Semantic Segmentation of color images via Feature Extraction Techniques

Cheruku Sandesh Kumar 1, Vinod Kumar Sharma 2, Abhay Sharma 3
Ashwani Kumar Yadav 4 and Archek Praveen Kumar 5

1 Assistant Professor, ASET, Amity University, India.
2 Assistant Professor, ASET, Amity University, India.
3 Assistant Professor, ASET, Amity University, India.
4 Assistant Professor, ASET, Amity University, India.
5 Associate Professor, Malla Reddy College of Engineering for Women, India.

sandeshkumarc7@gmail.com
vinod.gwalior@gmail.com
abhaysharma@gmail.com
ashwaniy2@gmail.com
archekpraveen@gmail.com

Abstract. In this research semantic segmentation (SS) by deep neural network is used. Segmentation of contextual information with the use of patch-patch between the regions and background. The patch-patch context is done with the help of ANNs (Artificial Neural Networks) and Adaboost to identify the semantic correlation of adjacent patches. Efficient piece-wise training of deep model is then assigned to eschew continuous inference through background propagation. For Patch-background context capturing the network is designed with the help of multi-scale image input and slide Pyramid pooling for the efficient performance. In this model we have used both ANNs and Adaboost. The simulation results are verified in the semantic segmentation dataset like MATLAB contextual modeling. The proposed technique will gave better accuracy. The evaluation metrics of invented method has been done in terms pixel accuracy. The model achieves outperforming innovation than the other algorithms on semantic segmentation.

Keywords: Deep Learning, ANN, Adaboost, Contextual Models, SS-Semantic Segmentation.
1. Introduction

Image SS has a category label for image pixel that plays important role in the complete scene understanding of an image. The related approaches like ANNs have pixel-level labeling [1] [2] [17]. There are many ANNs methods deep ANNs [2] [3] is widely used because it has good computational efficiency, dense prediction and end-to-end training. Contextual information or data has the main cues for scene understanding areas. The spatial context is applied in semantic compatibility relations between one object and its adjacent objects/stuff where compatibility relation is a point of co-occurrence of visual patterns [19]. Considering on a highway, a bottle on a stool, context encodes incompatibility relations would be the example a boat on the highway. The relations can have good scales for assumption, object part-to-part relation and part-to-part object relations. Contextual information is mainly in finding cue for isolated object that has visual uncertainty. The spatial context is a broad area of research has given in [4].

In our findings we try to explore spatial context to obtained good results of semantic segmentation. Spatial context has patch-to-patch and patch-to-background context. The patch-to-patch context has semantic relationships on visual patterns of two stuffs of a image. Patch-to-background context had semantic relations in image patch and whole background region. In our research we have explored contextual relations based on ANN and Adaboost. ANN captures semantic correlation on nearby patches [3] [5] [6] [7].

In our model we used ANN and Adaboost technique to refine our samples of resolution prediction to sharpen the object boundaries and local smoothness of images. Fig 3.1 shows the prediction of our method. The patch-background context is traversed in this regard. ANN based techniques of multi-scale image network gives good output when compared to recent semantic segmentation methods [1] [8]. In this model the use of multi-scale networks to encode background data and then slide window on activation maps is applied to encapsulate intelligence from background regions of various sizes. The piece-wise learning of ANN and Adaboost ignores continuous inference on back propagation learning of deep model [9].
2. Feature Classification using ANN’s

There are many nodes which act on the model. The algorithm used for ANN is given in Fig. 2.

Take all the features and separate according to the classes done in LDA

Each class is feeded as input to the input layer of ANN model

Specify the set of weights to the input signals and every neuron

Add all the input feature vectors with its synaptic weights

Generate the output by considering a threshold $V_k$

**Fig. 2** – ANN Algorithm.
The output layer generates four types of outputs – true positive, true negative, false positive and false negative. The weights are added. The nodes take the input and process it to the next unit. Activation node is the output node and the value at output node is called node value. ANN architecture technology is equivalently related to brain. There are many processing elements in the various ANN models. Every unit consists of neurons which act on the signal inputs. With the help of proper connections, artificial neurons behave as biological neurons [20].

Back Propagation Algorithm (BPA) – BPA is a neural network which is used to recognise the image. BPA has three layers – input layer, hidden layer and output layer. Input and output layer presents the recognition. The output is back propagated from the output to hidden layer. The weights are varied to reduce the occurrences of errors [10]. The algorithm uses nodes as specified. In BPA there are 4233 nodes out of which 1344 are input nodes, 2768 are hidden nodes and the rest 121 are output nodes. The hidden layers are determined by the mapping of one dimensional space to another dimension. If m dimension is changed to n dimension then (m+1) hidden layers are required [11]. The recognition is done by testing and the same process is used for image recognition. One output node represents one output which may have objects, patterns, image regions and pixels. Training and testing code is written in this feature classification. The flow chart of feature classification by ANN is shown in Fig 3.
Fig. 3. ANN Flow Chart.

1. Start
2. Set the training file, output file and the training features of the neural network
3. Set the weights and the threshold needed in neural network
4. 
   - Finalize the node values at the output
   - Finalize the node error values
   - Finalize the weights by adding the momentum factor on every weight
5. Iterations
6. Finalize the weights of Neural network as the output
7. Stop
Multilayer Perceptron (MLP) deals with probabilities of estimated classes of image. MLP is mostly used for recognition of image with higher efficiency [12]. These acoustic a-posteriori probabilities are used for local acoustics.

All the a-posteriori probabilities which are generated by Hidden Markov Model (HMM) are used in image recognition. MLP phoneme probabilities are estimated to recognize the signals of image. The name signifies multiple layers. Each layer carries out the subtask of processing through parallel processing. This increases the speed of recognition. This can be functionally embedded with ANN. Time Lagged Recurrent Neural Network (TL-RNN) purely depends on RNN. RNN always takes vectors as inputs. All the features selected can be fed to the RNN network. This makes use of a function which is non-linear and is used to hide the input and generate the output. A neuron is assigned to every input and the neurons are multiplied to weights. Feedback to the input is given with delay. TL-RNN provides more classification accuracy as compared to MLP. The training procedure is complex in TL-RNN when compared with MLP. TL-RNN training algorithm is more sensitive and this in turn produces more errors.

3. Feature classification using adaboost

Adaboost is a decent simulation algorithm used in complicated areas like object classification, protein stricter analysis, speech processing etc. Adaboost is used in areas like biology, chemistry, physics and computers. Adaboost was designed in the year 1992 and became popular very early as the state of art for classification of complex and high dimensional data.

Adaboost mainly holds the changes detected even in a small training subset and finalizes those changes against a border corpus which had major applications across the technical world. A supervised design is used in Adaboost which classifies the unnoticed data which depends on labelled training data. Initially a group of training data is analysed where in a model can be designed which can be used for inside and outside the training data. The achievement needed to design a high quality training set is completely prudent, especially when high volume of data is classified. Adaboosts learning algorithm is occasionally expenditure effective technique for feature classifications. Adaboost have been recently used my many popular companies. Adaboost is designed on statistical learning theory. In 1992 Boser, Guyon and Vapnik designed the approach of Adaboost which was more suitable for algorithms designed. Adaboost was popular in recognizing hand writing and later it was used for all other methods, mainly in image analysis for image processing. Earlier it was found that even for image analysis in image processing Adaboost over comes important classification techniques like – ANN and pattern recognition. Adaboost technique has been used for
The classification of machine learning problems and results provided better acceptability and accuracy. In 1998 Joachmis, Dumais, Drucker proved excellent over other machine learning methods for image processing. Adaboost can be classified by many algorithms like - Reuters dataset, K-Nearest neighbour algorithms etc. Now a days Adaboost is used and implemented in all technical fields and also in academia. WHO (world health organization) and HDC (Health discover organization) used Adaboost in image processing tool for recognizing various images and licensed to Pfizer and MATLAB and PEKO software tools [17] 18.

Fig. 4. Framework of Adaboost classification.

The Fig.4 shows the framework of Adaboost classification. The first frame shows the training part, second one act as a validation and the final one classification. Training
uses a set of labelled images. Validation computes the accuracy of the design on unclassified data. The labelled data is directly used to find out the estimated accuracy which finally evaluates the error. The designed model is user friendly to classify the features in real time. Adaboost divides the input data sets in to various subsets, e.g. images are divided into digital images, film images, classical images etc. Similarly Adaboost is grouped in to all the areas of digital image processing. The Adaboost algorithm divides the data in to two parts - training and testing parts. Both of these with proper dimension classifications are done with greater efficiency and accuracy. Adaboost is an algorithm designed for human machine interface. This algorithm is used for classification of feature vectors with respect to the parameters extracted. N-dimensional state space vectors with its weights and coordinates of every feature are considered for classification. Hyperplane is used for classifying and grouping the classes. The detection and regression is done by using Adaboost. Adaboost efficiently works in larger dimensional state space. Adaboost is economic. Support vectors are defined as the group of vectors which are collectively decided from a function which are specifically collected for easy operations. There are various kernel functions for collecting the support vectors [21]. Adaboost is flexible in many applications. This works with decision boundaries, where the vectors with various classes are separated and grouped collectively using hyper plane. The differentiation line is a boundary which separates various classes of features.
The hyper plane is designed by using N dimensions. This always supports the vectors which are relevant to own class. Adaboost classification are given as - C- Adaboost (classification Adaboost) sort one, Nu- Adaboost sort two, R- Adaboost (regression Adaboost) sort one and R- Adaboost (regression Adaboost) sort two. Adaboost classifies and supports various features dataset for greater recognition accuracy. The Adaboost flowchart is shown in Fig 5.

4. Implementation

All the neural network layers are trained with the help of Stochastic Gradient Decent (SGD). The high-level features are extracted by Activation map-Net. As discussed
earlier the large number of pairwise connections wastes the Graphical Processing Unit (GPU) memory. Hence we parallelize the learning of pairwise connections asynchronously which in turn simplifies the GPU speed of 3.46 images/second.

4.1. Effective Training

For every node in Conditional Random Field (CRF), We perform sampling of pairwise connections. In sampling process we minimize the pairwise connections which sped up the learning without degrading the performance. Even with sampling there are large connections.

4.2. Gradient Update Asynchronously

This approach is mainly used for large data sets. The training set consists of huge number of similar instances instead of sweeping through all of them we are updating. In one SGD iteration we implement multiple sub iterations for gradient update of pairwise network and gather the gradients for Activation map-Net.

5. Practical experiments with validation set on matlab contextual modeling

The simulation of the introduced technique are verified on challenging SS test sets. They are MATLAB contextual modeling that unfolds different kind of scene images like counting indoor and outdoor scenes etc. The simulation of the proposed technique got outperforming performance on the above stated test sets. The execution is done on MATLAB contextual modeling of visual object classes” data set. The comparison has been done for various techniques with outstanding performance. The proposed model is trained using images that gave accuracy score of 69.2%, outperforming other techniques. The accuracy scores are shown in Table 1. The original dataset contains 1464, 1449 and 1456 images for training, validation, and testing respectively. The prognostic results of the proposed method are tested on Matlab contextual modeling. Fig. 6 shows – original image and prognostic.

![Input Image](image1.png) ![Prognostic](image2.png)

(a) Input Image (b) Prognostic

**Fig. 6.** Prognostic examples of our approach on MATLAB contextual modeling.
5.1 Results on Semantic Segmentation through ANN

The execution is done on MATLAB contextual modelling of visual object classes’ data set. The ANN method with classifiers is trained using images that gave accuracy score of 69.2%. The accuracy scores are shown in Table 1.

Table 1. Individual category results of ANN on the MATLAB Contextual Modeling accuracy scores.

| Classification Accuracy (%) |
|-----------------------------|
| Method | Chair | Table | Potted |
| ANN    | 45.6  | 69.2  | 43.3   |

5.2 Results on Semantic Segmentation by Adaboost

The execution is done on MATLAB contextual modelling of visual object classes’ data set. The DPN method with classifiers is trained using images that gave accuracy score of 59.9%. The accuracy scores are shown in Table 2.

Table 2. Individual category results of Adaboost on the MATLAB Contextual Modeling accuracy scores.

| Classification Accuracy (%) |
|-----------------------------|
| Method | Chair | Table | Potted |
| Adaboost | 28.8  | 59.9  | 57.7   |

6. Conclusion

In this research we used ANN and Adaboost technique to investigate context information to get efficient SS. This can be done by ANN related pair-wise potentials to model semantic relationships between regions of an image. The practical results were simulated in couple of datasets mentioned in this research which gave best performance. The technique introduced is well suited for all other SS related works.
In all, classification accuracy rates are generated and best results are selected for design of deep model. Hence after many experimental techniques, ANN gave the best classification accuracy of 69.2%. Adaboost gave accuracy score of 59.9% outperforming among other techniques. The distinct paths where the future research can focus on adaptive learning, apply deep model in other functions like, human pose estimation and face alignment, etc. To refine the boundaries. We can further use adaptive learning and VGG-19 layers network can be used.

7. References

[1] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *IEEE T. Pattern Analysis & Machine Intelligence*, 2013

[2] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. *In Proc. IEEE Conf. Comp. Vis. Pattern Recognition*, 2015

[3] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected CRFs. *In Proc. Int. Conf. Learning Representations*, 2015

[4] G. Heitz and D. Koller. Learning spatial context-Using stuff to find things. In Proc. European Conf. Computer Vision, 2008

[5] A. G. Schwing and R. Urtasun. Fully connected deep structured networks. *ArXiV*, 2015

[6] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. Torr, Conditional random fields as recurrent neural networks. *In Proc. Int. Conf. Comp. Vis.*, 2015

[7] J. Dai, K. He, and J. Sun. BoxSup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. *In Proc. Int. Conf. Comp. Vis.*, 2015

[8] M. Mostajabi, P. Yadollahpour, and G. Shakhnarovich. Feedforward semantic segmentation with zoom-out features. *In Proc. IEEE Conf. Comp. Vis. Pattern Recognition*, 2015

[9] C. A. Sutton and A. McCallum. Piecewise training for undirected models. *In Proc. Conf. Uncertainty Artificial Intelli*, 2005

[10] J. L. McClelland and D. E. Rumelhart. Explorations in Parallel Distributed Processing. *MIT Press, Cambridge, MA*, 1988

[11] G. Gudj onsson. Theory and Applications of Neural Circuits. *Independent Study, Electrical Engineering Department*, University of North Dakota, 1989
[12] J. Park, F. Diehl, M.J.F. Gales, M. Tomalin and P.C. Woodland. Training and Adapting MLP features for Arabic Speech Recognition. In Proc. of IEEE ICASSP, 2009

[13] Isabelle Guyon, and Andre Elisseeff, “An Introduction to Feature Extraction,” Studies in Fuzziness and Soft Computing, Springer-Verlag, 207: 1-25, 2006

[14] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In Proc. Adv. Neural Info. Process. Syst., 2014

[15] Ishant Shanu, Chetan Arora and Parag Singla. Min Norm Point Algorithm for Higher Order MRF-MAP Inference. IEEE International Conference on Computer Vision and Pattern Recognition, 5365-5374, 2016

[16] Anil Armagan, Martin Hirzer, Peter M. Roth and Vincent Lepetit. Learning to Align Semantic Segmentation and 2.5D Maps for Geolocalization. IEEE International Conference on Computer Vision and Pattern Recognition, 3425-3432, 2017

[17] Cheruku Sandesh Kumar, Ratnadeep Roy, Mayank Gupta, Achyut Sharma, Archek Praveen Kumar, Abhay Sharma. Velocity Estimation and Speed Tracking Through Segmentation. In International Journal of Control Theory an Applications (IJCTA), 9(23): 101-106, 2016

[18] Cheruku Sandesh Kumar, Ratnadeep Roy, Archek Praveen Kumar, Mayank Gupta. Segmentation on Moving Shadow Detection and Removal by Symlet Wavelet for Vehicle Detection. In IEEE International Conference on Computing for Sustainable Global Development (INDIACom), 2016

[19] Chenxi Liu, Liang-Chieh Chen, Florian Schroff, Hartwig Adam, Wei Hua, Alan Yuille, Li Fei-Fei. Auto-DeepLab: Hierarchical Neural Architecture Search for Semantic Image Segmentation. IEEE International Conference on Computer Vision and Pattern Recognition, 82-92, 2019.

[20] Bohan Zhuang, Chunhua Shen, Mingkui Tan, Lingqiao Liu, Ian Reid. Structured Binary Neural Networks for Accurate Image Classification and Semantic Segmentation. IEEE International Conference on Computer Vision and Pattern Recognition, 413-422, 2019

[21] Tong He, Chunhua Shen, Zhi Tian1 Dong Gong, Changming Sun, Youliang Yan. Knowledge Adaptation for Efficient Semantic Segmentation, IEEE International Conference on Computer Vision and Pattern Recognition, 578-587, 2019