

Image Classification through Dynamic Hyper Graph Learning

Ms. Govada Sahitya,

Dept of ECE,

*St. Ann's College of Engineering and
Technology, Chirala.*

J. Lakshmi Narayana,(Ph.D),

Associate Professor,

*Dept of ECE, St. Ann's College of Engineering and
Technology, Chirala.*

Abstract

Image classification has gained much attention in the recent years for variety of applications ranging from image processing, remote sensing, biomedical etc., To categorize an image into one of a set of classes many approaches have been proposed particularly much focus is given on graph based learning. Hyper graph learning has been investigated and found problems for generating hyper edges and how to handle set of hyper edges using statistical learning theory still it suffers with the problems of loss minimization. In this paper regularization of loss minimization is addressed for existing and new images also with dynamic hyper graph learning approach by giving promising performance over conventional methods.

I. Introduction

The objective of classification of an image is to automatically categorize all pixels in an image into land cover classes or information themes. This can be done in two ways. Using the Supervised methods, that will take the statics of the previous images, and unsupervised methods [8] that depend on the current image pixel intensity to derive some threshold value based on the intensities. There are many advantages using the hyper graph learning: First, Compared to simple graph learning, the hyper graph structure more reliable. Second, the simple graph sometime does not exist a simple similarity measure for pair wise data point.

A hyper graph is a special graph which contains hyper edges [1]. Normally, in the simple graphs each edge connects to set of vertices, but in hyper graph each edge connects arbitrary number of vertices. Let $V=\{v_1, v_2, \dots, v_n\}$ and $E=\{e_1, e_2, \dots, e_m\}$ defined on V . For any hyper edge $e \in E$ such that $\{v_1^e, v_2^e, v_6^e\}$ which is sub set of V . In a hyper graph vertices are connected by a hyper edge, each hyper edge is assigned a weight. In hyper graph

Learning, the weights of the hyper edges are set according to certain rules, for example, the weight of hyper edge [4] is calculated by average, median of the pair wise affinities within the hyper edge and constructs a hyper graph, and hyper edges are generated based on images and their nearest neighbors.

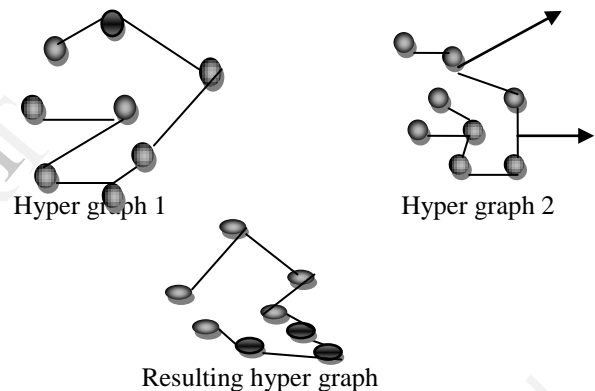


Fig 1: Hyper graph construction

In the dynamic hyper graph learning, first some features are extracted from image. Names or numbers are available for several images. Construct a hyper graph by using a set of hyper edges generated from samples and its neighbors. We construct some rules for that hyper graph. Based on that rules we classify the images. For example, we construct a hyper edge and generate rules for that hyper graph. For example, we take 2 or 3 images from that images, we calculate median, standard deviation (or) averages. We form rules for those images. Those pixels which having same rules that pixels are formed as a group. Finally, based on rules we classify the images.

Transductive learning methods are used for both labeled and unlabeled samples for improving the classification performance of conventional methods. Many methods have been proposed. For example, graph based and hyper graph learning.

Among this hyper graph learning has achieved promising performances. For example, Agarwal et al. applied hyper graph to clustering by using clique average to transform a hyper graph to a simple graph. Zass and Shashua adapted the hyper graph in image matching by using convex optimization. In, Tian et al. proposed a semi supervised learning method called Hyper prior to classify gene expression data by using biological knowledge. Wong and Lu proposed hyper graph based 3-D object recognition. Bu et al. Developed music recommendation by modeling relationship of different entities through a hyper graph to include music and users.

This paper includes the algorithm in simultaneous learning the labels of unlabeled samples and the weights of hyper edges, such that image classification performance can be significantly improved.

II. Existing methods

In the real world there are two classes of images: Textured images [5]. And Non-textured. The textured images are the ones that have some repeated patterns and they don't have any shape, and color such as fabric, rocks etc. Whereas the Non-Textured images have some shape, and color such as chairs man, mountains etc. We will define classifiers on the trained images, when a new image matches to any one of the classifier. We say that the image is having the defined texture or non texture.

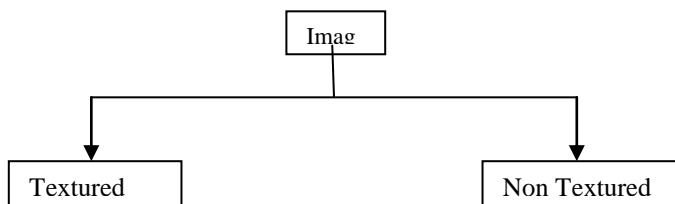


Fig 2: Image classes

Many classification techniques have been developed, such as decision tree, K-nearest neighbor [6], and support vector machines [7]. Since image labeling is labor intensive and time consuming, image classification frequently suffers from the training data insufficiency problem. These problems can be overcome by the Transductive learning methods. Among existing Transductive learning methods, graph-based learning achieves promising performance. In the graph based learning method [2]

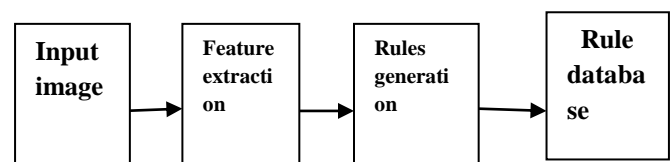
it suffers from two problems: First, The similarity estimation between samples and the Second, Ignore the relation in a high order. It is easily find two close samples according to the pair wise similarities, but it is not easy to predict whether there are three or more samples. These two problems can be resolved by hyper graph learning.

In hyper graph learning a set of vertices is connected by a hyper edge in a hyper graph. Each hyper edge is assigned a weight. Weights having certain rules. Weights will help improve classification. Hyper graph learning algorithms are star expansion, clique expansion, clique averaging, bola's expansion etc.,[3]. In the hyper graph learning technique two problems are to be addressed: First, How to generate the hyper edge and the Second, How to handle the large set of hyper edges. To avoid these two problems we use "Dynamic hyper graph learning".

In dynamic hyper graph learning we generate hyper edges by linking images and their nearest neighbors. By varying the size of the neighborhood we improve the hyper graph construction. In the proposed dynamic hyper graph learning a principled approach define to regularize the loss minimization based on the statistical learning theory. In this paper, the regularizing loss minimization is addressed with dynamic learning along with statistical learning theory.

III. Proposed method

In Dynamic hyper graph learning, the input image is converted into gray. We formulated different rules based the standard deviation of the entire pixels of the image. The pixels that pass the rule are going to be classified into one group, like wise different groups are formed with different hyper edges. The pixels connected through the hyper edge called hyper graph. The pixels that are common and part of all the hyper graphs [9] are going to be listed, for the generation of the new hyper graph. The resultant hyper graph is the output that classifies the image.



The following figure (3.a) shows the block diagram for rules generation.

In the above Fig, from the input image features are extracted. The extracted are then fed to the rules generation process, which then generates the rules. These newly generated rules are stored in the rule base for the further processing of the image.

The following figure shows the block diagram for dynamic hyper graph learning.

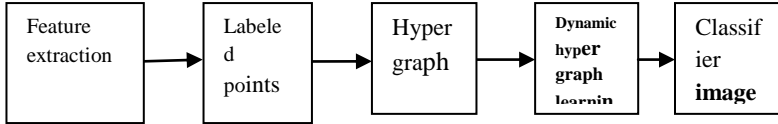


Fig (3.b): Dynamic hyper graph learning

In the above fig, the features are extracted from the input image. The labeled points are assigned to that features and construct a hyper graph for that features. And finally, the classifier image is obtained.

Table 1: Rule Base

Rules	Condition	Value
Rule1	Classes>maxclassifier	Not Matched
Rule2	Classes<minclassifier	Not Matched
Rule3	Classes>=avgstd &&classes<=maxclassifier <=maxclassifier	Matched

The Rule base is prepared and is used to classify the input image. The rules are constructed based on the analysis of the features extracted from different images of the same class.

IV. Algorithm

1. Convert rgb to gray.
2. Calculate standard deviation for images.

Calculate maximum, minimum and threshold by using standard deviation.

$$\begin{aligned}
 max &= \max(f(:)). \\
 min &= \min(f(:)). \\
 mean &= \frac{1}{M * N} \sum_{i=1}^m \sum_{j=1}^n f(i, j).
 \end{aligned}$$

Construct adjacent matrix.

3. Finding adjacent matrix to find hyper graph.
4. Rules generation as shown in the table 1.
5. Finally classify the images.

V. Results and Discussion

The bellow result is showing, four different images in the forest about a tiger. The features are extracted from the four images after converting them into gray scale images. The features are stored, for analysis. The image to be classified is also taken and processed to get features. Once the features from the image to be classified are collected, they will be compared with the all existing features, marking with some threshold. Some predefined rules are applied to classify it. Thus if the input image matches with any one of the image classifier, we say a "Match" occurred, otherwise "Not Matched".

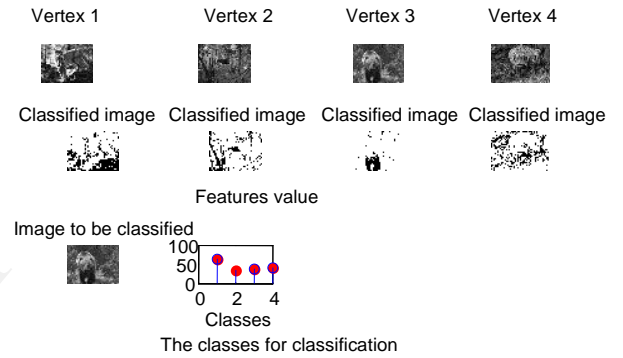


Fig 4: Classification result for Forest

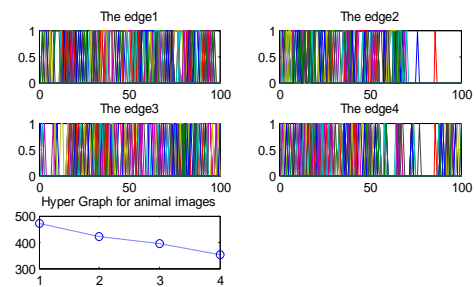


Fig 5: The edge details of existing Method

The feature vector for the existng method [472,420,393,353]

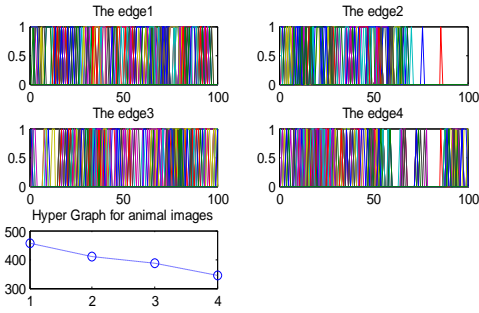


Fig 6: The edge details for proposed method

The feature vector for the proposed method is [457,408,385,343]. Here we can understand that the misclassification can be reduced in the proposed method and hence the performance is increased.

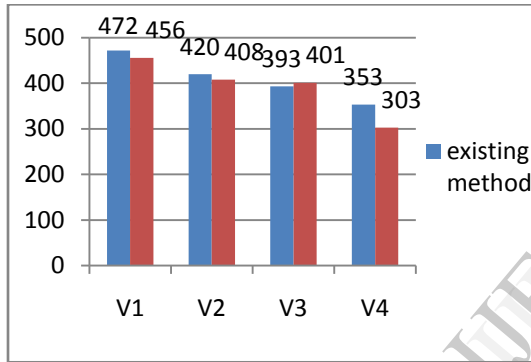


Fig 7: The edge weight vs. vertices

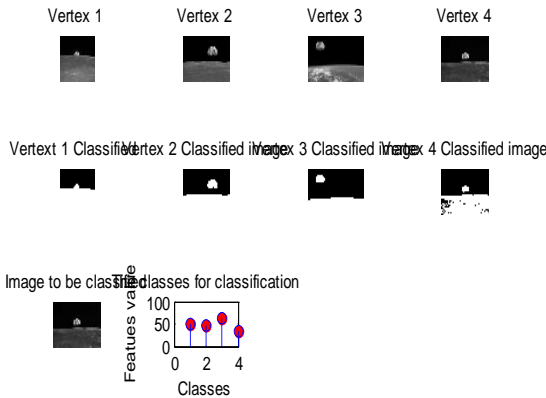


Fig 8: Classification result for moon

Here four different images about a moon are collected in different times. The features are extracted from the four images after converting

them into gray scale images. The features are stored, for analysis. The image to be classified is also taken and processed to get features. Once the features from the image to be classified are collected, they will be compared with the all ready existing features, marking with some threshold. Some predefined rules are applied to classify it. Thus if the input image matches with any one of the image classifier, we say a "Match" occurred, otherwise "Not Matched".

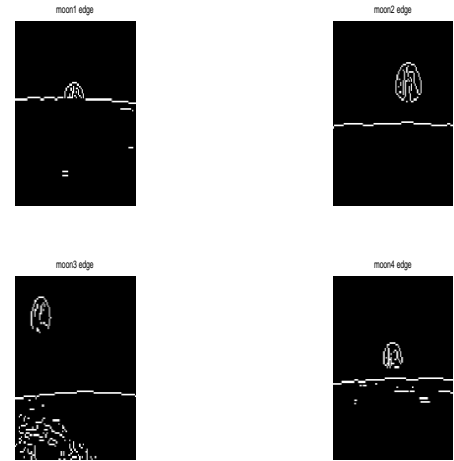


Fig 9: Moon Edges

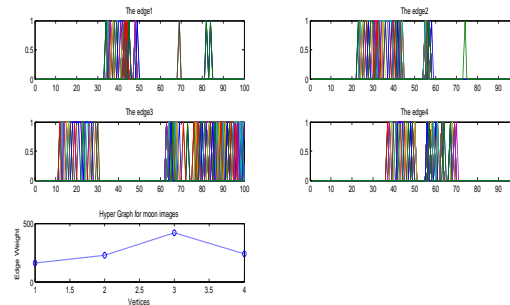


Fig 10: The edge details for existing method

The feature vector for the existing method [157,228,416,238],

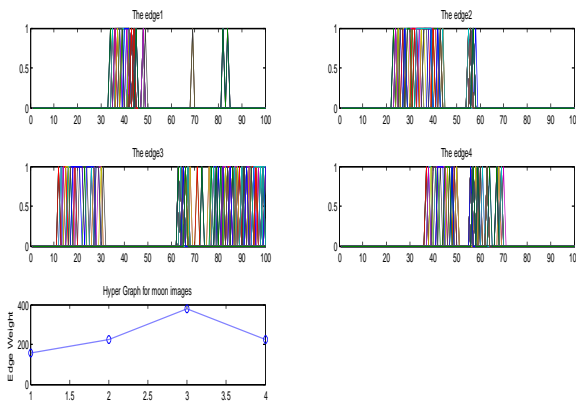


Fig 11: The edge details for proposed method

The feature vector for the proposed method is $fv = [155, 224, 380, 224]$. Here we can understand that the misclassification can be reduced in the proposed method and hence the performance is increased.

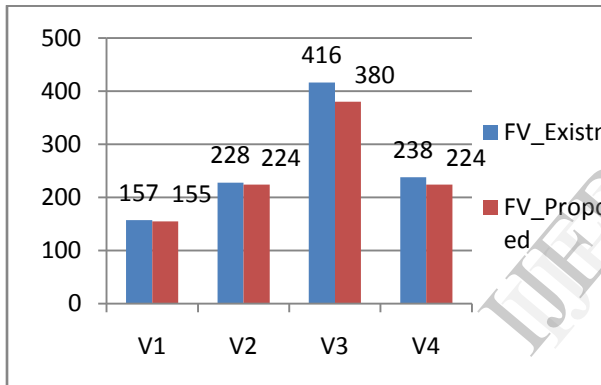


Fig 12: The edge weight Vs Vertices

VI. Conclusion

The objective of this project is to define classifiers and also verifying new images for analysis by using Dynamic hyper graph learning technique. This technique gives promising performance over conventional methods. The proposed method along with investigating a robust hyper edge construction, but also presents a simultaneous learning of the labels of unlabeled images and the weights of hyper edges.

VII. References

- [1] Ze Tian, TaeHyun Hwang and Rui Kuang, "a hypergraph-based learning algorithm for classifying arraycgh data with spatial prior," Genomic Gensips2009, IEEE International workshop May 2009, pp.1-4.
- [2] Jun Yu, Dacheng Tao, Senior Member, IEEE, and Meng Wang, Member, "adaptive hypergraph learning and its application in image classification," IEEE transaction on img proc, vol. 21, No.7, july 2012, pp.3262-3272.
- [3] Yuchi huang, "Hyper graph based visual categorization and segmentation," October 2010.
- [4] Li Pu and Boi Faltings, "Hypergraph Learning with Hyperedge Expansion," Artificial Intelligence Laboratory Ecole Polytechnique Federale de Lausanne.
- [5] Robert M. Haralick, K. Shanmugam, "Textural features for image classification," IEEE transactions on systems, man and cybernetics, vol SMC-3, NO-6, november 1973, pp.610-621.
- [6] J.A. Hartigan and M.A. Wong, "Algorithm AS136, a k-means clustering algorithm," applied statistics, vol.28, pp.100-108, 1978.
- [7] Christopher J. C Burges, "a tutorial on support vector machines for pattern recognition," kulwer academic publishers, Bosten manufactured in Netherlands, pp.1-43.
- [8] Yuchi Huang, "Hyper graph based visual categorization and segmentation," new Brunswick, new jersey, October 2010.
- [9] Y. Huang, Q. Liu, S. Zhang, and D. Metaxas, "Image retrieval via probabilistic hypergraph ranking," in *Proc. Int. Conf. comput. Vis. Pattern Recog.*, San Francisco, CA, 2010, pp. 3376-3383.