International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-2, July 2019

Classifying Handwritten Digit Recognition using CNN and PSO

Priyanka B. Barhate, G. D. Upadhye

Abstract: ***A normal human can easily recognize any written or typed or scanned text, numbers, etc., but when it comes to a machine, it is difficult to find out what exactly that given text or numbers. It will be difficult to recognize a handwritten digit for a machine. Many machine learning methods were used to fix the handwritten digit recognition issue. It is growing in more convoluted domains, so its training complexity is also increasing. To overcome this complexity problem, many algorithms have been implemented. In this paper, the Convolutional Neural Network (CNN) and Particle Swarm Optimization (PSO), those two approaches do use for recognition of the isolated handwritten digit. Customized PSO is used to reduce the overall computation time of the proposed system. The customized PSO used with CNN, to decreases the required number of epochs for training. It is used to identify digits in the MNIST handwritten digital database to predict the number. The system has achieved an average of 94.90% accuracy.***

Index Terms: ***Pattern Recognition, Handwritten Digit Recognition, Convolutional Neural Network, Particle Swarm Optimization, Machine Learning.***

I. INTRODUCTION

Computer digitization technology is used in a relatively new field and pattern recognition is now one of the challenging tasks for human handwriting recognition. Some practical applications of handwritten recognition are postal zip code recognition, bank cheque, writer identification, form processing, etc. Recognition of pattern is a common machine learning field used to solve different real-life issues. Pattern recognition is a popular domain in machine learning which is used to resolve various real-life problems. Many researchers have been worked on Pattern recognition and implemented a number of different pattern recognition algorithms in machine learning. We converge on two methods for handwriting digit recognition, which are CNN and PSO. The MNIST dataset which contains images of the handwritten digit is used in this paper. CNN accomplish with a multilayer of the simple neural network. It is the most important algorithm to classify handwritten digit along with customized PSO. It compares the optimization capability of CNN and PSO in terms of training time and recognition accuracy.

Revised Manuscript Received on 30 July 2019.

***** Correspondence Author

Ms. Priyanka B. Barhate*, Computer Engineering Department, JSPM's Rajarshi Shahu College of Engineering, Tathawade, Pune, India..

Mr. G. D. Upadhye, Computer Engineering Department, JSPM's Rajarshi Shahu College of Engineering, Tathawade, Pune, India..

II. LITERATURE SURVEY

Meer Zohra et al. [2] presented the analysis of accuracies and performance measures of algorithms Support Vector Machine (SVM), K-Nearest Neighbor (KNN) and Convolutional Neural Networks (CNN). CNN, KNN, and SVM have achieved an accuracy of 99.4%, 97.1%, and 97.9% respectively.

Shengfeng Chen et al. [5] compared the performance of five machine learning classifier models name as Neural Network, K-Nearest Neighbor (K-NN), Random Forest, Decision Tree and Bagging with gradient boost. Using K-NN classifier result shown outperform Neural Network with significant improved computational efficiency without sacrificing performance. They both outperformed the other classifiers: Random Forest, Decision Tree and Bagging with gradient boost.

Mahmoud Abu Ghosh et al. [6] focused on methods of the neural network such as the Deep Neural Network (DNN), CNN, and the Deep Belief Network (DBN). Herein, using all three methods, the performance of identification calculates toward the effective system of the algorithm because authors were used random dataset. So, the shuffled dataset provides less accuracy and overwhelms time.

In [7], authors created a model capable of recognizing and identifying handwritten numbers from the image. They used the Convolution Neural Network ideas. Authors used Machine Learning and Neural Networks ideas.

In [9], Retno Larasati et al. proposed a new algorithm, ensemble neural network that combined with an ensemble decision tree. This ensemble was used for the classifier to classify MNIST and USPS dataset. The model gave an overall accuracy of 84% for MNIST dataset and 74% for USPS dataset with high computation time.

Nurul Ilmi et al. [10] presented the Local Binary Pattern described as a feature extraction method and K-Nearest Neighbor as a classification algorithm. This system was performed on a handwritten image C1 type used by the Indonesian General Elections Commission to encourage the member of the council to enter the election outcome into the database. An accuracy with 89.81% and 70.91% was achieved on the MNIST dataset and C1 form respectively.

In [12], Caiyun Ma et al. proposed a handwritten approach to digit recognition based on particular extraction a nd profound analysis of multi-functions. The approach had an accuracy of about 94.2%.

J. Pradeep et al. [14] proposed a handwritten characters recognition system for English alphabets with no multilayer feed-forward neural network extraction feature. This method achieved average recognition success rates.

Table 1: Relative Description Table of Different Techniques

Work Reference	Year	Technique	Result (%)
Shengfeng Chen et al. [4]	2018	K-Nearest Neighbor	96.7
Nikolao Toulgaridis et al. [8]	2017	Neural Network	91
Retno Larasati et al. [9]	2017	Ensemble Neural Network	84
Nurul Ilmi et al. [10]	2016	K-Nearest Neighbor	70.91
Caiyun Ma et al. [12]	2015	Multi-feature extraction	94.2
Ujjwal Bhattacharya et al. [15]	2009	MLP	70.85

III. DATASET

MNIST database is used in this proposed system. It consists of 60000 training set of images and 10000 testing set of images. It is a subset of a wider collection available from NIST. Pixels are organized row-wise. Each image size is 28×28 pixels, with a total of 784 pixels. The MNIST dataset is out there as binary files hold on in associate IDX file format. We extract the isolated written pictures from every file and store them in .png form.

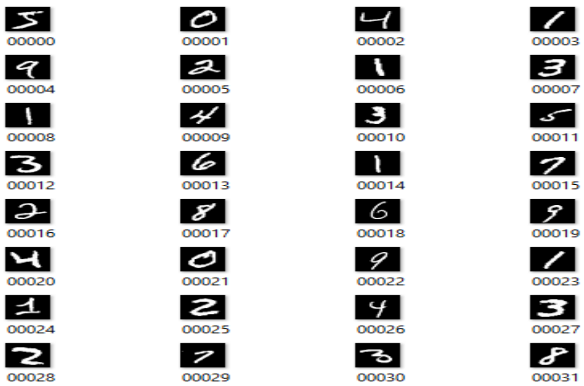

Figure 1: Sample of MNIST Handwritten Digits

IV. PROPOSED METHODOLOGY

A. *Convolutional Neural Network*

The Convolutional Neural Network (CNN) determines the complex representation of visual data using large amounts of data. It is basically inspired by a human visual system because a human can see colors. Convolutional networks identify images as volumes that mean three-dimensional objects. Because digital color images have a red, green and blue (RGB) encoding and mixing those three colors to produce the color spectrum. Each layer performs a dot product of input pixel and moved toward the next layer [10]. CNN has the following some layers of operations along with basic CNN basic structure shown in Fig. 2.

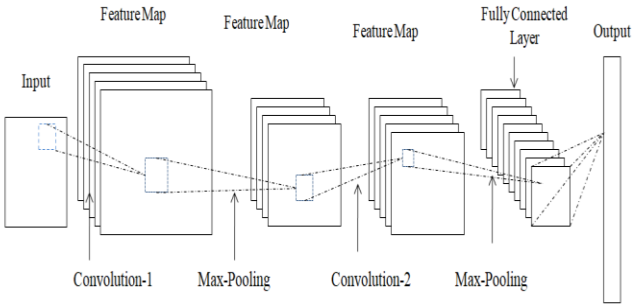

Figure 2: CNN Basic Structure

1) Convolution layer

This layer is the center building section of a CNN. The layer's parameters incorporate a collection of learnable filters, which have a little receptive field however extend through the complete extent of the data size [7].

2) Pooling layer

This layer is also an essential unit of a CNN used to furthe r reduce the spatial dimensionality of a convolution layer out put, thus minimizing the network's amount of parameters an computational complexity and controlling overfitting [5].

3) Fully Connected layer

This layer is used to combine convolution and pooling layer characteristics to generate a likely class to score for the input image classification [5].

B. *Particle Swarm Optimization*

Particle Swarm Optimization (PSO) is a community based method of stochastic optimization created by Dr. Kennedy and Dr. Eberhart in 1995, motivated by as flock of birds or a social behavior school of fish. The potential solutions in PSO are called particles, the present optimal level particles travel through to the problem space and connected with the best alternative, i.e. fitness, every particle keeps a record of all its coordinates. This fitness value becomes called pbest. If a particle takes the entire population as its neighbors, its value is the best global value and is called gbest. Fig 3 shows the flowchart of the PSO algorithm. In equation 1 and 2 [17, the common type of particle swarm optimizer is described. The formulas used to adjust CNN parameters are the followi ng concepts:

$$v_{id}(t+1) \leftarrow w \times v_{id}(t) + c_1 r_1\big(P_{id}(t) - x_{id}(t)\big) + c_2 r_2\big(P_{gd}(t) - x_{id}(t)\big) \quad (1)$$

$$x_{id}(t+1) \leftarrow x_{id}(t) + v_{id}(t+1) \quad (2)$$

Where,

- v_{id} is the particle velocity i together with size d,
- x_{id} is the position of particle i in d,
- c_1 is as weight applied to the part of cognitive learning,
- c_2 is as impact of the social learning part is comparable in weight,

- r_1 and r_2 are random number in the range of zero and one (0,1), generated individually,
- P_{id} is the prior best particle place also known as pbes t, i.e. the particle's position,
- P_{gd} is the best place for the whole population, also re cognized as the gbest.
- w is the weight of inertia.

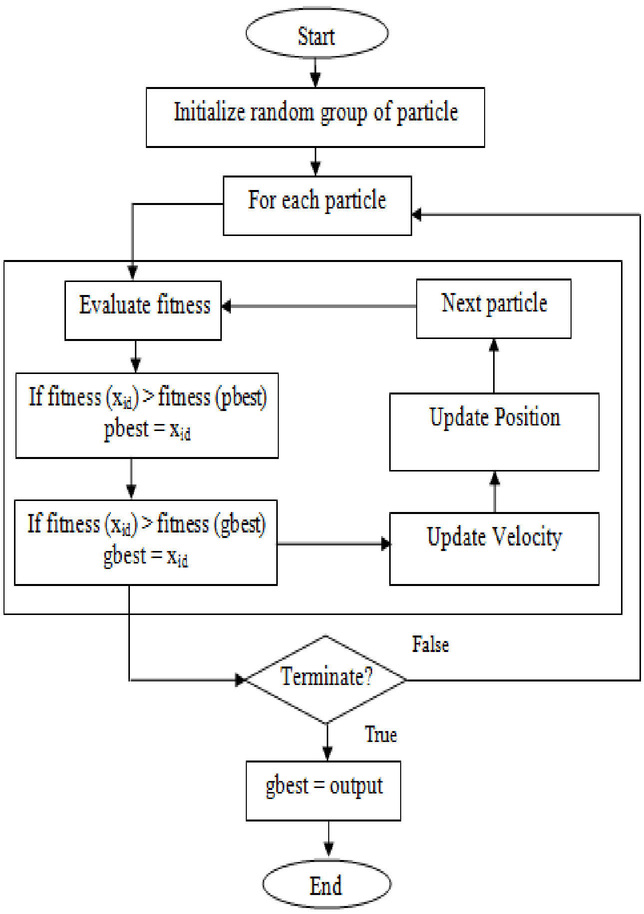

Figure 3: PSO Flowchart

V. PROPOSED SYSTEM ARCHITECTURE

In the proposed system architecture, the input is as an image which will be in .jpg or .png format. The input image is then pre-processed to transform the image into a grayscale image. After the preprocessing dataset is trained by using CNN consisting of options two primary layers called convol ution and max-pooling layer and finishing with a fully conne cted layer. These layers are connected to each other with weights. PSO randomly initializes the particle from the output vector of CNN and then for each particle it evaluates the fitness by pbest and gbest values by continuously updating positions and velocities of the particle.

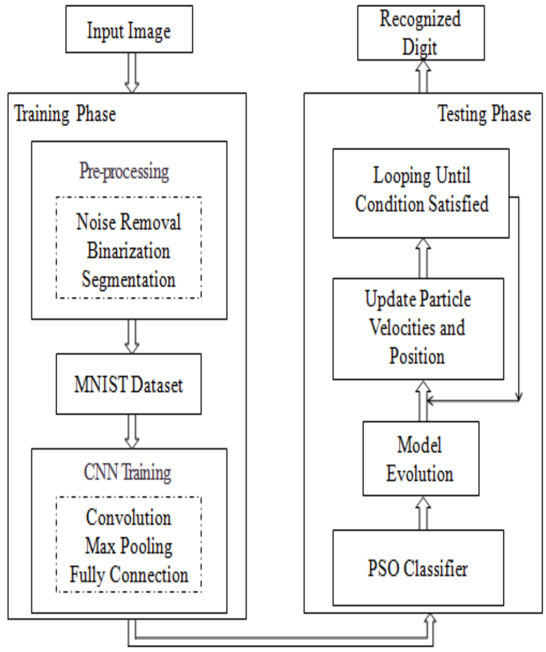

Figure 4: Proposed System Architecture

VI. VISUALIZATION OF PROPOSED ARCHITECTURE

In our system, the image input size is $28 \times 28 \times 1$ matrix, i.e. breadth × height × depth. CNN consists of two convolutional layers and each layer followed by max-pooling layers. The input image is then provided to the convolution layer. These layers extract a higher-level feature that recovers the data for the pixels and consists of collection of separate layers. Every layer is convolved to the image individually and we end up with 5 function maps of size $28 \times 28 \times 1$. This will decrease the number of different parameters in this system and increase the efficiency of the computation. Changeable size kernels that were used in the presented CNN model's intermediate layers. Initially, all kernels are allocated a randomized value. The value of the kernels is changed in each epoch to extract the image's features.

Max pooling operation is performed to reduce dimensionality size. This layer is implemented after the first convolutional layer. There are three ways to do this pooling: Max, Min and Average pooling. The proposed system used max-pooling. To reduce the overall the sizes of the feature maps, it takes the maximum of the block they pool. In this layer, a 3×3 kernel is chosen with stride value 1 and then the kernel is put into the output of the previous convolutional layer. After several layers of convolutional and max-pooling, we flatten the output class into vectors and then provide these to a next layer i.e. fully connected layer. Neurons in a layer that is fully connected complete links to all prior layer activations, meaning that all prior layer neurons are taken. In the proposed system's testing phase, PSO is used. PSO randomly initializes the particle from the output vector of CNN and then for each particle it evaluates the fitness by pbest and gbest values by continuously updating particle positions and velocities.

We used equations 1 and 2 to calculate velocity and position. The proposed CNN model followed by PSO shown in Fig. 4. The proposed customized PSO classifier is to keep records about the best solutions discovered and used as the best position. In the proposed method we have set the parameter as: number of particles=32, number of iterations=300, weights=0.9, acceleration coefficient=2, max velocity of particle=0.005, velocity decay=1.

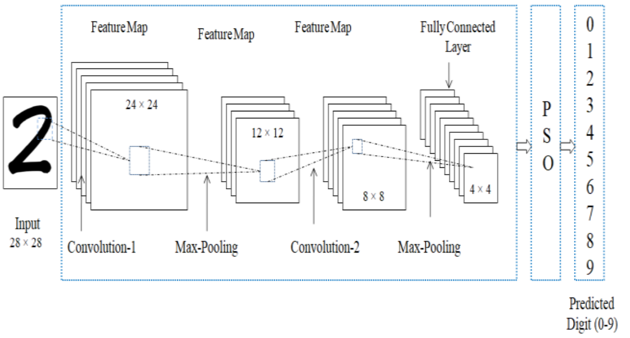

Figure 5: Proposed System Visualization

A. Algorithm/ Pseudo Code

By combining CNN with the PSO, a new algorithm referred to as customized PSO algorithm is specifying in this paper. The customized PSO algorithm can be summarized as follows:

Algorithm 1: CNN Algorithm

1: Take the input image
2: Initialize CNN model
3: Apply a convolution (C1) operation to the input
4: Do Max-pooling
5: Again apply convolutional (C2) Operation
6: Do max-pooling again
7: Change negative pixel numbers by zero in ReLU Module.
8: Fully connection

Algorithm 2: Customized PSO Pseudo Code

1: Randomly initialize CNN parameters
2: Model Evaluation
3: **while** terminating condition is not reached
4: **if** fitness(x_{id}) > fitness(gbest)
5: gbest = x_{id}
6: **end if**
7: **if** fitness(x_{id}) > fitness(pbest)
8: pbest = x_{id}
9: **end if**
10: Update velocity with Eq. 1.
11: Update position with Eq. 2.
12: **end while**

VII. RESULT OF EXPERIMENT

The experimental Model is implemented in Python programming languages. The experimental setup computer is an Intel Core i3-6006U CPU @ 2.00GHz with 4 GB RAM. The confusion matrix is used when working with algorithms. It is a tabular form of execution of matrices and is checked on the collection of testing data that identifies the respective true values. Here the system has achieved an average of 94.90% accuracy. In Fig. 6, the confusion matrix is displayed.

```
Confusion matrix:
[[87  0  0  0  1  0  0  0  0  0]
 [ 0 88  1  0  0  0  0  0  1  1]
 [ 0  0 85  1  0  0  0  0  0  0]
 [ 0  0  0 79  0  3  0  4  5  0]
 [ 0  0  0  0 88  0  0  0  0  4]
 [ 0  0  0  0  0 88  1  0  0  2]
 [ 0  1  0  0  0  0 90  0  0  0]
 [ 0  0  0  0  0  1  0 88  0  0]
 [ 0  0  0  0  0  0  0  0 88  0]
 [ 0  0  0  1  0  1  0  0  0 90]]
```

Figure 6: Confusion Matrix

The CNN model loss and accuracy is shown in Fig. 7 and our proposed systems combined methods means CNN and system accuracy graph is shown in Fig. 8. Best particle position is of PSO shown in Fig. 9.

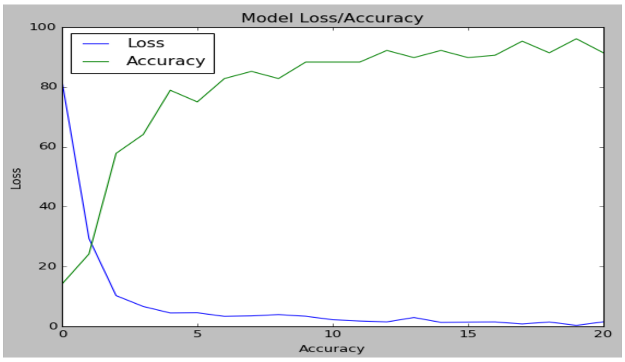

Figure 7: CNN Loss and Accuracy Graph

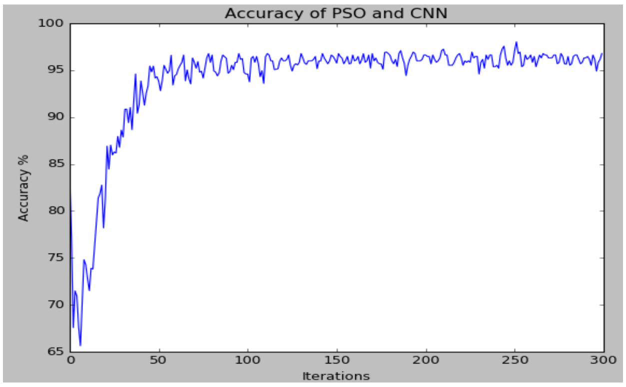

Figure 8: System Accuracy Graph

DOI: 10.35940/ijrte.B3675.078219
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

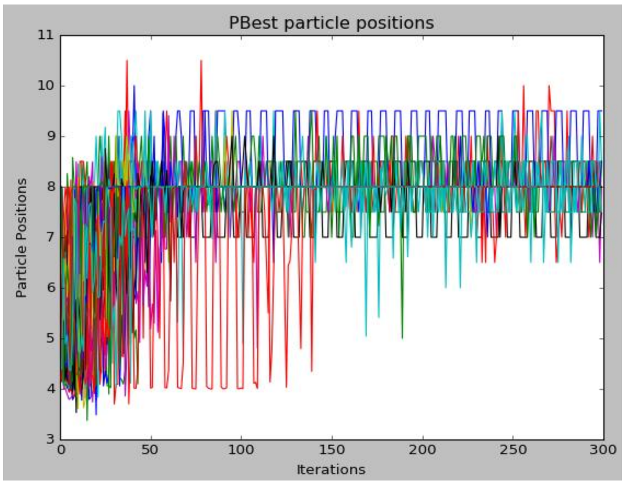

Figure 9: Particle Position Graph

VIII. CONCLUSION

In the presented paper, a handwritten digit method for recognition established at the CNN and PSO algorithm. These algorithms help to reduce the not so effective long training time problem. This paper has concentrated on customized PSO with CNN based model for recognizing handwritten digits. Training of the CNN model is a very difficult task and this takes a long time for computation. In addition, hardware becomes a big issue in training these models. Reducing the dependency on CPU by decreasing the learning rate and makes the task more achievable on traditional computing systems. Consequently, PSO is used in the proposed system to overcome this type of problem. Customized PSO reduced the training time and also reduced increase accuracy by a significant amount. There are some efforts in the future job that will be searched for further improvement, primarily on PSO efficiency with other fitness function in the area of pattern recognition. In the future, this system can be upgraded to recognize the handwritten alphabet

REFERENCES

1. The MNIST database of handwritten digits, http://yann.lecun.com/exdb/mnist/
2. Meer Zohra, D. Rajeswara Rao, "A Comprehensive Data Analysis on Handwritten Digit Recognition using Machine Learning Approach", IJITEE, Vol. 8, Issue-6, 2019.
3. D. T. Mane, U. V. Kulkarni, "Visualizing and Understanding Customized Convolutional Neural Network for Recognition of Handwritten Marathi Numerals", ICCIDS, 2018.
4. Shengfeng Chen, Rabia Almamlook, Yuwen Gu, Dr. Lee wells, "Offline Handwritten Digits Recognition Using Machine learning", ICIEOM 2018.
5. Rezoana Arif, Md. Abu Siddique, Mohammad Mahmudur Rahman Khan, Mahjabin Rahman Oishe, "Study and Observation of the Variations of Accuracies for Handwritten Digits Recognition with Various Hidden Layers and Epochs using Convolutional Neural Network", 4th iCEEiCT, 2018.
6. Mahmoud M. Abu Ghosh, Ashraf Y. Maghari, "A Comparative Study on Handwriting Digit Recognition Using Neural Networks", ICPET, 2017.
7. Nimisha Jain, Kumar Rahul, Ipshita Khamaru, Anish Kumar Jha, Anupam Ghosh, "Hand Written Digit Recognition using Convolutional Neural Network (CNN)", IJIACS, 2017.
8. Nikolaos Toulgaridis, Eleni Bougioukou, Theodore Antonakopoulos, "Architecture and Implementation of a Restricted Boltzmann Machine for Handwritten Digits Recognition", 6th MOCAST, 2017.
9. Retno Larasati, Dr. Hak KeungLam, "Handwritten Digit Recognition using Ensemble Neural Networks and Ensemble Decision Tree", IEEE, 2017.
10. Nurul Ilmi, Tjokorda Agung Budi W, Kurniawan Nur R, "Handwriting Digit Recognition using Local Binary Pattern Variance and K-Nearest Neighbor Classification", 4th ICoICT, 2016.
11. Ankita Mishra, D. S. Singh, "Handwritten Digit Recognition Using Neural Network Approaches And Feature Extraction Techniques: A Survey", IJCESR, 2016.
12. Caiyun Ma, Hong Zhang, "Effective Handwritten Digit Recognition Based on Multi-feature Extraction and Deep Analysis", FSKD, IEEE, 2015.
13. Aiquan Yuan, Gang Bai, Po Yang Yanni Guo, Xinting Zhao, "Handwritten EnglishWord Recognition based on Convolutional Neural Networks", ICFHR, 2012.
14. J. Pradeep, E. Srinivasan, S. Himavathi, "Neural Network based handwritten character recognition system without feature extraction", ICCCET, 2011.
15. Ujjwal Bhattacharya, B.B. Chaudhuri, "Handwritten Numeral Databases of Indian Scripts and Multistage Recognition of Mixed Numerals", Vol. 31, No. 3, IEEE, 2009.
16. Daniel Bratton, J. Kennedy, "Defining a Standard for Particle Swarm Optimization", Swarm Intelligence Symposium (SIS), IEEE, 2007.
17. Nasser Omer, Siti Mariyam Shamsuddin, "Handwritten Digits Recognition using Particle Swarm Optimization", 2nd Asia International Conference on Modelling & Simulation, 2008.
18. James Kennedy, Russell Eberhart, "Particle Swarm Optimization", Proceedings of ICNN'95, 1995.

AUTHORS PROFILE

Priyanka B. Barhate received the degree in diploma of Information Technology from Government Polytechnic, Jalgaon, in 2013 and Bachelor of Engineering degree in Information Technology from Dr. D. Y. Patil College of Engineering, Pune, in 2016. Currently she is pursuing Master of Computer Engineering at JSPM'S Rajarshi Shahu College of Engineering, Tathawade, Pune, affiliated to Savitribai Phule Pune University. She is currently working as Lecturer in Department of Information Technology, at G. S. Moze College of Engineering, Pune. She has published 2 research papers in reputed international journals and conferences. Her research interest is Machine Learning and Artificial Intelligence.

Gopal D. Upadhye received the Bachelor of Engineering in year 2004 from Government College of Engineering, Aurangabad and Master of Technology from Dr. B. A. T. University, India in year 2011. Currently he is pursuing Ph.D. degree from S. G. G. S. I. E. & T., Nanded, Maharashtra, India. He is an ISTE member and currently working as Assistant Professor in Department of Computer Engineering, Rajarshi Shahu College of Engineering, Tathawade, Pune, affiliated to Savitribai Phule Pune University, since 2008. He has 12 years of teaching experience. He has published 4 research papers in reputed international journals and conferences. His area of research is Machine Learning for Pattern Classification.