JCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.-6, Issue-12, Dec 2018 E-ISSN: 2347-2693

Behavior of SVM based classification for varying sizes of heap-grain images

Vishwanath S. Kamatar[1], Rajesh Yakkundimath[2*], Girish Saunshi[3]

[1,2]Department of Computer Science & Engineering, K.L.E. Institute of Technology, Hubli-580030, India
[3] Department of Information Science & Engineering, K.L.E. Institute of Technology, Hubli-580030, India

[*]*Corresponding Author: rajeshymath@gmail.com*

Available online at: www.ijcseonline.org

Accepted: 20/Dec/2018, Published: 31/Dec/2018

Abstract— This paper describes the behavior of support vector machine based classification for varying sizes of heap-grain samples. Different grains like cow peas, green gram, ground nut, green peas, jowar, red gram, soya and toor dal are considered for the study. The color and texture features are used as input to the SVM classifier. The recognition accuracy is observed for specific size training and mixed size training methods. The recognition accuracy is found to be 100% for the test samples with which the classifier is trained and decreased when training and testing samples are different. The work finds application in automatic recognition and classification of food grains by the service robots in the real world.

Keywords— Classification, feature extraction, grain samples, support vector machine

I. Introduction

The object recognition in computer vision is a task of finding the given object in an image or video sequence. Humans recognize an image with little effort, despite the fact that the objects vary somewhat with different viewpoints, different sizes, scales or even when translated or rotated. Objects are even recognized by the humans, when they are partially obstructed from view. But the same task is a challenge for computer vision systems. For any object in an image, there are many 'features' that need to be extracted for the description of an object. The present work involves processing of images of different types of grains, extracting the features of the grains and finally developing a suitable SVM model to recognize the different types of heap-grain images. Images of heaps of different grains are obtained using camera. The color and texture features are extracted. These features are used to train the SVM classifier and new images (not trained) are given as input for the classifier to find the accuracy of recognition. Many computer vision applications exist today. In order to know the state-of-the-art in this area we have carried out the literature survey and following is the gist.

II. Related Work

(Yuyong Cui, et al., 2008) have proposed a method to estimate abundances from hyper spectral image using probability outputs of support vector machines (SVM) and training a SVM with a gauss kernel function. The authors have discussed the relationship between kernel functions and nonlinear mappings and mapped spaces. A new compound kernel function is proposed. They have compared the compound kernel with other kernels in hyper spectral image classification. The results have shown that the proposed method is more accurate than the other methods. (Qing Song, et al., 2002) have proposed a robust support vector machine for pattern classification. The work aims at solving the over-fitting problem when outliers exist in the training data set. The incorporation of the average techniques to the standard support vector machine (SVM) training made the decision function less detoured by outliers, and also controlled the amount of regularization automatically. Experiments for the bullet hole classification problem have shown that the number of the support vectors is reduced, and the generalization performance is improved significantly compared to that of the standard SVM training. (Jing Li, et al., 2006) have proposed relevance feedback (RF) schemes based on support vector machines (SVMs) widely used in content-based image retrieval (CBIR). The performance of SVM-based RF approaches is often poor when the numbers of labeled feedback samples are small. The authors have developed a new machine learning technique, multi-training SVM (MTSVM), which has the merits of the co-training technique and a random sampling method in the feature space. Based on the proposed MTSVM algorithm, the above two problems can be mitigated. Experiments are carried out on a large image set of some 20000 images, and the preliminary results demonstrated that the developed method consistently improved the performance over conventional SVM-based RFs in terms of precision and standard deviation, which were used to evaluate the effectiveness and robustness of a RF algorithm, respectively. (Evgeniy Gabrilovich and Shaul Markovitch, 2004) have proposed the text categorization algorithms to represent documents as bags of words. The previous studies have found that the large numbers of features are relevant for text categorization with support vector machines peaked when no feature selection is performed. Authors have described a class of text categorization problems that are characterized with many redundant features. Even though most of these features are relevant, one is able to concisely capture only few features to obtain the desired categorization. (Subhransu Maji, et al., 2008) have proposed a classifier using kernelized SVM that requires evaluating the kernel for a test vector and each of the support vectors. One can build histogram intersection kernel SVMs (IKSVMs) with runtime complexity of the classifier logarithmic in the number of support vectors as opposed to linear for the standard approach. Further, authors have shown that by pre-computing auxiliary tables, we could construct an approximated classifier with constant runtime and space requirements, independent of the number of support vectors, with negligible loss in classification accuracy on various tasks. This approximation also applies to $1 - \chi 2$ and other kernels of similar form. (Bhaskar Mehta, et al., 2008) have discussed the characteristics of image spam and proposed two solutions for detecting image-based spam and compared with the existing techniques. The one solution, which uses the visual features for classification, offers an accuracy of about 98%. SVMs (Support Vector Machines) are used to train classifier using judiciously decided color, texture and shape features. (Amit David and Boaz Lerner, 2005) have implemented structural risk minimization and cross-validation in order to optimize kernel and parameters of a support vector machine (SVM) and multiclass SVM-based image classifiers, thereby enabling the diagnosis of genetic abnormalities. Authors have suggested an SVM for the classification of images required for genetic syndrome diagnosis. Using the principle of SRM and cross validation procedure authors have selected a model for the SVM evaluating linear, polynomial and Gaussian kernels. The SVM has extended to multi-class problems using the ECOC algorithm, accurately classified FISH signals as real or artifacts of two genetic abnormalities. Accurate performance of the SVM in comparison to other state-of-the-art classifiers demonstrates the benefit of SVM-based genetic syndrome diagnosis. (Reda A. El-Khoribi, 2008) has introduced a novel approach to supervised classification of multispectral images. The approach uses a new discriminative training algorithm for discrete hidden Markov tree (HMT) generative models applied to the multi-resolution ranklet transforms. This present study evaluates the performance of the new training method and comparing its performance with the baseline HMT classifiers. The algorithm developed uses the sufficient statistics of the HMT generative model to form a fixed length training vector to be used in linear discriminant classifiers (like SVM). The algorithm proves considerable amount of improvement over the baseline HMT when applied to land cover images. (Yasemin Altun, et al., 2003) have presented a novel discriminative learning technique for label sequences based on a combination of Support Vector Machines and Hidden Markov Models which authors call Hidden Markov Support Vector Machine. The proposed architecture handles dependencies between neighboring labels using Viterbi decoding. In contrast to standard HMM training, the learning procedure is discriminative and is based on a maximum/soft margin criterion. Compared to previous methods like Conditional Random Fields, Maximum Entropy Markov Models and label sequence boosting, HM-SVMs have a number of advantages. Most notably, it is possible to learn non-linear discriminant functions via kernel functions.

From the literature survey, it is observed that SVM is not applied to the task of recognition and classification of fruits and food grains and hence the motivation for the present work. The work carried emphasizes on analysis of Support Vector Machine based method for automatic recognition and classification of heap-grain samples. Initially images of grains are acquired using a digital camera. From these images, the color and texture features are extracted. These features are input to the SVM classifier for training. The untrained grain images are given as input to the classifier for testing. The recognition accuracy is found for specific size training and mixed size training method.

The paper is organized into five sections. Section three contains information about proposed methodology, image acquisition, feature extraction and support vector machines. The results and discussions are given in section four. Section five deals with conclusion and future scope of the work.

III. METHODOLOGY

The devised methodology consists of four phases namely image acquisition, feature extraction, development of SVM classifier and reporting of results. The block diagram of the proposed methodology is shown in Fig. 1

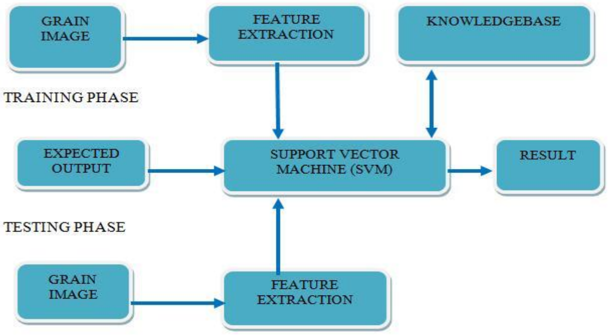


Figure 1. Block diagram of proposed methodology

3.1 Image acquisition

The sample images are obtained by taking around 2 kg of grain kernels into a large plastic bag and shaking it to mix the grain thoroughly. The grains are slowly poured onto a black card sheet until it takes the shape of a heap (cone). The images are taken from the top keeping the distance of 12 inches between the lens of camera and the top of heap. This process is repeated for each of the grain samples. A total of 800 images of heap samples are acquired (100 images of each grain type). We have used the color camera FINEPIX F450, 5.2 Mega pixels and the images acquired are of size 2272 X 1704 pixels. An image acquisition set up is as shown in Fig. 2. The sample images of heap-grain samples are shown in Fig. 3. The grains considered are cow peas, green gram, groundnut, green peas, jowar, red gram, soya and toordal for the study. Table 1 gives the list of common names and scientific names of grains taken for the study.

Table 1. List of grains and scientific names

Sl No.	Grains	Scientific name
1	cow peas	Vigna unguiculata
2	green gram	Vigna radiata
3	groundnut	Arachis hypogaea
4	green peas	Pisum sativum
5	jowar	Sorghum bicolor
6	red gram	Cajanus cajan
7	soya	Glycine max
8	toordal	Cajanus indicus

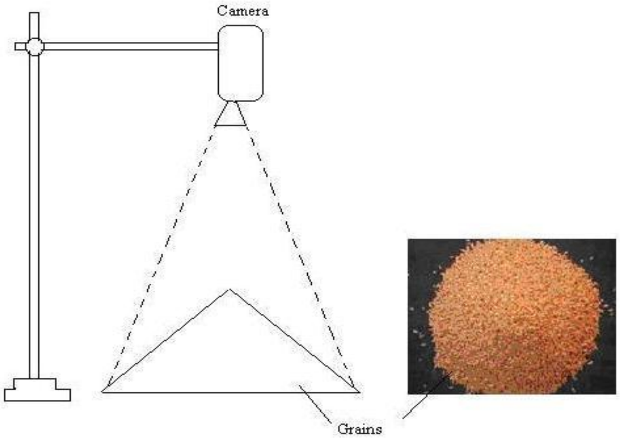
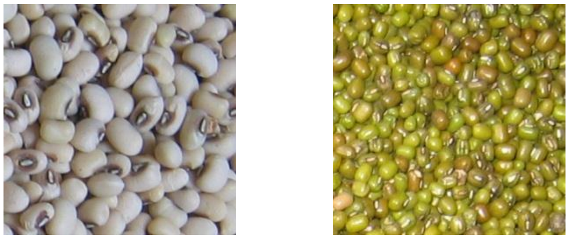


Figure 2. Image acquisition setup


a) cow peas b) green gram

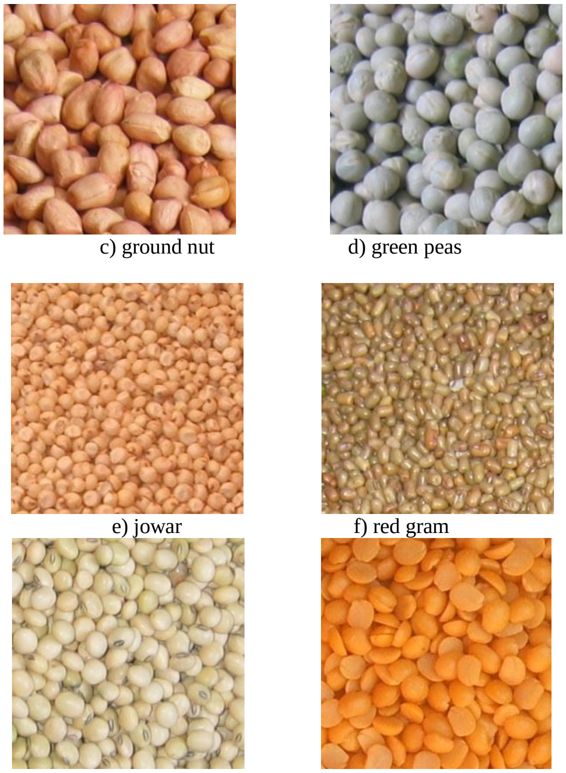
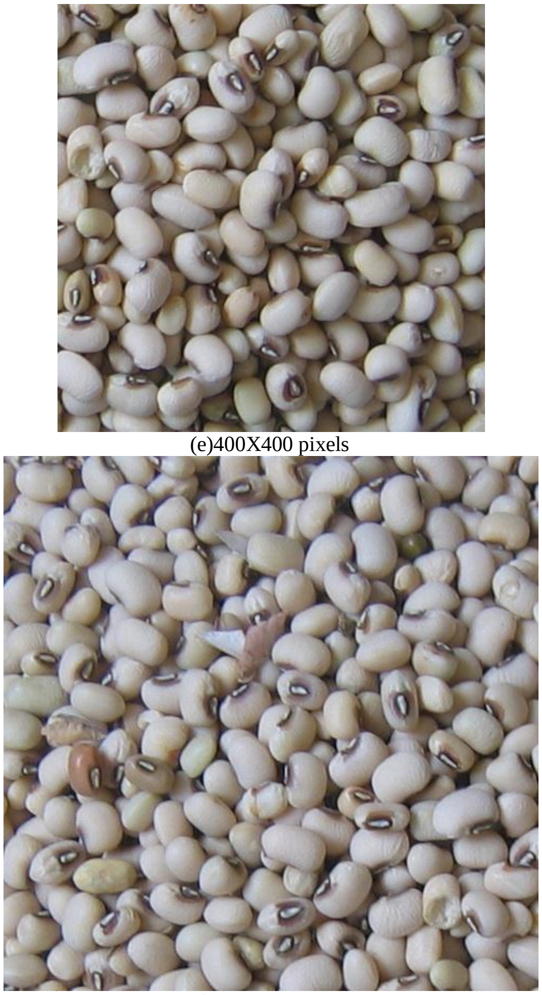

c) ground nut d) green peas
e) jowar f) red gram
g) soya h) toor dal

Figure 3. Different types grains used for training and testing

The preprocessing techniques used in the work are deblurring, smoothing, noise elimination, edge sharpening, thinning and cropping. Cropping of the images is performed to get the images of different sizes, namely, 64X64, 128X128, 256X256, 300X300, 400X400 and 500X500 pixels. We have used the software Photoshop 7.0. The images of different sizes are given in Fig. 4.

(a)64X64 pixels (b)128X128 pixels (c)256X256 pixels

(d) 300X300 pixels


(e)400X400 pixels

(f)500X500 pixels

Figure 4. Different size heap-grain samples of cowpeas used for training and testing

3.2 Feature extraction

Food grains are normally classified using color features. In certain food grains types, namely, cow peas, green gram, groundnut, green peas, jowar, red gram, soya and toordal there is overlap in color hence we have used both color and texture features.

3.2.1 Color feature extraction

From the original images, RGB components are separated and the Hue (H), Saturation(S) and Value (V) components are extracted. The equations (1),(2) and (3) are used to obtain Hue, Saturation and Value parameters of the image samples. The mean, variance and range for all these 6 components are calculated and a total of 18 color features are extracted and stored suitably for later usage in training SVM. The steps involved in color feature extraction are given in Algorithm 1. Table.2 gives the color features of cow peas.

$$H = \begin{cases} \theta & if\ B \leq G \\ 360 - \theta & if\ B > G \end{cases} \qquad (1)$$

$$\theta = cos^{-1}\left\{\frac{\frac{1}{2}[(R-G)+(R-B)]}{[(R-G)^2+(R-B)(G-B)]^{1/2}}\right\}$$

$$S = 1 - \frac{3}{(R+G+B)}[min(R,G,B)] \qquad (2)$$

V=maximum value of either R,G or B
(3)

Algorithm 1: Color feature extraction
Input: Original 24-bit color image.
Output: 18 color features.
Start
Step 1: Separate the RGB components from the original 24-bit input color image.
Step 2: Obtain the HSV components from RGB components using the equations (1), (2) and (3).
Step 3: Find the mean, variance, and range for each RGB and HSV components.
Stop.

Table 2. Color features for image sample (cowpeas) of size 256x256 pixels

Sl No.	Parameters	Mean	Varience	Range
1	Red	+0.6416	+0.0230	+0.8235
2	Green	+0.6095	+0.0305	+0.8314
3	Blue	+0.5944	+0.0397	+0.9804
4	Hue	+0.3534	+0.1152	+0.9957
5	Saturation	+0.1311	+0.0153	+1.0000
6	Value	+0.6513	+0.0247	+0.8745

3.2.2 Textural feature extraction

The heap-grain samples exhibit different textures and provide information about the variation in the intensity of a surface by quantifying properties such as smoothness, coarseness, and regularity. The most widely accepted models are co-occurrence and run-length matrices and we have used the co-occurrence matrix method for texture feature extraction. A total of 30 textural features are extracted. The equations (4) to (12) are used to evaluate the textural features. Algorithm 2 is used to compute the co-occurrence matrix. Algorithm 3 is used for textural feature extraction. Table.3 gives the textural features of cow peas.

Mean (μ) $= \sum_{x} x \sum_{y} P(x,y)$

(4)

Variance

$$= \sum_{x,y} (x-\mu)^2 P(x,y) \quad (5)$$

Range

$$= Max(p(x,y)) - min(p(x,y)) \quad (6)$$

Energy= $\sum_{i=1}^{Ng} \sum_{j=1} p^2 d(i,j)$ (7)

Entropy= $-\sum_{I,J} P(i,j) \log P(i,j)$ (8)

Homogeneity= $\sum_{i=1}^{Ng} \sum_{j=1}^{Ng} \frac{p_d(i,j)}{1+|i-j|}$ (9)

Maximum Probability $= max(P(x,y)$) (10)

Contrast= $\sum_{n=0}^{Ng-1} n^2 \sum_{|i-j||} Pd(i,j)$ (11)

Inverse Difference Moment = $\sum_{x,y;x\neq y} \frac{P^{\lambda}(x,y)}{|x-y|^k}$ (12)

where μ_x, μ_y σ_x, σ_y are means and standard deviations defined by,

$$\mu_x = \sum_x x \sum_y P(x,y), \quad \mu_y = \sum_y y \sum_x P(x,y),$$

$$\sigma_x = \sum_x (x-\mu_x)^2 \sum_y P(x,y)$$

and $$\sigma_y = \sum_y (y-\mu_x)^2 \sum_x P(x,y)$$

Algorithm 2 : Calculation of Co-occurrence matrix $P_{f,d}$(x, y) from the image f(x, y).
Input : Input gray level image f(x, y) (matrix of size M*N)
Output : Co-occurrence matrix $P_{f,d}$(x, y) for d=1 in the direction f.
Start
Step 1 : Assign $P_{f,d}$(x, y)=0 for all x, y Є [0, L], where L is the maximum gray level.
Step 2 : For all pixels(x1, y1) in the image, determine (x2, y2), which is at distance d in
direction f and perform
$P_{f,d}$[f(x1, y1), f(x2, y2)]= $P_{f,d}$[f(x1, y1), f(x2, y2)] + 1
Stop

Algorithm 3: Textural feature extraction
Input: RGB components of original image
Output: 30 Textural features
Start
Step 1: For all the separated RGB components derive the Gray Level Co-occurrence Matrices
(GLCM) *Pφ,d (x, y)* for four different values of direction *φ* (*0^0, 45^0,90^0 and 135^0*) and
d=1 which are dependent on direction *φ.*

Step 2: Compute the co-occurrence matrix, which is independent of direction using the
Algorithm 2.

Step 3 GLCM features namely, mean, variance, range, energy, entropy, homogeneity, sum mean,
maxprob, contrast and inverse difference moment, are calculated using equations (4)
to (12).
Stop

Table 3. Textural features for image sample(cowpeas) of size 256x256 pixels

Sl.no	Features	Red GLCM	Green GLCM	Blue GLCM
1	Mean	+0.9961	+0.9961	+0.9961
2	Variance	+32.5904	+25.5575	+16.2260
3	Range	+0.0050	+0.0046	+0.0027
4	Energy	+0.0024	+0.0019	+0.0013
5	Entropy	+16.4688	+16.9168	+17.5240
6	Homoginity	+0.6331	+0.6054	+0.5559
7	SumMean	+356.0322	+338.1971	+329.8455
8	Maxprob	+0.0050	+0.0046	+0.0027
9	Contrast	+15.1332	+16.1472	+17.3426
10	Idm	+0.7131	+0.6870	+0.6375

3. 3 Support Vector Machine Based Recognition

Support vector machines comprise of a set of related supervised learning methods used for classification and regression. Viewing input data as two sets of vectors in an *n*-dimensional space, SVM constructs a separating hyperplane in the space, one which maximizing the margin between the given two data sets. To calculate the margin, two parallel hyperplanes are constructed, one on each side of the separating hyperplane, which are "pushed up against" the two data sets. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the neighboring data points of both classes, since in general the larger the margin the better the generalization error of the classifier. Classifying data is a common need in machine learning. Suppose some given data points belong to one of two classes and the goal is to decide, which class a new data point will be in. In the case of support vector machines, a data point is viewed as a *p*-dimensional vector (a list of *p* numbers), and we want to know whether we can separate such points with a (*p*−1) dimensional hyperplane. This is called a linear classifier. The classifier is also known as a maximum margin classifier. The Fig 5 shows the principle of support vector machines.

In this work, we have used the OSU-SVM toolbox available in MATLAB 7.0. The core of this toolbox is based on Dr. Lin's Lib SVM version 2.33. It is developed by Junshui Ma, Los Alamos National Lab and Yi Zhao, EE department, Ohio State University.

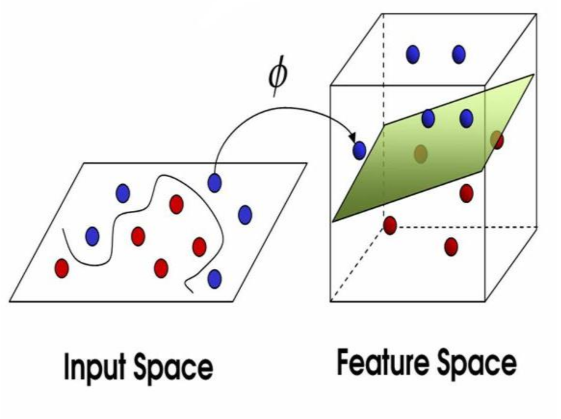


Figure 5. Principle of support vector machines

3.3.1 Training and testing of SVM

The features color, texture and combined color and texture are used to train the SVM classifier. We have adopted two methods for the training, namely, training with images of specific size and images of different sizes. Totally, 800 heap-grain image samples are used for the specific size training. For mixed size training, 2400 heap-grain image samples of different sizes are used. We have validated the process of classification by taking 15% of the trained images. The results of experimentation are given in section 4.0.

IV. Results and Discussion

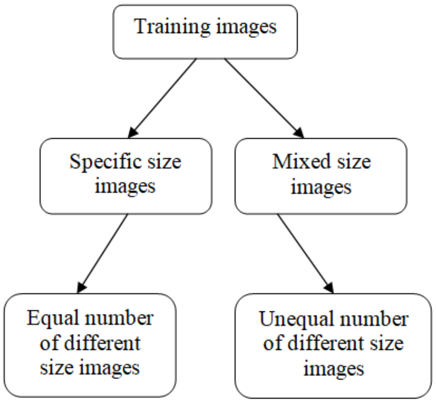

Training images
Specific size images
Mixed size images
Equal number of different size images
Unequal number of different size images

Figure 6. Classification of training methods

We have given a comparative study of the performance of the method with three feature sets and two types of training methods used. The classification of training methods used in our experimentation is as shown in Fig 6. The percentage of recognition and classification accuracy is defined as the ratio of correctly recognized image samples to the total number of test image samples. The percentage accuracy is calculated as given by the equation (13).

$$\text{Percentage accuracy} = \frac{\text{[illegible]}}{\text{[illegible]}} \quad (13)$$

4.1 Accuracy of classification using specific size training

In this experimentation, we have trained the SVM classifier with images having specific size such as 64x64, 128x128, 256x256, 300x300, 400x400 and 500x500 pixels. We have used 800 images by choosing 100 images of each grain type. We have tested the accuracy of recognition with SVM classifier using varying sized images.

The graph shown in Fig. 7 gives the recognition accuracy when the classifier is trained with images of size 64x64 pixels and subjected to testing with images of all sizes. It is observed that recognition accuracy is 100%, 87% and 94% for the images of sizes 64x64 pixels using color, texture and combined color and texture features respectively. It is further observed that the accuracy reduces, when the sizes of the images increase more than 64x64 pixels for which the machine is trained. Since color of food grain is a dominant feature in recognition, it is observed the maximum recognition accuracy is found with color features irrespective of varying sizes of images.

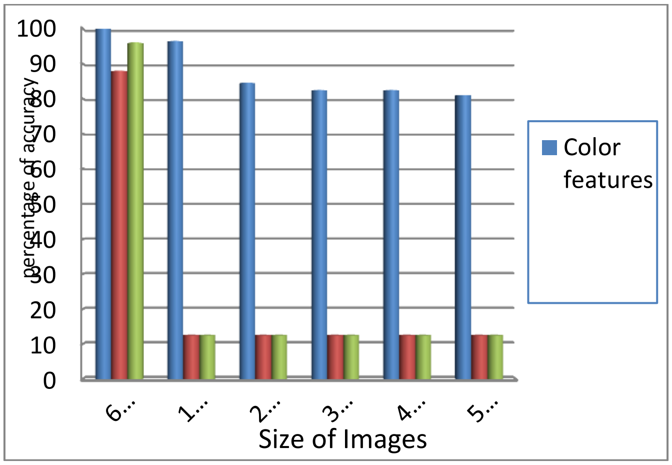


Figure 7. Results of training with images of 64x64 pixels

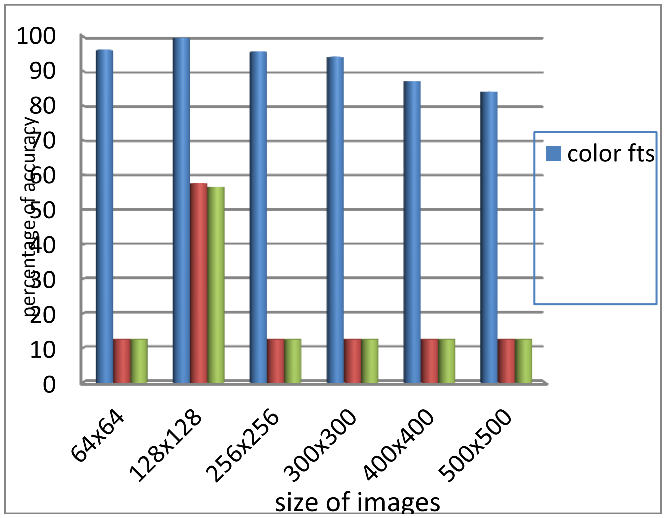


Figure 8. Results of training with images of 128x128 pixels

The graph shown in Fig.8 gives the recognition accuracy when the classifier is trained with images of size 128x128 pixels and tested with images of all the sizes. From the graph shown in Fig 8 it is observed that 100%, 57% and 55% recognition accuracy for the images of sizes 128x128 pixels using color, texture and combined color and texture features respectively. It is observed that the accuracy is 100% for images of size 128x128 pixels. But for other sizes the accuracy is reduced.

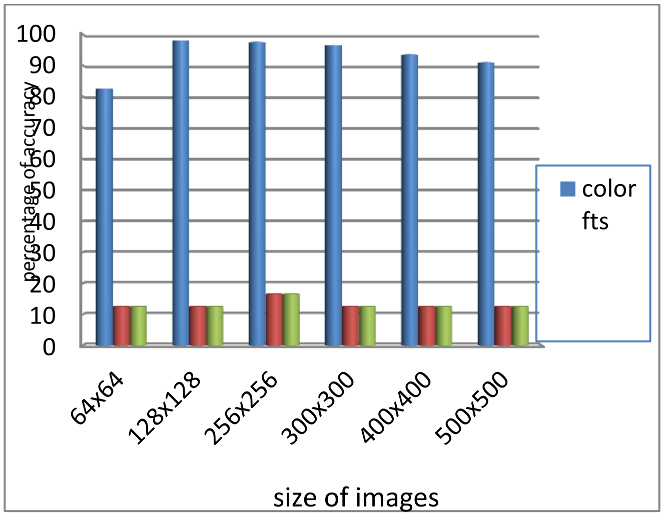


Figure 9. Results of training with images of 256x256 pixels

The graph in Fig. 9 gives the recognition accuracy when the classifier is trained with images of size 256x256 pixels. It is observed that using the color features, the accuracy is 99% and 98% for 128x128 and 256x256 pixels size images. But for other sizes, the accuracy is reduced.

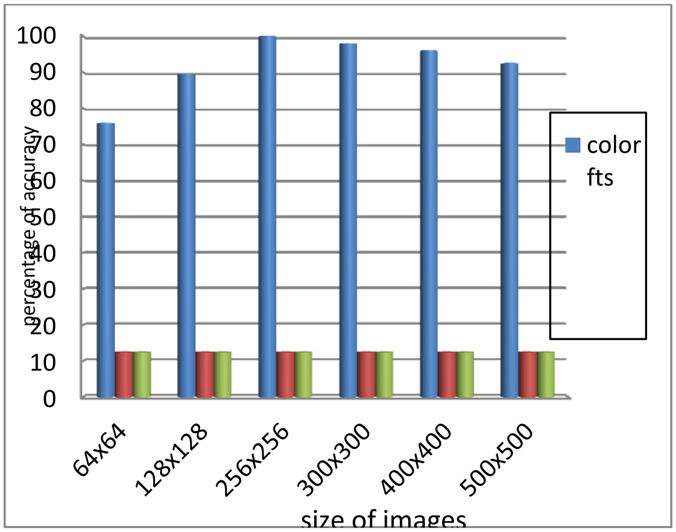


Figure 10. Results of training with images of 300x300 pixels

The graph in Fig. 10 shows the recognition accuracy when the classifier is trained with images of size 300x300 pixels. Using the color features, the recognition accuracy is 100% and 99% for 256x256 and 300x300 images respectively. But for other sizes, the accuracy is reduced.

.

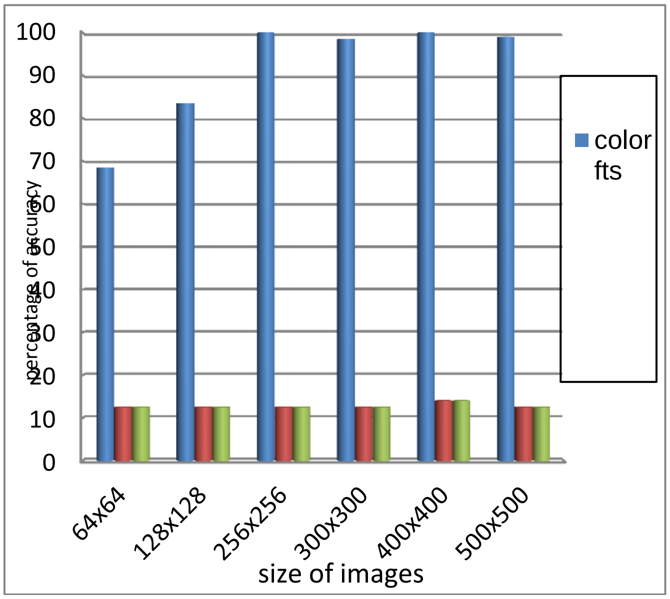


Figure 11. Results of training with images of 400x400 pixels

The graph in Fig. 11 shows the recognition accuracy when the classifier is trained with images of size 400x400 pixels. Using the color features, the recognition accuracy is reduced only for 64x64 and 128x128 pixels size images. From images with size 256x256 pixels and onwards the recognition accuracy is constant.

.

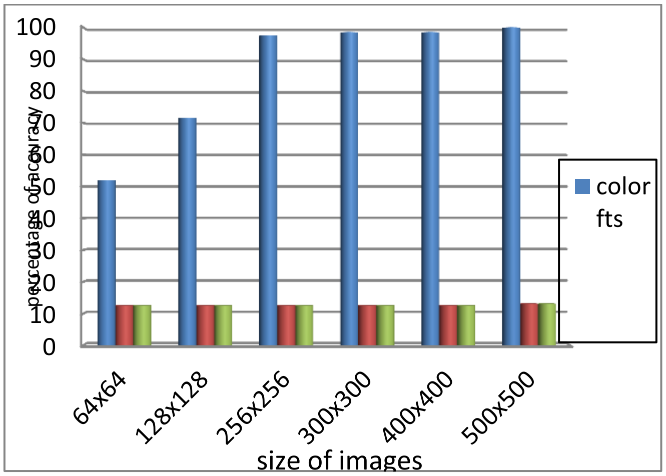


Figure 12. Results of training with images of 500x500 pixels

The graph in Fig. 12 shows the recognition accuracy when the classifier is trained with images of size 400x400 pixels. Using the color features, the recognition accuracy is reduced only for images of sizes 64x64 and 128x128 pixels. From images of sizes 256x256 pixels and onwards the recognition accuracy is almost constant. Using texture features, combined color and texture features recognition accuracy is only 12% and is constant for all other size images.

From the graphs shown in Fig 7 through Fig 12 it is evident that color features are suitable for recognition and classification of heap-grain images using support vector machine classifier. The image size with 128x128 onwards the average accuracy is found to be good.

4.2 Accuracy of classification using mixed size training

We have trained the SVM classifier by mixing all different size images to know the recognition accuracy. We have used totally 2400 images having 300 images of each grain type. The graph shown in Fig. 13 gives the percentage of recognition when we have trained the classifier with all different size images and tested with different size images. The graph shows that for color, texture and combined color and texture features, the percentage accuracy of recognition is found to be 100% for all sizes given input the color features.

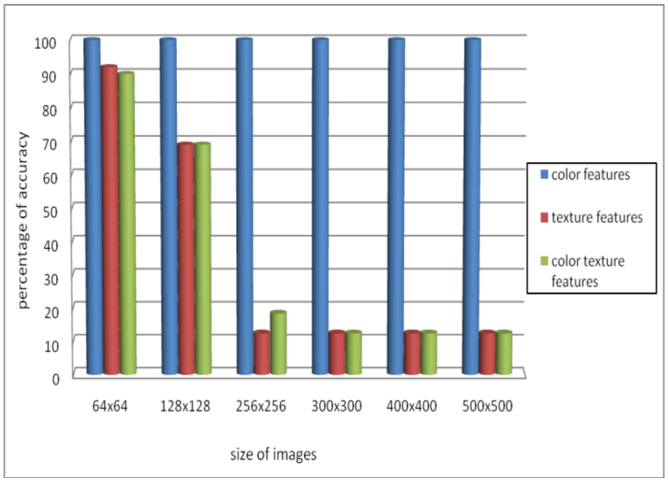


Figure 13. Training with all size images

It is observed that in mixed training, only the color features are suitable for recognition and classification of heap-grain images.

We have carried out experimentation by using only color features. Here, we have trained the classifier with 2400 images and in the first phase 50% images with particular size and rest 50% with all other sizes of images. In the second phase, 75% images with particular size and rest 25% with all other sizes of images. We have input the different sizes of images for testing. The graph shown in Fig. 14 shows the comparison with mixing of equal number of all sized images for training, 50% of images with 64x64 size and rest 50% all size images for training and 75% of images with 64x64 pixels size and rest 25% all size images for training. It is revealed from the graph that for equal number of all sized images, the recognition accuracy is 100%. It is also observed from the graph that for 50% of 64x64 pixels size and rest 50% all size images for training, it has given 98% and 97% accuracy for 64x64 and 128x128 pixels respectively. For all other size images, it has given 100% accuracy. It is also evident from the graph that for 75% of 64x64 images and 25% remaining images, it has given accuracy of 100% for 64x64 and 128x128 images and accuracy is decreased as the sizes of test images are increased.

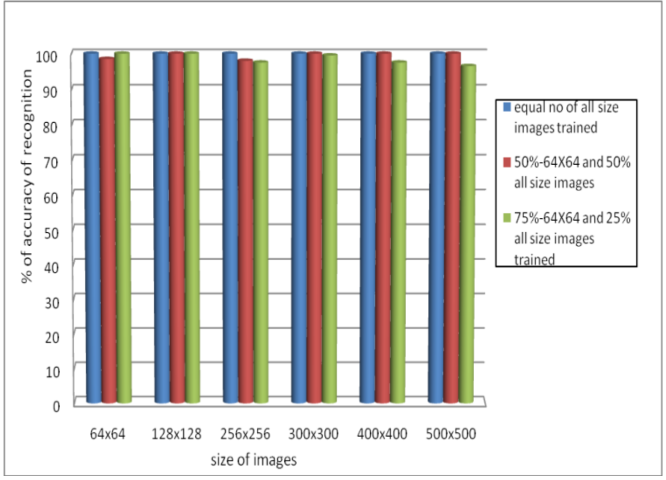


Figure 14. Recognition accuracy for training with different percentage of 64x64 size images

We have carried out same experimentation for 128x128, 256x256, 300x300, 400x400 and 500x500 pixels size images. The graphs shown in Fig. 15 through Fig. 19 gives the recognition accuracies for 128x128, 256x256, 300x300, 400x400 and 500x500 pixels size images respectively.

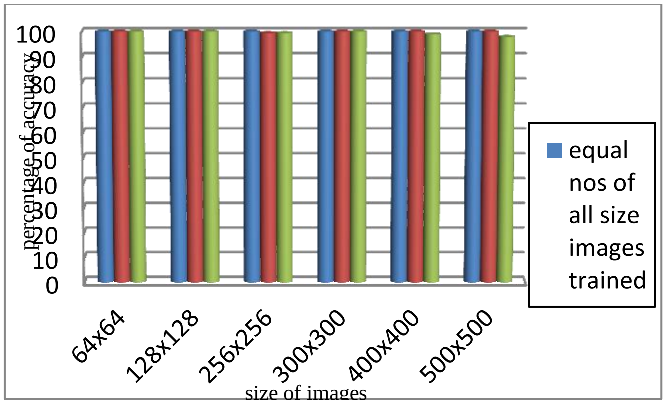


Figure 15. Recognition accuracy for training with different percentage of 128x128 size images

From the graph shown in Fig. 15 it is revealed that for equal number of all size images, it has given 100% accuracy. For 50% of 128x128 size and the rest 50% of all size images used for training has given 100% accuracy. For 75% of 128x128 and 25% the rest of all size images, it has given accuracy of 100% for 64x64 and 128x128 images and accuracy is decreased as the sizes of test images are increased.

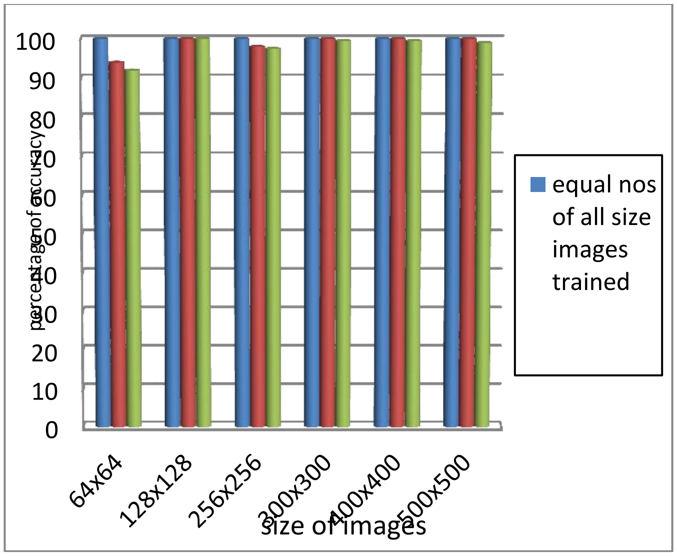


Figure 16. Recognition accuracy for training with different percentage of 256x256 size images

From the graph shown in Fig. 16 it is observed that for equal number of all size images, it has given 100% accuracy. For 50% of 256x256 pixels size and rest 50% of all size images used for training, it has given 100% accuracy for the images having higher sizes than 256x256. For 75% 256x256 pixels and 25% of the rest images, it has given good accuracy for the images having size higher than 256x256 pixels and the accuracy is decreased for smaller size images than 256x256 pixels.

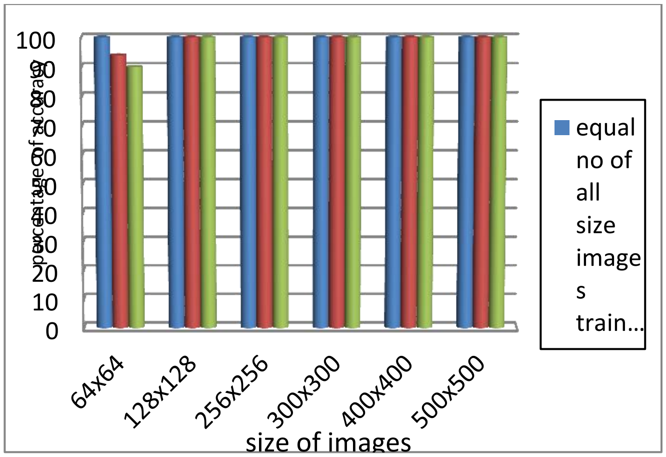


Figure 17. Recognition accuracy for training with different percentage of 300x300 size images

From the graph shown in Fig 17 it is observed that for equal no of all size images, it has given 100% accuracy. For 50% of 300x300 size and rest 50% of all size images used for training, it has given 100% accuracy for the images having all sizes except 64x64 size images. For 75% 300x300 pixels and 25% of rest images, it has given 100% accuracy for the images having all size except 64x64 pixels.

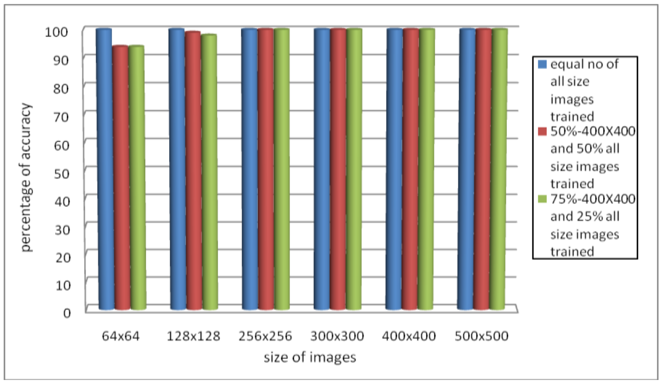


Figure 18. Recognition accuracy for training with different percentage of 400x400 size images

From the graph shown in Fig. 18 it is observed that for equal number of all size images, it has given 100% accuracy. For 50% of 400x400 pixels size and rest 50% all size images training, it has given 100% accuracy for the images having size 256x256 pixels and onwards. For 75% 400x400 pixels and 25% rest images, it has given 100% accuracy for the images having size 256x256 pixels and onwards.

From the graph shown in Fig. 19 it is observed that for equal number of all size images it has given 100% accuracy. For 50% of 500x500 pixels size and rest 50% of all size images used for training, it has given 100% accuracy for the images having size 256x256 pixels and onwards. For 75% 500x500 pixels and 25% rest all size images, it has given 100% accuracy for the images having size 256x256 pixels and onwards.

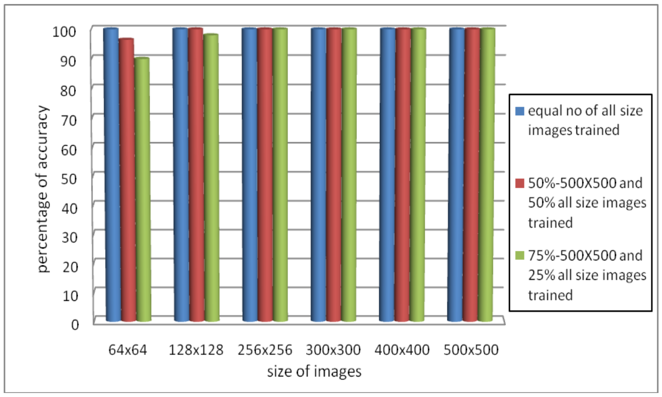


Figure 19. Recognition accuracy for training with different

V. Conclusion and Future Scope

In this paper we have proposed methodology for recognition of grains with different sizes. The proposed methodology works well when the SVM classifier is trained with equal number of images of different sizes are used for training. Other than color features it is also observed that the combined color and texture features also perform better in terms of both recognition and classification. It is also observed from the plots that the recognition accuracy is 100% for the test samples with which the classifier is trained. And for other sizes of images the recognition accuracy decreases. One interesting observation is that for combined features, one can train the classifier with higher sized images and for testing lower sized image of size 64X64 would be adequate. The whole idea behind testing the classifier is to make testing time short compared to training. One can afford longer training time but not the testing time. The testing time needs to be short to achieve real-time applications.

The color and texture features of the heap-grain samples are considered in the work. The individual color features have given recognition accuracy of 100%. Since some of the grains may have similar colors, we have used texture as another feature. The texture features have not given good results compared to color features. However, the combined features are tested, which have given poor recognition accuracies. Hence the overall recognition accuracy using the color is is found to be more suitable.

Acknowledgment

We would like to express our thanks to Dr.B.S.Anami, principal, K.L.E. Institute of Technology, for his valuable suggestions.

References

[1] Yuyong Cui, Zhiyuan Zeng and Bitao Fu (2008), *Hyperspectral Image Classification Based on Compound Kernels of Support Vector Machine* , Proceedings of the 8th International Symposium on Spatial Accuracy Assessment in Natural Resources and Environmental Sciences, China page. 263-269.

[2] Qing Song, , Wenjie Hu, and Wenfang Xie (2002) ,*Robust Support Vector Machine With Bullet Hole Image Classification*, IEEE Transactions on Systems Man and Cybernetics-Part C:Applications and Reviews, Vol.32. No.4, page.440-448

[3] Li, Jing; Allinson, Nigel; Tao, Dacheng and Li, Xuelong (2006). *Multitraining support vector machine for image retrieval*, , Vol.15, No,11, page 3597- 3601.

[4] Evgeniy Gabrilovich, Shaul Markovitch, (2004), *Text Categorization with Many redundant Features: Using Aggressive Feature Selection to Make SVMs Competitive with C4.5* Proceedings of the 21st International Conference on Machine Learning.

[5] Subhransu Maji Alexander C. Berg Jitendra Malik, (2008), *Classification using Intersection Kernel Support Vector Machines is*

efficient, IEEE Conference on Computer Vision and Pattern Recognition.

[6] Bhaskar Mehta Saurabh Nangia (2008), Detecting Image Spam using Visual Features and Near Duplicate Detection, WWW 2008.

[7] Amit David, Boaz Lerner (2005), *Support vector machine-based image classification for genetic syndrome diagnosis, Pattern Recognition Letters*, 2Vol.6, page. 1029–1038.

[8] Reda A. El-Khoribi, (2008) *Support Vector Machine Training of HMT Models for Multispectral Image Classification*, International Journal of Computer Science and Network Security, Vol.8, page 9.

[9] Yasemin Altun, Ioannis Tsochantaridis, Thomas Hofmann, (2003), *Hidden Markov Support Vector Machines*, Proceedings of the Twentieth International Conference on Machine Learning.

[10] Corinna Cortes and V. Vapnik, (1995) *Support-Vector Networks*, Machine Learning, Vol.20, page 273-297

[11] M. Aizerman, E. Braverman, and L. Rozonoer (1964).*Theoretical foundations of the potential function method in pattern recognition learning, Automation and Remote Control,* Vol.25, page 821-837.

Authors Profile

Mr.Vishwanath Kamatar obtained his Bachelor of Engineering in Computer Science & Engineering in 2005, VTU, Belagavi and M.Tech in Computer Science & Engineering in 2009, VTU, Belagavi. Since 2010 he is working as Assistant Professor in K.L.E.Institute of Techology, Hubli, Karnataka, India. He is working for his doctoral degree in Computer Science in VTU, Belgavi.

Dr.Rajesh Yakkundimath obtained his M.Tech in Computer Science & Engineering in 2007, VTU, Belagavi and Ph.D. in Computer Science & Engineering in 2016, VTU, Belagavi, and has been presently working as Associate Professor in K.L.E.Institute of Techology, Hubli, Karnataka, India. His research area of interests are Image processing, Pattern Recognition and Artificial Intelligence. He has published 30 research papers in peer reviewed International Journals.

Mr.Girish Saunshi obtained his Bachelor of Engineering in Computer Science & Engineering in 2005, VTU, Belagavi and M.Tech in Computer Science & Engineering in 2010, VTU, Belagavi. Since 2011 he is working as Assistant Professor in K.L.E.Institute of Techology, Hubli, Karnataka, India. He is working for his doctoral degree in Computer Science in VTU, Belgavi.