1.92 µsec	2.92 µsec
Cubic random	1.93 µsec	3.93 µsec
Square +random	1.92 µsec	2.92 µsec
NAND continuous	2.83 µsec	2.87 µsec
XOR continuous	2.72 µsec	2.56 µsec
Cubic random AND XOR	3.96 µsec	4.01 µsec
Square +random AND XOR	3.82 µsec	3.52 µsec
Cubic random CONTINUOS XOR	4.77 µsec	3.99 µsec

The analysis is very direct. The simulations are running in the same platform and just focuses in performance time based. All problems are considered simple problems due to the limitations of processing a complex problem with a complex set of aggregation rules which will jeopardize the accuracy of the analysis.

Nevertheless, it is indicative to see that there is a variation in the performance when the level of complexity slightly increases which suggest that aggregation can be a good approach when the level of complexity increases.

VI. Conclusions

Membrane computing has been growing since George Paun defined it in 1998. Since then new variations have been suggested to try to fit this model to new realities. The main goal for this unconventional paradigm is to improve the performance of the traditional algorithms due to the inherent limitation of the model. Simulations are still a big part of membrane computing and they are useful to extract right conclusions about the new model. In particular, this model is a great candidate to be applied to complex models that require an aggregated solution that is part of other sub solution whole super solutions as long as the defined rules in the MCA are followed. The aggregation factor that is linked to the minimal membrane cells is the component that complement the use membrane computing as a whole and as unite aggregated model. As the creation of this factor generates difficulties because it depends on the nature of the problem, it does not damage the performance during the execution as the factor is calculated in compiling time. New techniques to atomize the generation of λ as this could create a complete dynamic model that fully adjust to the problem and create the right MCA. The necessity of opening the line of research is out of question. The field is growing and new experiments are required. MCA systems are provided as a natural solution to upgrade the nature of membrane computing by not only taking advantage of the properties of the membrane cells but by the way these cells are aggregated. The future work will be involving complex problems in complex aggregated structures, so the analysis can be more relevant. Nevertheless, the evidence points out that aggregation is a natural solution to deal with complex problems that nowadays are being processed by conventional approaches such as backtracking or dynamic programming.

References Références Referencias

1. Adelman, (1994) L.M.: Molecular computation of solutions to combinatorial problems, Science, 226:1021 1024, 1994. Amir-Kroll H., Sadot A., Cohen I.R. and Harel D. (2008), "GemCell: A generic platform for modelling multi-cellular". Theoretical Computer Science 391: 276-290.

2. Arteta (2018) an isomorphism based algorithm to solve complex problems), Alberto Arteta, Luis F. Mingo, Juan Castellanos Wseas Transactions on Information Science and Applications.

3. Arteta (2014) "New Approach to Optimize Membrane Systems, Journal of bioinformatics" Jan 2014, Vol. 1 Iss. 1, PP. 1-6 http://www.bowenpublishing.com/jbi/Publisher.aspx

4. Arteta (2013) Technical P-systems: Operating in the stock markets with Transition P-systems.

5. Alberto Arteta, Angel Luis Castellanos, Nuria Gómez Blas "Information Research and

Applications" (i.TECH 2013).

6. Arteta (2012) Membrane systems working with the P-factor: best strategy to solve complex problems", ICAEE 2012, published by Advanced Science Letters alberto Arteta Luis Fernando Mingo, Nuria Gomez (ISSN: 1936-6612). JCR Indexed in ISI Web of Science, SciVerse, 2010 Impact factor: 1.253 DOIhttp://dx.doi.org/10.1166/asl.2013.4453,Volume n19, number 5 1490-1495.http://www.aspbs.com/ science.htm

7. Arteta (2012)-2 Meia Systems: membrane encrypted information application systems." Natural Information Technologies Madrid International Journal "Information Theories and Applications", Vol. 19, Number 2, 2012, ISSN 1313-0455 7 pages cirrus Google Scholar.

8. Arteta (2012)-3 Solving Diophantine equations with a parallel membrane computing model", Alberto Arteta, Nuria Gomez, Rafael Gonzalo International Journal "Information Models and Analyses" Vol.1 / 2012 SSN 1314-6416 (printed) 6 páges.

9. Arteta (2011) "Solving complex problems with a Bioinspired model", Engineering of Application of Artificial Intelligence, Alberto Arteta Luis Fernando Mingo Nuria Gomez ISSN: 0952-1976 Volumen 24, Issue 6, September 2011, Pages 919–927. JCR Indexed 2010 Impact factor 1.44 Q1. http://www.sciencedirect.com/science/article/pii/S09 52197611000571

10. Arteta (2010) Intelligent P-systems", Alberto Arteta, Angel Luis Castellanos, Jose Luis Sanchez I. TECH 2010, International Conference ITHEA Journal 07/2010 " New Trends in Information Technologies" ISBN 978-954-16-0044-9 Sofia (Bulgaria).

11. Arteta (2009) P-systems: Study of randomness when applying evolution rules". International Book Series Information Science and Computing", Alberto Arteta, Luis Fernández, Fernando Arroyo 09/2009 9 pages ISSN 1313-0455 pages (15-24).

12. Arteta (2009)-2 "Analysis of P-systems under multiagents perspective", Alberto Arteta, Angel Goñi, Juan Castellanos2009, 12 pages ISSN 13130455 , "Information Science and Computing" Pages (117-128).

13. Angeleska, A. et al.: RNA-guided DNA assembly, J. Theoretical Biology, 248, 706-720, 2007.

14. Ardelean, I. et al.: A computational model for cell differentiation, Bio Systems, 76, 1-3:169-176, 2004.

15. Balazsi G., Barabasi A.-L. and Oltvai Z. N. (2005), "Topological units of environmental signal processing in the transcriptional regulatory network of Escherichia coli". PNAS 102 (22) 7841-46.

16. Bashor C.J., Horwitz A.A., Peisajovich S.J. and Lim W.A. (2010), "Rewiring Cells: Synthetic Biology as a Tool to Interrogate the Organizational Principles of Living Systems". Annual Review of Biophysics 39: 515-37.

17. Blow N. (2009), "Systems biology: Untangling the protein web". Nature 460: 415-418.

18. Bray D. (2009), "Wetware: A Computer in Every Living Cell: The Computer in Every Living Cell ".

The Computer in Every Living Cell ". Yale University Press, New Haven.

19. Brooks, R.A. (2001), "The Relationship Between Matter and Life". Nature 409: 409-11.

20. Cao H., Romero-Campero F.J., Heeb S., Camara M. and Krasnogor N. (2010), "Evolving cell models for systems and synthetic biology". Systems and Synthetic Biology 4 (1): 55-84.

21. Carroll S.B. (2005), "Endless Forms Most Beautiful". W. W. Norton, Chicago.

22. Danchin A. (2009), "Bacteria as computers making computers". FEMS Microbiol Rev. 33(1):3-26.

23. Dasgupta, D. (Eds.): Artificial Immune Systems and Their Applications, Springer 1998.

24. Dassow, J., and V. Mitrana: On some operations suggested by genome evolution. In: R.B. Altman, et al. (Eds.) Proc. Second Pacific Symposium on Biocomputing, 97--108, 1997.

25. Dassow, J., V. Mitrana, and A. Salomaa. Context-free evolutionary grammars and the structural language of nucleic acids. BioSystems 43:169--177, 1997.

26. Dassow, J., V. Mitrana, and A. Salomaa: Operations and language generating devices suggested by the genome evolution, Theoretical Computer Science, 270, 1-2: 701-731, 2002.

27. Edelman G.M. (1988), "Topobiology: An introduction to molecular embriology". Basic Books, New York.

28. Ehrenfeucht A, T. Harju, I. Petre, D.M. Prescott and G. Rozenberg: Computation in Living Cells: Gene Assembly in Ciliate, Springer, 2003.

29. Engelbrecht, A.: Fundamentals of Computational Swarm Intelligence, Wiley and Sons, 2005.

30. Federici D, Downing K. (2006), "Evolution and development of a multicellular organism: scalability, resilience, and neutral complexification". Artif Life 12(3): 381-409.

31. Freund, R, C. Martin-Vide and V. Mitrana: On some operations suggested by gene assembly in ciliates. New Generation Computing 20:279-293, 2002.

32. Frutos (2010) Improving active rules performance in new P system communication architectures" ,Juan Alberto de Frutos, Luis Fernández, Carmen Luengo, Alberto Arteta Information Technologies and Knowledge, 10 ISSN 1313-048X Vol. 4, Number 1, 2010 Pages(3-18)

33. "Membrane Computing:Non deterministic technique to calculate extinguished multisets" Information Technologies and KnowledgeISSN 1313-048X Vol. 4, Number 1, 2010. (Madrid) 09/2010 Pages (30-41).

34. Frutos(2009) "Usefulness States in New P System Communication Architecture", Juan Alberto de Frutos, Fernando Arroyo, Alberto Arteta http://www.springerlink.com/content/978-3-54095884-0/Lecture Notes in Computer Science, 2009, Volumen 5391/2009, 169-186, DOI:10.1007/978-3540-95885-7_13 SJR 0.29 Q2

35. Han J., Bertin N., Hao T., Goldberg D.S., Beniz G.F., Zhang L.V., Dupuy D., Walhout A.J.M., Cusick M.E., Roth F.P. & Vidal M. (2004), "Evidence for dynamically organized modularity in the yeast protein–protein interaction network". Nature 430, 88-93.

36. Haynes K.A. & Silver P.A. (2009), "Eukaryotic systems broaden the scope of synthetic biology". JCB 187 (5): 589-96. Ho L. & Crabtree G.R. (2010), "Chromatin remodelling during development". Nature 463, 474-84.

37. Holcombe M. and A. Bell: Computational models of immunological pathways. In Information Processing in Cells and Tissues, Holcombe M, Paton R (eds.). Plenum Press: New York and London; 213-226, 1998.

38. Huh D., Matthews B.D., Mammoto A., MontoyaZavala M., Hsin H.Y. and Ingbe D.E. (2010), "Reconstituting Organ-Level Lung Functions on a Chip". Science 25: 328 (5986): 1662-68.

39. Hunter P., Robbins P. and Noble D. (2002), "The IUPS human physiome project". Eur J Physiol 445:1-9.

40. Istrail, S., De-Leon, B.T. and Davidson, E.: The regulatory genome and the computer, Developmental Biology 310, 187-195, 2007.

41. Kari L and L. Landweber: Computational power of gene rearrangements. American Mathematical Society 54:207-216, 2000. Karp, R.M.: Mathematical challenges from genomics and molecular biology. Notices of the AMS, 49: 544553, 2002. Kondo S. & Miura T. (2010), "Reactiondiffusion model as a framework for understanding biological pattern formation". Science 329(5999): 1616-20.

42. Luscombe NM, Babu MM, Yu H, Snyder M, Teichmann SA, Gerstein M. (2004), "Genomic analysis of regulatory network dynamics reveals large topological changes". Nature 16 (431): 308-12.

43. Ma'ayan A., Blitzer R.D. and Iyengar R. (2005), "Toward predictive models of mammalian cells". Annual Review of Biophysics and Biomolecular Structure 34: 319-49.

44. Manea, F., Martin-Vide, C. and V. Mitrana: Accepting Networks of Evolutionary Word and Picture Processors: ANoble D. (2002), "Modelling the Heart--from Genes to Cells to the Whole Organ". Science 1 295: 1678-82.Oda K., Kimura T., Matsuoka Y., Funahashi A., Muramatsu M. and Kitano H. (2004), "Molecular Interaction Map of a Macrophage". Signalling 2 (14): 1-12.

45. Palmer R.A. (2004), "Symmetry breaking and the evolution of development". Science 306 (5697): 82833. Nun, G.: Membrane Computing: An Introduction. Springer Verlag, Berlin, 2002.

46. Pisinger D. (2003), "Where are the hard knapsack problems?". Comput. Oper. Res. 32 (9): 2271-84.

47. Prescott, D., Ehrenfeucht, A. and Rozenberg, G.: Template guided recombination for IES elimination and unscrambling of genes in stichotrichous ciliates, J. Theoretical Biology 222, 323-330, 2003.

48. Purnick P.E.M. & Weiss R. (2009), "The second wave of synthetic biology: from modules to systems". Nature Reviews Molecular Cell Biology 10: 410-22.

49. Ray L.B. (2010), "Networks Evolving Bacteria and Computers". Sci. Signal 3 (123): 160-9.

50. Reif, J. And LaBean, T.: Autonomous programmable biomolecular devices using self-assembled DNA nanostr tures, Commun. ACM 50, 46-53, 2007.

51. Schlosser G. & Wagner G.P. (2004), "Modularity in development and evolution" Chicago University Press. Chicago. Sprinzak D., Lakhanpal A., LeBon L., Santat L.A., Fontes M.E., Anderson G.A., GarciaOjalvo J. and Elowitz M.B. (2010), "Cis-interactions between Notch and Delta generate mutually exclusive signalling states". Nature 465: 86-90.

52. Suzuki Y, and H. Tanaka: Symbolic chemical system based on abstract rewriting system and its behavior pattern. Artif LifeRobotics 1:211-219, 1997.

53. Szathmary E. (2001), "Evolution: Developmental circuits rewired". Nature 411: 143-45.

Classification of Image using Convolutional Neural Network (CNN)

Md. Anwar Hossainα & Md. Shahriar Alam Sajibσ
Assistant Professor, Department of Information & Communication Engineering, Faculty of Engineering & Technology, Pabna University of Science & Technology, Pabna, Bangladesh.
Student, Department of Information & Communication Engineering, Faculty of Engineering & Technology, Pabna University of Science & Technology, Pabna, Bangladesh.

ABSTRACT

Computer vision is concerned with the automatic extraction, analysis, and understanding of useful information from a single image or a sequence of images. We have used Convolutional Neural Networks (CNN) in automatic image classification systems. In most cases, we utilize the features from the top layer of the CNN for classification; however, those features may not contain enough useful information to predict an image correctly. In some cases, features from the lower layer carry more discriminative power than those from the top. Therefore, applying features from a specific layer only to classification seems to be a process that does not utilize learned CNN's potential discriminant power to its full extent. Because of this property we are in need of fusion of features from multiple layers. We want to create a model with multiple layers that will be able to recognize and classify the images. We want to complete our model by using the concepts of Convolutional Neural Network and CIFAR-10 dataset. Moreover, we will show how MatConvNet can be used to implement our model with CPU training as well as less training time. The objective of our work is to learn and practically apply the concepts of Convolutional Neural Network.

Keywords: convolutional neural network, CIFAR-10 dataset, MatConvNet, relu, softmax*.*

1. INTRODUCTION

Convolutional Neural Networks (CNN) becomes one of the most appealing approaches recently and has been an ultimate factor in a variety of recent success and challenging applications related to machine learning applications such as challenge ImageNet object detection, image classification, and face recognition. Therefore, we consider CNN as our model for our challenging tasks of image classification. We use CNN for segmentation and classification of the images in academic and business transactions. We use image recognition in different areas for example automated image organization, stock photography, face recognition, and many other related works.

a) CIFAR-10 Database

The CIFAR-10 database (Canadian Institute for Advanced Research database) is a collection of images.

We use this dataset to train machine learning and computer vision algorithms. CIFAR-10 database is the contribution of Alex Krizhevsky and Geoffrey Hinton. This dataset has 60,000 colored images. It has ten classes, and they are an airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck. The images are of size 32x32 pixels. The dataset consists of 50,000 training and 10,000 testing examples. It is a database for people who want to try learning techniques and pattern recognition methods on real-world data while spending minimal efforts on preprocessing and formatting. We will use this database in our experiment.

b) Convolutional Neural Networks

Convolutional neural networks are deep artificial neural networks. We use CNN to classify images, cluster them by similarity (photo search), and perform object recognition within scenes. It can be used to identify faces, individual, street signs, tumors, platypuses and many other aspects of visual data. The convolutional layer is the core building block of a CNN. The layer's parameters consist of a set of learnable filters (or kernels) which have a small receptive field but extend through the full depth of the input volume. During the forward pass, each filter is convolved across the width and height of the input volume, computing the dot product, and producing a 2-dimensional activation map of that filter. As a result, the network learns about the filters. The filter activates when they see some specific type of feature at some spatial position in the input. Then the activation maps are fed into a downsampling layer, and like convolutions, this method is applied one patch at a time. CNN has also fully connected layer that classifies output with one label per node.

II. Related Works

Image recognition has an active community of academics studying it. A lot of important work on convolutional neural networks happened for image recognition [1,2,3,4]. The most dominant recent works achieved using CNN is a challenging work introduced by Alex Krizhevsky [5], who used CNN for challenge classification ImageNet. Active areas of research are: object detection [14,15,16], scene labeling [17], segmentation [18,19], face recognition, and variety of other tasks [20,21,22].

III. Methodology

Deep Learning has emerged as a main tool for self-perception problems like understanding images, the voice from humans, robots exploring the world. We aim to implement the concept of the Convolutional Neural Network for the recognition of images. Understanding CNN and applying it to the image recognition system is the target of the proposed model. Convolutional Neural Network extracts the feature maps from the 2D images by using filters. The Convolutional neural network considers the mapping of image pixels with the neighborhood space rather than having a fully connected layer of neurons. The Convolutional neural network has been proved to be a very dominant and potential tool in image processing. Even in the fields of computer vision such as handwriting recognition, natural object classification, and segmentation, CNN has become a much better tool compared to all other previously implemented tools.

a) The architecture of the Proposed Model

When one starts learning deep learning with the neural network, he realizes that one of the most supervised deep learning techniques is the) D (Convolutional Neural Network. We design Convolutional Neural Network to recognize visual patterns directly from pixel images with minimal preprocessing. Almost all CNN architectures follow the same general design principles of successively applying convolutional layers to the input, periodically downsampling (Max pooling) the spatial dimensions while increasing the number of feature maps. Moreover, there are also fully connected layers, activation functions and loss function (e.g., cross entropy or softmax). However, among all the operations of CNN, convolutional layers, pooling layers, and fully connected layers are the most important ones. Therefore, we will quickly introduce these layers before presenting our proposed model.

The Convolutional layer is the very first layer where it can extract features from the images. Because pixels are only related to the adjacent and close pixels, convolution allows us to preserve the relationship between different parts of an image. Convolution is filtering the image with a smaller pixel filter to decrease the size of the image without losing the relationship between pixels. When we apply convolution to a 7x7 image by using a filter of size 3x3 with 1x1 stride (1-pixel shift at each step), we will end up having a 5x5 output.

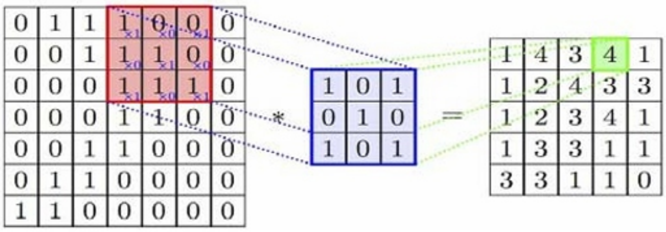


Fig. 1: Convolution Operation

When constructing CNN, it is common to insert pooling layers after each convolution layer, so that we can reduce the spatial size of the representation. This layer reduces the parameter counts, and thus reduces the computational complexity. Also, pooling layers help with the overfitting problem. We select a pooling size to reduce the amount of the parameters by selecting the maximum, average, or sum values inside these pixels. Fig.2 shows the max pooling and average pooling operation.

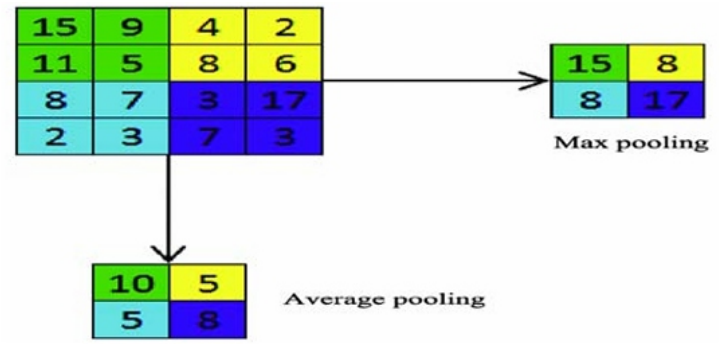


Fig. 2: Max pooling and Average pooling operation

A fully connected network is in any architecture where each parameter is linked to one another to determine the relation and effect of each parameter on the labels. We can vastly reduce the time-space complexity by using the convolution and pooling layers. We can construct a fully connected network in the end to classify our images.

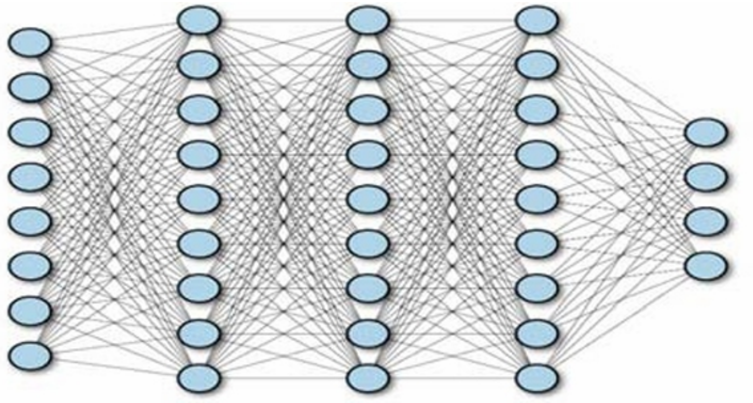


Fig. 3: Fully connected layer

Fig.4 shows the overview look of our proposed convolutional neural network. It is very much similar to the other image recognition architectures [1,2,3,4] but has changed in the number of filters, neurons and activation functions for better performance. We can divide our model into six sequences of layers.

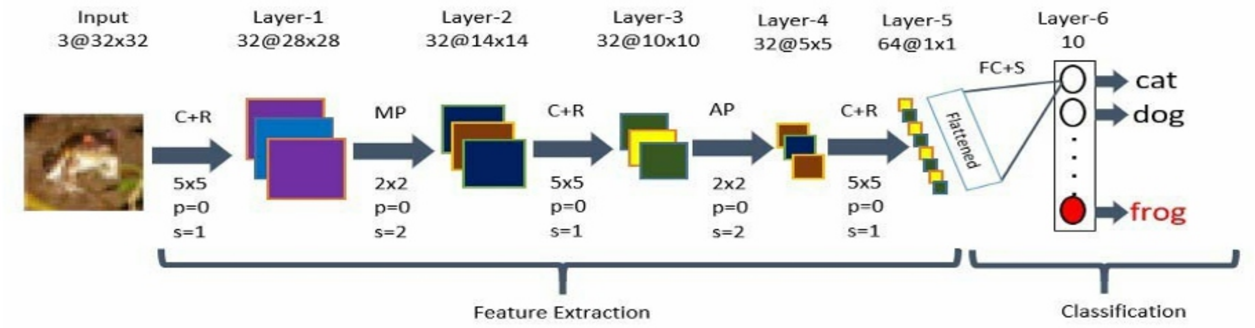


Fig. 4: The architecture of our proposed CNN

b) Explanation of the Model

A simple convolutional network is a sequence of layers. The layer transforms one volume of activations to another through a differentiable function. We use three main types of layers to build network architecture. They are a convolutional layer, pooling layer, and fully connected layer. We will stack these layers to form six layers of network architecture. We will go into more details below.

Fig.4 shows the architecture of our proposed CNN model. At first, we need some pre-processing on the images like resizing images, normalizing the pixel values, etc. After the necessary pre-processing, data is ready to be fed into the model.

Layer-1 consists of the convolutional layer with ReLu (Rectified Linear Unit) activation function which is the first convolutional layer of our CNN architecture. This layer gets the pre-processed image as the input of size n*n=32*32. The convolutional filter size (f*f) is 5*5, padding (p) is 0(around all the sides of the image), stride (s) is 1, and the number of filters is 32. After this convolution operation, we get feature maps of size 32@28*28 where 32 is the number of feature maps which is equal to the number of filters used, and 28 comes from the formula ((n+2p-f)/s) +1= ((32+2*05)/1) +1=28. Then the ReLu activation is done in each feature map.

Layer-2 is the max pooling layer. This layer gets the input of size 32@28*28 from the previous layer. The pooling size is 2*2; padding is 0 and stride is 2. After this max pooling operation, we get feature maps of size 32@14*14. Max pooling is done in each feature map independently, so we get same number feature maps as the previous layer, and 14 comes from the same formula ((n+2p-f)/s) +1. This layer has no activation function.

Layer-3 is the second convolutional layer with ReLu activation function. This layer gets the input of size 32@14*14 from the previous layer. The filter size is 5*5; padding is 0, the stride is 1, and the number of filters is 32. After this convolution operation, we get feature maps of size 32@10*10. Then ReLu activation is done in each feature map.

Layer-4 is the average pooling layer. This layer gets the input of size 32@10*10 from the previous layer. The pooling size is 2*2; padding is 0 and stride is 2. After this max pooling operation, we get a feature map of size 32@5*5. Layer-5 is the third convolutional layer with ReLu activation function. This layer gets the input of size 32@5*5 from the previous layer. The filter size is 4*4; padding is 0, the stride is 1, and the number of filters is 64. After this convolution operation, we get feature maps of size 64@1*1. This layer acts as a fully connected layer and produces a one-dimensional vector of size 64 by being flattened.

Layer-6 is the last layer of the network. It is a fully connected layer. This layer will compute the class scores, resulting in a vector of size 10, where each of the ten numbers corresponds to a class score, such as among the ten categories of CIFAR-10 dataset. For final outputs, we use the softmax activation function.

In this way, CNN transforms the original image layer by layer from the main pixel values to the final class scores. Note that some layers contain parameters, and others don't. In particular, the convolution/fully connected layers perform transformations that are a function of not only the activations in the input volume but also of the parameters (the weights and biases of the neurons). On the other hand, the Relu/pooling layers will implement a fixed function. We train the parameters in the convolutional/fully connected layers with stochastic gradient descent. By this process, we will prepare the trained model which will be used to recognize the image present in the test data. Thus, we can classify the images as Class- airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, trucks.

IV. Implementation

To implement our CNN architecture, we will use MatConvNet. MatConvNet is an implementation of Convolutional Neural Networks (CNN) for MATLAB [23]. We built our model by using MatConvNet so that our model has greater simplicity and flexibility. It exposes the building blocks of CNN as easy-to-use MATLAB functions, providing routines for computing linear convolutions with filter banks, feature pooling and many more. In this manner, MatConvNet allows fast prototyping of new CNN architectures; at the same time, it supports efficient computation on CPU and GPU allowing to train complex models on large datasets such as ImageNet ILSVRC.

Convolutional Neural Networks (CNN) is the current state-of-art architecture for the image classification task. As shown in Fig.4 our proposed 2-D Convolutional Neural Network (CNN) model is designed using MatConvNet backend for the well known CIFAR10 image recognition task. The whole workflow can be to preparing the data, building and compiling the model, training and evaluating the model and saving the model to disk for reuse.

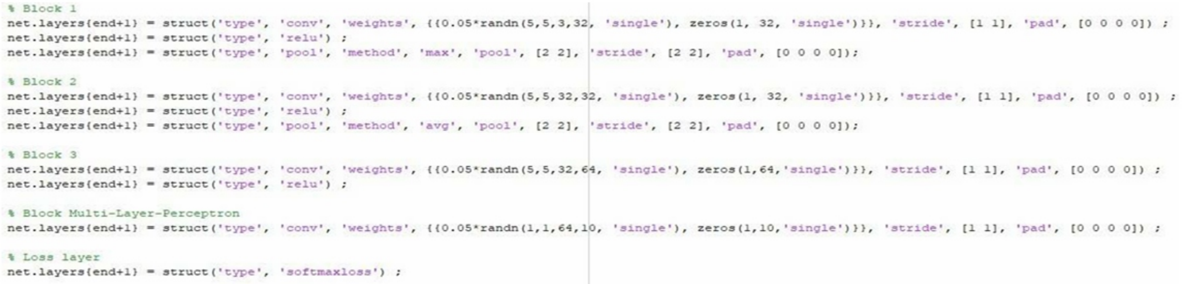
Preparing the data is the first step of our approach. Before we build the network, we need to set up our training and testing data, combine data, combine labels and reshape into the appropriate size. We save the dataset of normalized data (single precision and zero mean), labels, and miscellaneous (meta) information.

Building and compiling the model is the second step. To create the CNN, we must initialize MatConvNets SimpleNN network and then define important initialization parameters for example batch size, number of epochs, learning rate, etc.

The batch size determines the number of samples for the training phase of the CNN. The CNN will process all the training data, but only in increments of the specified batch size. We can use batch size for

We can use batch size for computational efficiency, and its value will be dependent on the user's available hardware. An epoch is a successful forward pass and a backward pass through the network. It's usually beneficial to set its value high and then to reduce it once if one is satisfied with the convergence at a particular state (chosen epoch) in the network. Learning rate is a very sensitive parameter that pushes the model towards convergence. Finding its best value will be an experimental process unless one invokes more powerful techniques such as batch normalization. In our experiment, we use batch size 60, several epochs 300 and learning rate 0.0001 for maximum accuracy.

Now we can build our CNN by creating each layer individually as shown in fig 5. Afterward, we will invoke objective and error layers that will provide a graphical visualization of the training and validation convergence after completing each epoch. MatconvNet initializes the weights by using Gaussian distribution.



```
% Block 1
net.layers{end+1} = struct('type', 'conv', 'weights', {{0.05*randn(5,5,3,32, 'single'), zeros(1, 32, 'single')}}, 'stride', [1 1], 'pad', [0 0 0 0]) ;
net.layers{end+1} = struct('type', 'relu') ;
net.layers{end+1} = struct('type', 'pool', 'method', 'max', 'pool', [2 2], 'stride', [2 2], 'pad', [0 0 0 0]);

% Block 2
net.layers{end+1} = struct('type', 'conv', 'weights', {{0.05*randn(5,5,32,32, 'single'), zeros(1, 32, 'single')}}, 'stride', [1 1], 'pad', [0 0 0 0]) ;
net.layers{end+1} = struct('type', 'relu') ;
net.layers{end+1} = struct('type', 'pool', 'method', 'avg', 'pool', [2 2], 'stride', [2 2], 'pad', [0 0 0 0]);

% Block 3
net.layers{end+1} = struct('type', 'conv', 'weights', {{0.05*randn(5,5,32,64, 'single'), zeros(1,64,'single')}}, 'stride', [1 1], 'pad', [0 0 0 0]) ;
net.layers{end+1} = struct('type', 'relu') ;

% Block Multi-Layer-Perceptron
net.layers{end+1} = struct('type', 'conv', 'weights', {{0.05*randn(1,1,64,10, 'single'), zeros(1,10,'single')}}, 'stride', [1 1], 'pad', [0 0 0 0]) ;

% Loss layer
net.layers{end+1} = struct('type', 'softmaxloss') ;
```

Fig. 5: CNN layers in MatConvNet

The third step is the training and evaluating the model. Training a CNN requires computing the derivative of the loss concerning the network parameters. We calculate the derivatives using an algorithm called back propagation which is a memoryefficient implementation of the chain rule for derivatives. We built the model and performed a random gradient descent training according to the Stochastic Gradient Descent (SGD) training algorithm. We have used SGD training algorithm to adjust the weight of the connection between neurons so that the loss reaches a minimum value or stops after several epochs. We have used training. It is important to note that GPU training will dramatically help training time for CNN.

Lastly, we can begin the training of CNN by supplying the training data, the constructed model and the current batch of data. When training the CNN, only the data specified for training play a role in minimizing error in the CNN. We feed the training data through the network for the forward pass and backward pass. The validation data is just used to see how the CNN responds to new similar data. We do not use the validation data to train the network. Afterward, we save the trained CNN and prepare for the testing phase.

During the training phase of the CNN, the simple network will produce three plots (Objective, Top1error, and Top5error) for each epoch. The top1 error is the chance that class with the highest probability is the correct target. In other words, CNN guesses the target correctly. The top5error is the chance that the true target is one of the top five probabilities. The objective for the simple network should mirror the form of the top1 and top5 error. In all the plots, we represent the training error and validation error by blue and orange respectively.

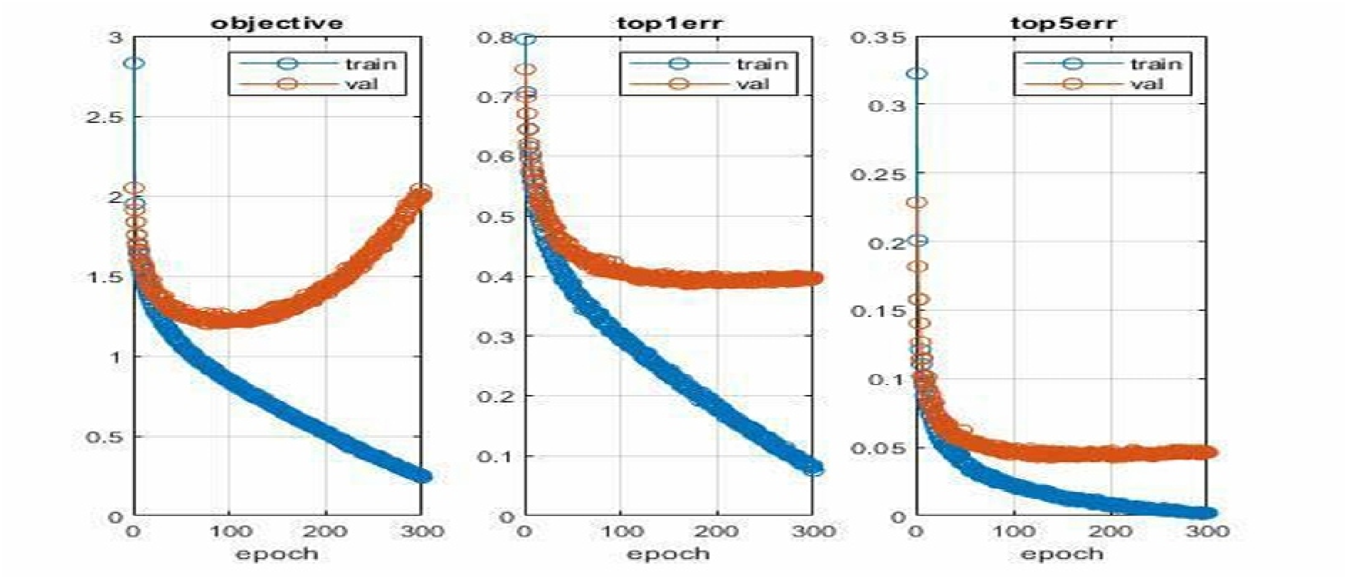


Fig. 6: Objective, Top1error, and Top5error during training

Finally, by using the testing data, we can evaluate our model. The following are an example of classification outputs from the simple network on the CIFAR-10 data.

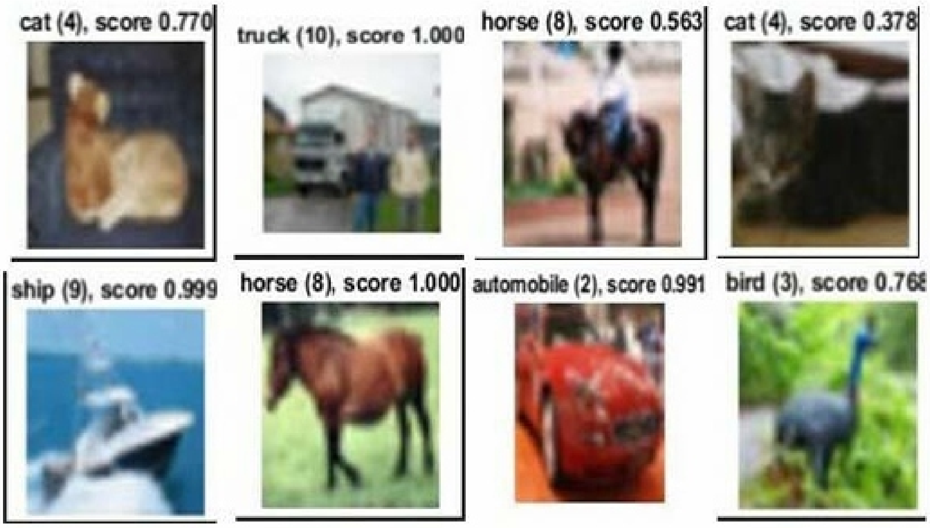


Fig. 7: Some correct recognized outputs

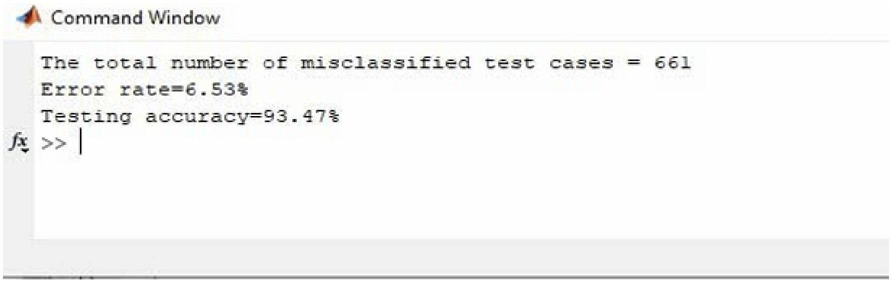
Fig. 8: Some wrong recognized outputs

We can determine the test cases that show failed classifications. The model can't identify some images because of limitations in the input of standard data images. Moreover, missing pixels caused by image compression and image sharpness problems are also reasons for misclassification.

The fourth and final step is to save the model in the disk for reuse. We store the trained model in a MATLAB file format. Hence the saved model can be reused later or easily ported to other environments too.

V. Results and Discussion

Among 10,118 test cases, our model misclassifies total of 661 images after three hundred epochs which correspond to 93.47% recognition rate shown in Table1. The results are pretty good for three hundred epochs and for such a simple model with CPU training and less training time (about 3 hours).



```
The total number of misclassified test cases = 661
Error rate=6.53%
Testing accuracy=93.47%
>>
```

Fig. 9: Error rate and accuracy of our model.

Although there are some images which are difficult to identify, our model will be able to classify them correctly. For example, our model can recognize the following image and classify it as 'deer':

Fig. 10: Correct recognition of the bad image

Table 1: Summary of the experiment

Batch size	No. of epochs	Testing accuracy
100	250	76.82%
70	300	82.28%
60	300	93.47%

Test accuracy 93.47% implies that the model is trained well for prediction. Training set size affects the accuracy increases as the number of data increases. The more data in the training set, the smaller the impact of training error and test error and ultimately the accuracy can be improved.

Conclusion and Future Work

Here we demonstrate a model which can recognize and classify the image. Later it can be extended for object recognition, character recognition, and real-time object recognition. Image recognition is an important step to the vast field of artificial intelligence and computer vision. As seen from the results of the experiment, CNN proves to be far better than other classifiers. The results can be made more accurate by increasing the number of convolution layers and hidden neurons. People can recognize the object from blurry images by using our model. Image recognition is an excellent prototype problem for learning about neural networks, and it gives a great way to develop more advanced techniques of deep learning. In the future, we are planning to develop a real-time image recognition system.

References Références Referencias

1. *Kuntal Kumar Pal, Sudeep K. S.(2016)." Preprocessing for Image Classification by Convolutional Neural Networks", IEEE International Conference on Recent Trends in Electronics Information Communication Technology, May 2021, 2016, India.*

2. *Hayder M. Albeahdili, Haider A. Alwzwazy, Naz E. Islam (2015)."Robust Convolutional Neural Networks for Image Recognition", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 6, No. 11, 2015.*

3. *Jingkun Qin, Haihong E, Meina Song and Zhijun Ren(2018)."Image Retrieval Based on a Hybrid Model of Deep Convolutional Encoder", 2018 the International Conference of Intelligent Robotic and*

Control Engineering.

4. K Sumanth Reddy, Upasna Singh, Prakash K Uttam(2017)." Effect of Image Colourspace on Performance of Convolution Neural Networks", 2017 2nd IEEE International Conference on Recent Trends in Electronics Information & Communication Technology (RTEICT), May 19-20, 2017, India.

5. Krizhevsky, Alex, Sutskever, Ilya, and Hinton, Geoffrey. ImageNet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems 25 (NIPS'2012). 2012.

6. Marc'Aurelio Ranzato, Fu-Jie Huang, Y-Lan Boureau, Yann LeCun, ―Unsupervised Learning of Invariant Feature Hierarchies with Applications to Object Recognition‖ CVPR, 2007.

7. Matthew D Zeiler and Rob Fergus. Stochastic pooling for regularization of deep convolutional neural networks. arXiv preprint arXiv:1301.3557, 2013.

8. Ian J Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. Maxout networks. arXiv preprint arXiv:1302.4389, 2013.

9. Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov‖ Dropout: A Simple Way to Prevent Neural Networks from Overfitting‖ Journal of Machine Learning Research 15 (2014) 1929-1958.

10. Fabien Lauer, Ching Y. Suen, and G´erard Bloch ―A trainable feature extractor for handwritten digit recognition‖ Journal Pattern Recognition 2007.

11. Chen-Yu Lee, Saining Xie, Patrick Gallagher, Zhengyou Zhang, Zhuowen Tu, ― DeeplySupervised Nets ―NIPS 2014.

12. M. Fischler and R. Elschlager, ―The representation and matching of pictorial structures,‖ IEEE Transactions on Computer, vol. 22, no. 1, 1973.

13. Kevin Jarrett, Koray Kavukcuoglu, Marc'Aurelio Ranzato and Yann LeCun ‖ What is the Best MultiStage Architecture for Object Recognition?‖ ICCV'09, IEEE, 2009.

14. Kaiming, He and Xiangyu, Zhang and Shaoqing, Ren and Jian Sun ―Spatial pyramid pooling in deep convolutional networks for visual recognition‖ European Conference on Computer Vision, 2014.

15. Ross Girshick, ―Fast R-CNN ―arXiv preprint arXiv:1504.08083, 2015

16. X. Wang, M. Yang, S. Zhu, and Y. Lin. Regionlets for generic object detection. In ICCV, 2013. 8.

17. Karen Simonyan and Andrew Zisserman ―VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGESCALE IMAGE RECOGNITION‖ arXiv:1409.1556v5 [cs.CV] 23 Dec 2014.

18. C. Couprie, C. Farabet, L. Najman, and Y. LeCun. Indoor semantic segmentation using depth information. Internatinal Conference on Learning Representation, 2013. 2.

19. R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. CoRR, abs/1311.2524, 2013. 4.

20. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. FeiFei. Imagenet: A large-scale hierarchical *image database. In CVPR, 2009. 2.*

21. L. N. Clement Farabet, Camille Couprie and Y. LeCun. Learning hierarchical features for scene labeling. PAMI, 35(8), 2013. 1, 2.

22. Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew Y. Ng ―Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representations‖.

23. "MatConvNet Convolutional Neural Networks for MATLAB" Andrea Vedaldi, Karel Lenc, Ankush Gupta.

image database. In CVPR, 2009. 2.

21. L. N. Clement Farabet, Camille Couprie and Y. LeCun. Learning hierarchical features for scene labeling. PAMI, 35(8), 2013. 1, 2.

22. Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew Y. Ng ―Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representations‖.

23. "MatConvNet Convolutional Neural Networks for MATLAB" Andrea Vedaldi, Karel Lenc, Ankush Gupta.

Instructions for Authors

Essentials for Publishing in this Journal

1 Submitted articles should not have been previously published or be currently under consideration for publication elsewhere.

2 Conference papers may only be submitted if the paper has been completely re-written (taken to mean more than 50%) and the author has cleared any necessary permission with the copyright owner if it has been previously copyrighted.

3 All our articles are refereed through a double-blind process.

4 All authors must declare they have read and agreed to the content of the submitted article and must sign a declaration correspond to the originality of the article.

Submission Process

All articles for this journal must be submitted using our online submissions system. http://enrichedpub.com/ . Please use the Submit Your Article link in the Author Service area.

-

Manuscript Guidelines

The instructions to authors about the article preparation for publication in the Manuscripts are submitted online, through the e-Ur (Electronic editing) system, developed by **Enriched Publications Pvt. Ltd**. The article should contain the abstract with keywords, introduction, body, conclusion, references and the summary in English language (without heading and subheading enumeration). The article length should not exceed 16 pages of A4 paper format.

Title

The title should be informative. It is in both Journal's and author's best interest to use terms suitable. For indexing and word search. If there are no such terms in the title, the author is strongly advised to add a subtitle. The title should be given in English as well. The titles precede the abstract and the summary in an appropriate language.

Letterhead Title

The letterhead title is given at a top of each page for easier identification of article copies in an Electronic form in particular. It contains the author's surname and first name initial .article title, journal title and collation (year, volume, and issue, first and last page). The journal and article titles can be given in a shortened form.

Author's Name

Full name(s) of author(s) should be used. It is advisable to give the middle initial. Names are given in their original form.

Contact Details

The postal address or the e-mail address of the author (usually of the first one if there are more Authors) is given in the footnote at the bottom of the first page.

Type of Articles

Classification of articles is a duty of the editorial staff and is of special importance. Referees and the members of the editorial staff, or section editors, can propose a category, but the editor-in-chief has the sole responsibility for their classification. Journal articles are classified as follows:

Scientific articles:

1. Original scientific paper (giving the previously unpublished results of the author's own research based on management methods).

2. Survey paper (giving an original, detailed and critical view of a research problem or an area to which the author has made a contribution visible through his self-citation);

3. Short or preliminary communication (original management paper of full format but of a smaller extent or of a preliminary character);

4. Scientific critique or forum (discussion on a particular scientific topic, based exclusively on management argumentation) and commentaries. Exceptionally, in particular areas, a scientific paper in the Journal can be in a form of a monograph or a critical edition of scientific data (historical, archival, lexicographic, bibliographic, data survey, etc.) which were unknown or hardly accessible for scientific research.

Professional articles:

1. Professional paper (contribution offering experience useful for improvement of professional practice but not necessarily based on scientific methods);
2. Informative contribution (editorial, commentary, etc.);
3. Review (of a book, software, case study, scientific event, etc.)

Language

The article should be in English. The grammar and style of the article should be of good quality. The systematized text should be without abbreviations (except standard ones). All measurements must be in SI units. The sequence of formulae is denoted in Arabic numerals in parentheses on the right-hand side.

Abstract and Summary

An abstract is a concise informative presentation of the article content for fast and accurate Evaluation of its relevance. It is both in the Editorial Office's and the author's best interest for an abstract to contain terms often used for indexing and article search. The abstract describes the purpose of the study and the methods, outlines the findings and state the conclusions. A 100- to 250-Word abstract should be placed between the title and the keywords with the body text to follow. Besides an abstract are advised to have a summary in English, at the end of the article, after the Reference list. The summary should be structured and long up to 1/10 of the article length (it is more extensive than the abstract).

Keywords

Keywords are terms or phrases showing adequately the article content for indexing and search purposes. They should be allocated heaving in mind widely accepted international sources (index, dictionary or thesaurus), such as the Web of Science keyword list for science in general. The higher their usage frequency is the better. Up to 10 keywords immediately follow the abstract and the summary, in respective languages.

Acknowledgements

The name and the number of the project or programmed within which the article was realized is given in a separate note at the bottom of the first page together with the name of the institution which financially supported the project or programmed.

Tables and Illustrations

All the captions should be in the original language as well as in English, together with the texts in illustrations if possible. Tables are typed in the same style as the text and are denoted by numerals at the top. Photographs and drawings, placed appropriately in the text, should be clear, precise and suitable for reproduction. Drawings should be created in Word or Corel.

Citation in the Text

Citation in the text must be uniform. When citing references in the text, use the reference number set in square brackets from the Reference list at the end of the article.

Footnotes

Footnotes are given at the bottom of the page with the text they refer to. They can contain less relevant details, additional explanations or used sources (e.g. scientific material, manuals). They cannot replace the cited literature.

The article should be accompanied with a cover letter with the information about the author(s): surname, middle initial, first name, and citizen personal number, rank, title, e-mail address, and affiliation address, home address including municipality, phone number in the office and at home (or a mobile phone number). The cover letter should state the type of the article and tell which illustrations are original and which are not.

Notes: