How to Cite:
Kumar, K. S. (2022). Swarm optimization based texture classification in extreme scale variations. *International Journal of Health Sciences*, *6*(S2), 4546–4567. https://doi.org/10.53730/ijhs.v6nS2.6079

Swarm optimization based texture classification in extreme scale variations

K. Santhosh Kumar
Assistant Professor, Providence College For Women, Coonoor
Email: bbksan@gmail.com

***Abstract*---**Texture analysis has remained a remarkably trendsetting and productive field of research in the last two decades. There has been much progress, but the impact of illumination changes on automated texture classification and segmentation has gained very less focus. Research work carried out in the field of texture identification frequently focuses on the identification of textures with intraclass changes including illumination, rotation, viewpoint and small scale variations. Consequently, variations in texture owing to modifications in scale constitute among the ones that are difficult to manage. In this research work, as the first step, the classification of textures due to vast changes in scale is studied. In order to deal with this problem, first the solution is introduced and then the scale changes are reduced based on the predominant patterns in the texture. Inspired by the challenges imposed by this issue, a novel swarm intelligence approach known as Ant Colony Optimization (ACO) algorithm is introduced for modifying the components in the hidden layers used during the network training, for the extraction of more useful semantic texture patterns. After this, an Improved Support Vector Machine (ISVM) is introduced by rendering an appropriate kernel function such that the feature weights can be used for feature classification. Since large scale changes do not exist for sure in many standardized texture databases, in order to provide support to the newly introduced extreme-scale features of texture interpretation, a novel dataset has been developed, known as the Extreme Scale Variation Textures (ESVaT), so that the performance of the proposed research output can be tested. In addition, the performance of the proposed technique is tested on the KTHTIPS2b dataset, OS dataset and another dataset artificially obtained from Forrest, yields much better performance in comparison with benchmarked ones.

***Keywords*---**ant colony optimization (ACO), improved support vector machine (ISVM), genetic algorithm network (GANet).

***Corresponding author**: Kumar, K. S.; Email: bbksan@gmail.com*
Manuscript submitted: 27 Feb 2022, Manuscript revised: 18 March 2022, Accepted for publication: 05 April 2022

Introduction

Texture is one among vital features exploited for the identification of objects or areas present in pictures. Human vision gets perception of scenes with changes in the intensity and color forming specific redundant patterns known as texture. Texture can be observed in several images ranging from multispectral remote sensed information to microscopic imaging. In order for a classification or segmentation to be successful, the image texture has to be defined clearly [1]. Texture feature extraction refers to the process of establishing definitions of a textured surface in terms of quantifiable parameters. Feature extraction is associated with the measurement of texture features in terms of a set of descriptors or quantitative feature metrics, frequently called as a feature vector [2]. Texture classification approaches are utilized for the grading of products in accordance with their visual aspects. The major objective of texture classification is about finding a best matched classification for a certain texture among the available textures. For achieving this, prior information of the classes that are to be identified is necessary. With this knowledge being available and the extraction of texture features done, classification approaches can be used for performing the classification [3-4].

Texture recognition frequently highlights on the identification of textures with intraclass changes like brightness, rotation, viewpoint, and small scale changes. Then again, in several practical applications, the big-scale changes or modifications of scale may dramatically affect the visual aspects of texture underneath, like it is found in resolved state in few images. For example, as illustrated in Figure 1, as the scale becomes increasingly and substantially coarser, from left to right, the corresponding texture category also changes; for instance, the top row (grasses) varies from the grass to lawn to a field that is close to being featureless. Especially in applications including robot vision or remote surveillance, large scale variations can happen very regularly(Fig. 1 (d)), if one single image has both near and farther distances, implying that in contexts that involve independent or machine vision it tends to become critical to carry out the texture analysis under large scale changes. As far as the best of knowledge goes, none of the available texture classification techniques can deals with extreme scale variations [5-6].

Fig. 1: A varying scale can dramatically influence the visual aspect of textures: (a) grasses, (b) wheat, and (c) trees, with the images from considerably diverse viewpoints, primarily sampling the textures from an extensive degree of underlying contiguous scale, (d) includes cases of continuous scale changes within one single image that are more challenging. All the images are acquired from the Internet.

Problem domains associated with texture, such as texture analysis or synthesis, texture segmentation or shape from texture, have gained the attention of several researchers present around the world recently. Researchers have attempted and also continually recommending several models, algorithms, and approaches in this field. An extensive analysis of different multifaceted techniques recommended, in fields such as texture definition, modelling, textural feature extraction, and segmentation/classification, empowered by exhaustive experiments and thus advising on selecting suitable algorithms in the form of solutions to these problems.

Therefore, an extremely efficient framework for identifying textures having massive scale changes is introduced, which first searches through the scale proposals and then eliminating the errors found in scale proposals by investigating the prominent texture primitives. Moreover, the proposed technique given below demonstrates efficiency as the scale proposal searching and reducing techniques are extremely quick. The important contributions made by this research work are summarized as below:

- As far as the best of knowledge goes, path-breaking research towards the issue of identifying the textures showing massive scale changes, which turn involves scale changes that are of two or more orders of magnitude.
- Once there is a change in scale of a texture, the class of the resultant texture image also varies at some edge, and these boundaries are significant to help in the identification for labeling two images belonging to the same physical fabric to belong to diverse categories of texture.
- A new network is proposed, and referred as ACO, can be used in learning more useful texture patterns employing ant colony algorithm for modifying

the modules present in the hidden layers during the process of network training.
ACO algorithm is introduced for changing the modules present in the hidden layers during the process network training, with the aim of encouraging the learning of more useful semantic textural patterns. It offers high efficiency and it prevents the problem of inbuilt parallelism and can be employed in dynamic applications (adjusts to variations like new distances, etc)

- A huge texture dataset composed of 15,747 texture images with considerable variations in scale has been contributed, with an attempt to encourage the analysis on texture and scale.

Improved Support Vector Machine (ISVM) is introduced by rendering a desirable kernel function, on which the feature weights can be applied. The Kernels render the SVM that are capable of inherently mapping the non-linearly distinguishable selected features into another dimension, where they can be linearly separated. There is huge cost incurred in the mapping of the selected features onto higher dimensions. A greater dimension indicates a bigger vector that in turn, requires more memory and increased computational complexity. Luckily, SVMs do not have to save these high dimensional vectors in an explicit manner.

Literature Survey

Sharma et al [7] proposed a Local Higher-order Statistics (LHS) for texture categorization and facial analysis, relying on the use of higher-order local differential statistics as features. In contrast with models based on the global structure of textures and faces, it has been shown recently that small local pixel pattern distributions can be highly discriminative. Motivated by such works, the proposed model employs higher-order statistics of local non-binarized pixel patterns for the image description. Hence, in addition to being remarkably simple, it requires neither any user specified quantization of the space (of pixel patterns) or any heuristics for discarding low occupancy volumes of the space. This leads to a more expressive representation which, when combined with discriminative Support Vector Machine (SVM) classifier, consistently achieves state-of-the-art performance on challenging texture and facial analysis datasets outperforming contemporary methods (with similar powerful classifiers).

Cimpoi et al [8] proposed an Improved Fisher Vector (IFV) and Deep Convolutional-network Activation Features (DeCAF) for object recognition to texture recognition. Identify a vocabulary of forty-seven texture terms and use them to describe a large dataset of patterns collected "in the wild". The port from object recognition to texture recognition the IFV and DeCAF, they both outperform specialized texture descriptors not only and also in established material recognition datasets. And, also show that proposed describable attributes are excellent texture descriptors, transferring between datasets and tasks, in particular, combined with IFV and DeCAF, they significantly outperform the state-of-the-art by more than 10% on both FMD and KTH-TIPS-2b benchmarks.

Cimpoi et al [9] proposed a Fisher Vector pooling of a Convolutional Neural Network (FV-CNN) for texture attributes recognition in clutter, using a new dataset derived from the OpenSurface texture repository. FV-CNN substantially improves the state-of-the-art in texture, material and scene recognition. This approach achieves 79.8% accuracy on Flickr material dataset and 81% accuracy on MIT indoor scenes, providing absolute gains of more than 10% over existing approaches. FV-CNN easily transfers across domains without requiring feature adaptation as for methods that build on the fully-connected layers of CNNs. Furthermore, FV-CNN can seamlessly incorporate multiscale information and describe regions of arbitrary shapes and sizes. This approach is particularly suited at localizing "stuff" categories and obtains state-of-the-art results on MSRC segmentation dataset, as well as promising results on recognizing materials and surface attributes in clutter on the OpenSurfaces dataset.

Simonyan and Zisserman [10] proposed an effect of the Convolutional Network (ConvNet) depth on its accuracy in the large-scale image recognition setting. A detailed evaluation of networks of increasing depth using architecture with very small (3x3) convolution filters was focus of this work, which reveals that a noteworthy improvement on the prior-art configurations was accomplished by pushing the depth to 16-19 weight layers. For Image Net Challenge 2014 submission, these results were considered as the base, where team secured the first and the second places in the localisation and classification tracks correspondingly. And, also show that proposed representations simplify well to other datasets, where they accomplish the state-of-the-art results.

Gómez-Ríos et al [11] proposed a Convolutional Neural Network (CNN) for recognition of coral species based on underwater texture images pose. An accurate classification model for coral texture images was the main target of this work. Current datasets comprises of a huge imbalanced classes, while the images are subject to inter-class variation. The following points were examined here 1) several CNN architectures, 2) data augmentation techniques and 3) transfer learning. Achieved the state-of-the art accuracies using different variations of ResNet on the two current coral texture datasets, EILAT and RSMAS.

Hafemann et al [12] proposed a Convolutional Neural Network (CNN) for texture classification in two forest species datasets-one with macroscopic images and another with microscopic images. The usage of deep learning techniques, in particular CNN, for texture classification in two forest species datasets-one with macroscopic images and another with microscopic images. Given the higher resolution images of these problems, present a method that is able to cope with the high-resolution texture images so as to achieve high accuracy and avoid the burden of training and defining architecture with a large number of free parameters. On the first dataset, the proposed CNN-based method achieves 95.77% of accuracy, compared to state-of-the-art of 97.77%. On the dataset of microscopic images, it achieves 97.32%, beating the best published result of 93.2%.

Akbarizadeh [13] proposed a Genetic Algorithm (GA) for texture feature extraction of satellite images. In this approach, by changing the angle of the texture, cannot change the texture feature. Initially, image texture is convolved with a trained mask, and computes its energy. A genetic algorithm through various examples of textured images has also trained a regulatory mask. On a series of aerial images, we perform the Feature extraction experiments. The effectiveness of this feature for texture discrimination process of satellite images was given by the experimental result on both the agricultural and urban images.

Liu et al [14] proposed a Binary Rotation Invariant and Noise Tolerant (BRINT) for texture classification. The proposed approach is very fast to build, very compact while remaining robust to illumination variations, rotation changes, and noise. Develop a novel and simple strategy to compute a local binary descriptor based on the conventional Local Binary Pattern (LBP) approach, preserving the advantageous characteristics of uniform LBP. Points are sampled in a circular neighborhood, but keeping the number of bins in a single-scale LBP histogram constant and small, such that arbitrarily large circular neighborhoods can be sampled and compactly encoded over a number of scales. There is no necessity to learn a texton dictionary, as in methods based on clustering, and no tuning of parameters is required to deal with different data sets. Extensive experimental results on representative texture databases show that the proposed BRINT not only demonstrates superior performance to a number of recent state-of-the-art LBP variants under normal conditions, but also performs significantly and consistently better in presence of noise due to its high distinctiveness and robustness. This noise robustness characteristic of the proposed BRINT is evaluated quantitatively with different artificially generated types and levels of noise (including Gaussian, salt and pepper, and speckle noise) in natural texture images.

Zhou et al [15] proposed a Network Dissection method that interprets networks by providing meaningful labels to their individual units. The proposed method quantifies the interpretability of Convolutional Neural Network (CNN) representations by evaluating the alignment between individual hidden units and visual semantic concepts. By identifying the best alignments, units are given interpretable labels ranging from colors, materials, textures, parts, objects and scenes. Apply proposed approach to interpret and compare the latent representations of several network architectures trained to solve a wide range of supervised and self-supervised tasks. Then examine factors affecting the network interpretability such as the number of the training iterations, regularizations, different initialization parameters, as well as networks depth and width. Finally show that the interpreted units can be used to provide explicit explanations of a given CNN prediction for an image. Experiment results highlight that interpretability is an important property of deep neural networks that provides new insights into what hierarchical structures can learn.

Liu et al [16] proposed a Genetic Algorithm Network (GANet) for texture classification. GA to change the units in the hidden layers during network training, in order to promote the learning of more informative semantic texture patterns. Finally, adopt a Fisher Vector pooling of a Convolutional Neural Network filter bank (FVCNN) feature encoder for global texture representation. Because extreme scale variations are not necessarily present in most standard texture databases, to support the proposed extreme-scale aspects of texture understanding are developing a new dataset, the Extreme Scale Variation

Textures (ESVaT), to test the performance of proposed framework. It is demonstrated that the proposed framework significantly outperforms gold-standard texture features by more than 10% on ESVaT.

Proposed Methodology

Scale Proposal

Instinctively, the scale change of a texture image, which perfectly affects its appearance, though the direct estimation of global scale is very unstable. Propose to search a set of scale proposals or candidate scale levels in a given texture image, for accomplishing the scale invariance in texture classification. The number of scale proposals will be minimized by searching among the basic dominant Texel's that occur most frequently.

Scale Proposal Searching

In general, there can be not only multiple scales, but certainly uninterrupted modifications in scale within a single image. Recognizing the entire existing scale proposals and this is the main prerequisite for texture classification with extreme scale variations to be successful. In order to recognize the candidate texture element (texel) windows, use the BING algorithm and then calculate the scale proposals based on these windows. The rationale behind the BING searcher is that it looks for the scale proposals in real time and returns almost all of the potential texels in an image. As demonstrated in Fig. 2(a), resize an input image $I \in R^{W^0 \times H^0}$ to a sequence of quantized sizes characterized by scale ratio s, to be sensitive to a range of scales. In this experiments, choose $s\ = 0.95$ and creates an image pyramid of resized images of sizes$\{(W_0 s^m, H_0 s^n)\}$, to some lower limit (here set to ten pixels), which is defined by the region of support of the features being extracted, with an 8×8 feature extraction window recommended by [17]. Then compute the Normed Gradient (NG) one feature [17] (shown in Fig. 2(b)) of the entire pyramid. Here, need to observe that deliberately downsample unconnectedly along each axis, to consider for textures having different aspect ratios (as illustrated in Fig. 2(d)).

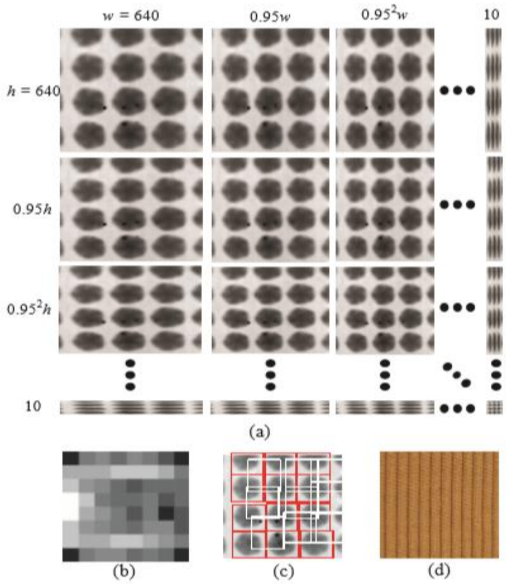


Fig. 2: Scale Proposal Searching using an Image Pyramid and the BING Searcher [14] (a) Images are downsized to obtain an image pyramid; (b) A single 64D linear model for selecting texture element proposals based on normed-gradient features; (c) Candidate texture element windows; (d) A texture image whose texels have a significantly different aspect ratio from those in (c)

Scan over the texels entire image pyramid with a 8×8 BING feature [17] for recognizing it within the texture image. As shown in Fig. 2(c), at specific scale level l a number of texels $\Omega - \{T_{l,k}\}$, indexed by location k, could be proposed. Every location in the image pyramid is characterized by it's rescaling relative to the original image, for maintaining the idea of scale clear, that is, pyramid image of size$\{(W_0 s^m, H_0 s^n)\}$ is said to be at scale $l = (s^m, s^n)$. Represent all texels found over the entire image pyramid as $\Omega = \{T_{l,k}\}$. In practice, typically recognize the thousands of potential texels over an entire image pyramid through the small thresholds for the BING searcher. The scales of these texels form the scale proposals for this texture image. Specifically, represent the set of all the scale proposals as the candidate scale proposal.

$$\mathbb{S}_{sp} = \{l | T_{l,k} \in \Omega\}$$

Scale Proposal Reduction

For an input image, the candidate scale proposals $\mathbb{S}_{sp}$ are usually redundant. Ought to minimize the number of the scale proposals for recognizing the basic texels that most frequently occur in the image, for efficiency consideration. Textures, whether they are regular or stochastic, comprises of repetitive patterns that display stationary statistics of some sort. Henceforth, the BING searcher is expected to appear repetitively and it identifies the texels. Search its similar texels over the texel proposal set from the same scale level, for each texel proposal $T \in \Omega$. From the candidate texel set Ω, only keep those texels having sufficiently many similar texels and remove the others, with the remaining texels forming a new set Ω^{re}having a corresponding reduced scale-proposal set $\mathbb{S}_{sp-re}$, a significant reduction in the number of scale proposals. Two more issues were considered here, according to this analysis; have to acquire a good reduced scale proposal set: how to compute the similarity among two texels, and how many texels were maintained. To calculate the distance between the $LBP_{8,1}^{u2}$ histograms of them, regarding the similarity measure between two texels. The candidate texel proposal set should survive with respect to texels for each $T \in \Omega$, and here recognize the set $\Omega'$of similar texels, whose similarities were huge enough when compared with some threshold η. Such that candidate T is preserved only if the number of similar texels surpasses threshold K, and only maintaining the dominant, most frequently happening texels, essentially those which are more stable and eliminates noise. The setting of the two threshold parameters η, K will be discussed.

Scale Boundary

The category of a texture image being constant at the time of some scale interval. In other words, the category of this texture image may also change, when the scale of a texture image changes significantly. The interesting part here is the location of the scale boundary, which splits the two images of the same physical material as different texture classes. According to the selected group of patches, Inspired by this finding, similarly develop a patch based method to infer boundary in scale. Provided with a texture dataset $\mathbb{S} = \{\mathbb{S}_c\}$ with C classes and for each class $\mathbb{S}_c$, randomly choose 10 images and then calculate its basis functions $\mathbb{X}_c = \{x_c\}$ on 16×16-pixel patches.

The proposed texture classification process is summarized in Fig. 3, consisting of the following steps:

1. To compute its reduced scale proposals $\mathbb{S}_{sp-re}$, for each image I in training/testing set, and it is described in scale proposal reduction.
2. To downsize each image I to acquire its image pyramid $\{I_p\}$ through its reduced scale proposals $\mathbb{S}_{sp-re}$. For the downsized images, let $I = I \cup \{I_p\}$ be the expanded set for image I. Thus, have a new training/testing set $\Theta_I^+ = \{I\}$.
3. The proposed ACO, for each image in the new training/testing set, to mine the global texture feature representation [15].
4. The extracted ACO features were categorized through Improved SVM classifier. For each image I, if its reduced scale proposals $\mathbb{S}_{sp-re}$ has M scale levels (i.e. $|\mathbb{S}_{sp-re}| = M$), then have $M + 1$ category labels for it after classification. Image I will be assigned the category label which occurs the most frequently.

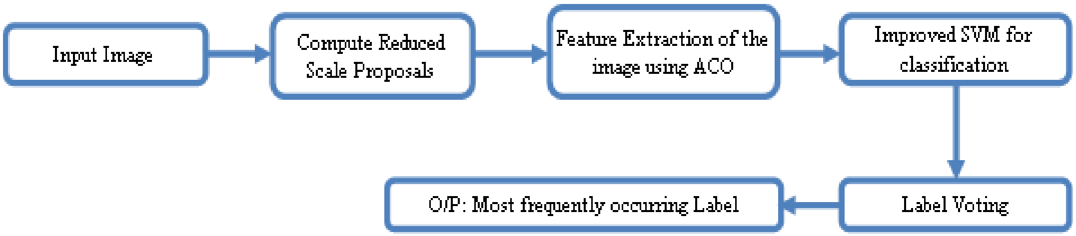


Fig. 3: Proposed Texture Classification Methodology Diagram

Input Image

On four datasets, the proposed framework is experimented, then they were synthesized dataset SForrest derived from Forrest, ESVaT, KTHTIPS2b [18] and OS.

Compute Reduced Scale Proposals

Here the each input images were downsized for acquiring its image pyramid through the reduced scale proposals. For further processing, reduced scale proposals were considered as new training set images.

Feature Extraction

The ants who live together in colonies will give a sophisticated communication system and they utilize chemical cues called as classification accuracy. The isolated ant will choose random number of features but an ant encountering previously laid classification accuracy will recognize it and decide to follow it with high probability and thus reinforce it with a further quantity of classification accuracy. The collective behaviour of a real ant colony will be represented which is a form of autocatalytic behaviour where many ants follow a trail, the more attractive that trail becomes. For recognizing the optimal ones, the features of artificial ants are: having some memory, not being completely blind and the process time is discrete. A colony of artificial ants was utilized by this technique that behaves as cooperative agents in a mathematical space where they are allowed to look for and reinforce pathways (selected features). The time when the ants are positioned on different nodes (sessions) with empty tabu lists and initial classification accuracy distributed equally on number of features connecting these sessions, an initialisation phase occurs. Ants update the level of classification accuracy while they were building their schedules by iteratively adding new sessions to the current partial schedule. Ants compute a set of feasible moves, at each time step and choose best selected features one according to some probabilistic rules on the heuristic information and classification accuracy level. The more profitable way is to select this move and resume the search, with the higher value of the classification accuracy and the heuristic information. The selected node is lined-up in the tabu list related to the ant for avoiding the selection process again. The heuristic information represents the nearer sessions around the current session. The tabu list for each ant will be full at end of every iteration and then compute and memorize the obtained cheapest schedule. For the following iteration, tabu lists was emptied and used, then the classification accuracy level will be updated, the process continuous until attain the number of iteration (stopping criteria) [19]:

Schedule Construction Stage

For measuring the transition probabilities, an ant leaves a classification accuracy trail on the connecting number of features to be collected by other ants, after each move. This begins from the initial session i, an explorer ant m selects probabilistically session j to detect next through the following transition rule:

$$P_m(i,j) = \begin{cases} \dfrac{\left[\tau_{(i,j)}\right]^{\alpha} . \left[\eta_{(i,j)}\right]^{\beta}}{\sum_{k\epsilon S_m(i)}\left[\tau_{(i,j)}\right]^{\alpha} . \left[\eta_{(i,j)}\right]^{\beta}} & if\ j\epsilon S_m(i) \\ 0 \ \ otherwise & \end{cases}$$

where

$\tau_{(i,j)}$: the intensity measure of classification accuracy deposited by each ant on the number of features (i,j). The intensity changes during the run of the program.

α : the intensity control parameter.

$\eta_{(i,j)}$: the visibility measure of the quality of the number of features(i,j). This visibility, which remains constant during the run of the program, is determined by $\eta_{(i,j)}=1/l_{(ij)}$, where $l_{(ij)}$ is the cost of move from session i to the session j.

β: the visibility control parameter.

$S_m(i)$: the set of sessions that remain to be observed by ant m positioned at session i.

The quality of the number of features(i,j) is proportional to its shortness and to the highest amount of classification accuracy deposited on it (i.e., the selection probability is proportional to selected features quality).

Pheromone Updating Stage

The classification accuracy level on the number of features will be modified by ants between sessions using the following updating rule:

$$\tau_{(i,j)} \longleftarrow \rho.\tau_{(i,j)} + \Delta\tau_{(i,j)}$$

where

ρ : the trail evaporation parameter.

$\Delta\tau_{(i,j)}$: the classification accuracy level.

The mechanism used here is: the ants through the amount of deposited classification accuracy will share the good selected features. Stagnation happen sat the time of the classification accuracy updating and this happens when the classification accuracy level is considerably varies among number of features connecting the observed schedule, which means: when compared with the other, some of these number of features have received higher amount of classification accuracy and an ant will continuously choose these selected features and neglect the others. Here, the same schedule will be built by the ants over and over again and the exploration of the search stops. By influencing the probability for choosing the next number of features, stagnation can be avoided, which depends directly on the classification accuracy level. Several ideas works according to the classification accuracy control strategy have been implemented, tested and examined, for better usage of the classification accuracy and exploit the search space of a schedule more effectively. Some of these ideas are: additional classification accuracy trail limits, smoothing of the classification accuracy trails, re-initialization of the classification accuracy trial and additional reinforcement of the classification accuracy, etc. Various approaches based on these ideas have been proposed in the upcoming section, implemented to effectively diverse the search space, and choose the best possible observation schedule.

Ant Colony System (ACS)

Ant Colony System (ACS) varies from the other ACO instances because of its strategy of constructing an observation schedule. This strategy classifies in three steps. An ant positioned on session i chooses the session j to notice by applying the following equation:

$$p_{(i,j)} = \begin{cases} \arg\max_{k \epsilon s_{m(i)}} \left[\tau_{(i,k)} . \eta_{(i,k)}^{\beta}\right] \; if q \leq q_0 \\ \quad I \;\; otherwise \end{cases}$$

where
I : a random variable selected according to the probability given by Equation 1.
q: a uniformly distributed random number to determine the relative importance of exploitation versus exploration q∈[0,..,1].
q_0 : a threshold parameter and the smaller q_0 the higher the probability to make a random choice ($0 \leq q_0 \leq 1$).

An ant located at session i samples the parameter q to move to session j, in each step of building a schedule. An ant selects the best selected features to reach the next session when ($q \leq q_0$) (exploitation). Else, the ant will probabilistically select the next session to be observed through with a bias toward the best possible selected features (biased exploration). While ants build their schedules, during they locally update the classification accuracy level of the number of features through the local updating rule as follows:

$$\tau_{(i,j)} \longleftarrow (1 - \varphi). \tau_{(i,j)} + \varphi. \tau_0$$

Where
φ : a persistence of the trail and the term (1-φ) can be interpreted as trail evaporation.
τ_0: the initial classification accuracy level which is assumed to be a small positive constant distributed equally on all the number of features the network since the start of the survey.

The classification accuracy information by dynamically changing the desirability of paths is done better and this is main target of the local updating rule. Ants will search with the help of this rule in a wide neighbourhood of the best previous schedule. The classification accuracy level is updated by applying the global updating rule only on the paths that belong to the best found schedule, when all ants have completed their schedule, since the beginning as follows:

$$\tau_{(i,j)} \leftarrow (1-\rho).\tau_{(i,j)} + \rho.\Delta\tau_{(i,j)}$$

$$\Delta\tau_{(i,j)} = \begin{cases} (C_m)^{-1} \; if(i,j)\epsilon Global - Best - Scedule \\ 0 \; otherwise \end{cases}$$

where
ρ: a classification accuracy decay parameter.
C_m : the cost of the best schedule performed from the beginning by ant m.
A greater amount of classification accuracy on the number of features of the best schedule will be given by this rule, therefore increasing the search around this schedule. Alternatively, only the best ant that took the shortest route is allowed to deposit classification accuracy. Hence, the texture features are extracted through ACO, which is further classier via an Improved SVM classifier.

Improved SVM for Classification

Improved Support Vector Machine (ISVM) is nothing but supervised learning algorithms, which is incorporated with the learning algorithms that examine the data used for classification and regression analysis. This algorithm and its other name is non-probabilistic linear classifier based on how it models and allocates the values to a training set. It is basically a two-class classifier. ISVM is a hyperplane i.e., there's a margin separating a set of non-related (negative and positive) selected features into high dimensional space through a nonlinear mapping function, therefore constructing an optimal separate hyperplane by the maximum margin among the two sets of vectors [20].

The selected features (labeled feature) are distributed into an n-dimensional space where n indicates the considered features. Each feature is the value of a particular coordinate. The selected features are either related or non-related, therefore be appropriate to and forming various classes. The classification of data based on relation with the other related data sets was the main target of ISVM. To classify the selected features make use of Hyperplanes and it forms the boundaries. Maximizing the margin between the hyperplane and the selected features was the main target here. The decision boundary that exists among the distributed data classes is chosen to be the one for which the margin is maximized. Computational learning theory motivates this idea and calls it as statistical learning theory.

For classification of the leaves, make use of the support vector machine. Into the classifier, fed the input features with a target class, followed by the mapping of training vectors into high dimensional space using a non-linear mapping function and a separated optimal hyperplane constructed by the maximum margin between the two sets of vectors. The vectors generated have class assignments and the by a maximum elect of class assignments, the classification decision is made.

A Sequential Backward Selection (SBS) processed by ISVM classifier. This method is a wrapper type, from the evaluation criterion point of view and the training process of ISVM classifier is combined with feature selection efficiently.
The following three steps were there in this ISVM algorithm:

- Train ISVM classifier with current samples and obtain information related with ISVM capability. For example, weight information of each feature when linear kernel is used in ISVM.
- According to some evaluation criterion, calculating score of each feature.
- Remove the feature corresponding to the smallest score from current feature set.

The output of this algorithm is a feature list, according to their importance the features were sorted. Weight value ω is employed to calculate evaluation criterion C_i, in the case of using linear kernel in ISVM.

$$C_i = (\omega_i)^2$$

Here, $\omega_i$is the ith element of weight vector w. If C_i is the smallest, the ith feature will be eliminated. Actually, equation (1) is an approximate measure of variation of class interval in ISVM.

$$J = \frac{1}{2}\|w\|^2$$

the weight vector after the ith feature is removed. Initially, input the sample into each sub classifier and use their respective feature subset in the testing process, by utilizing the trained ISVM to test a new sample. The mode of the new sample, according to output results of all sub classifiers.

Step 4: Thinning algorithm is a Morphological operation that is used to remove selected foreground pixels from binary images. It preserves the topology (extent and connectivity) of the original region while throwing away most of the original foreground [21] pixels. Here this algorithm is proposed for labelling purpose. In classification process, a testing image is considered being correctly classier only; it is assigned the correct subcategory label.

Results and Discussion

The experiments analysis was done using Mat lab. Matlab (matrix laboratory) is a programming language that help for a wide array of numerical and computing applications. The interface follows a language that is designed to look a lot like the notation through linear algebra. The estimation and detection accuracy were the main focus here also completeness as compared to the approaches and it will compute the effectiveness of the proposed approach.

Datasets and Experimental Setup

KTHTIPS2b has 11 texture categories and four physical samples per category. Each physical sample is imaged with three viewing angles, 4 illuminants and 9 different scales for acquiring various images. For KTHTIPS2b, one sample is available for training and the remaining three for testing, following [9].

KTHTIPS2b Textures
Fig. 4: Some example textures from KTHTIPS2b

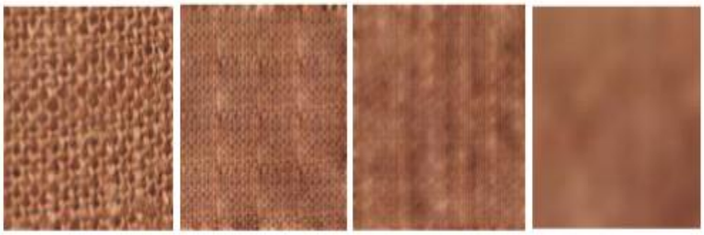


Fig. 5: Example of a Synthesized Texture at Different Scales

Experimental Results

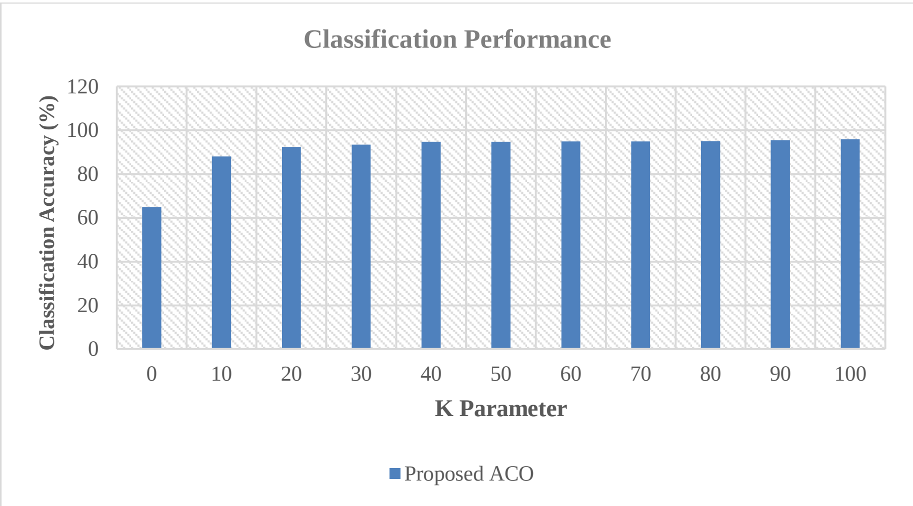


Fig. 6: Classification Performance as a Function of Parameter K

For parameter K, we regulate its value empirically as shown in Fig. 6. The performance (and computational complexity) of method gets maximized with K,[21-25] with performance very much levelling off at K = 20, so have set K = 20 for all experiments.

Results based on Screen Shot
Existing System

Input Images:

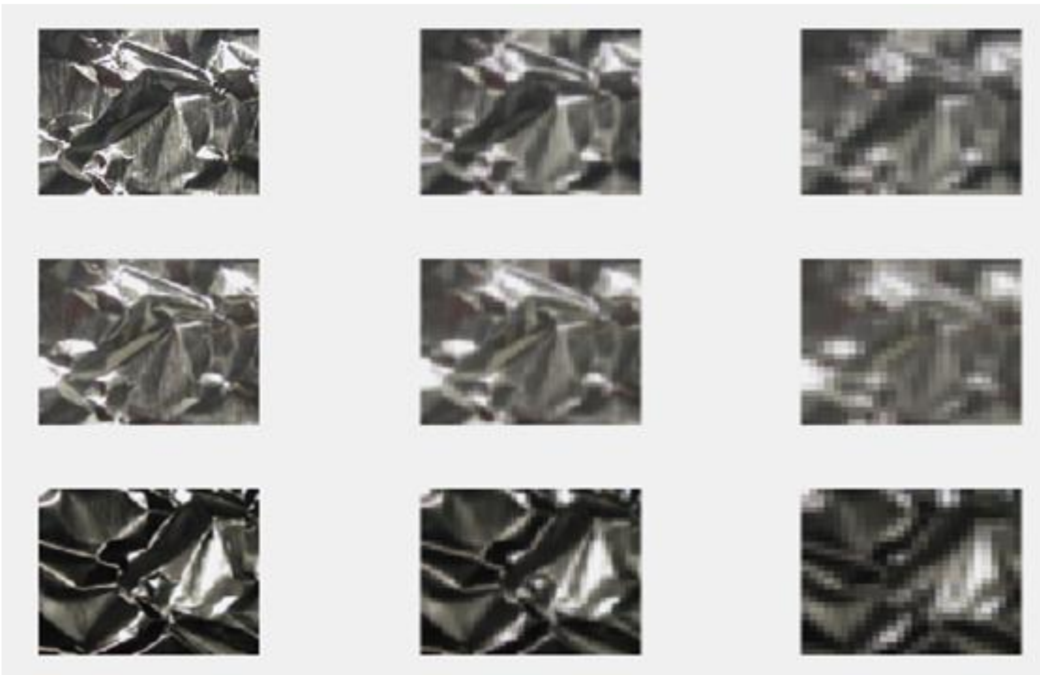
Getting Three Images

(a)

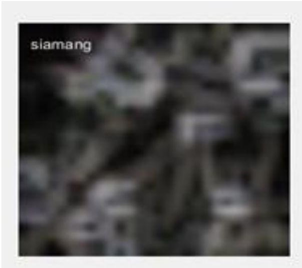
(b)

(c)

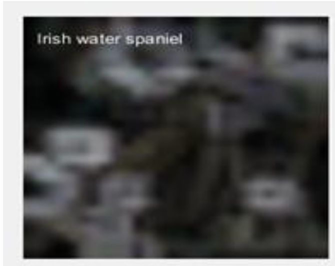
Reduce Scale Proposals



FV CNN Image Labels



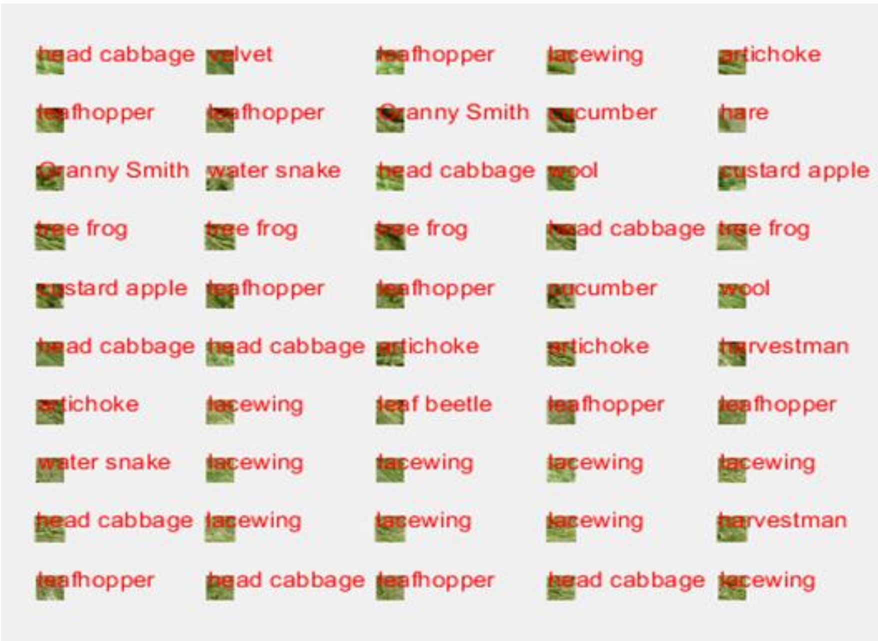
(a)



(b)

(c)

Frequent Occur Label

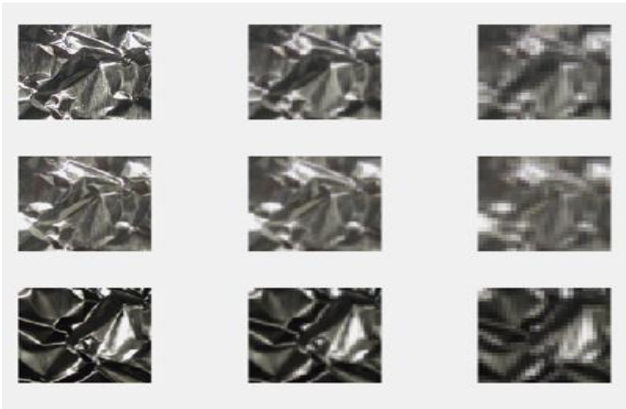



```
********************
Predicted Label
********************
     lacewing
```

Result based on Proposed System

The proposed System in which the same input images are taken for experimentation and its results are discussed below

Reduce Scale Proposals

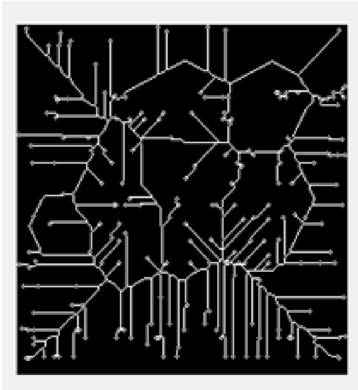


(a)

(b)

Ant Colony Optimization

Thinning Algorithm



(a)

(b)

(c)

Frequent Label Appearance


```
*******************
Predicted Label
*******************
     water snake
```

Conclusion and Future Work

Identifying the textures with intra-class variations such as illumination, rotation, viewpoint and small scale changes, were the main focus of the research in texture recognition. In contrast, on texture appearance a dramatic impact happens when change has in scale, which completely changes from one texture category to another. So, these changes were very hard to handle. In this research, the proposed method concentrates to identifying the textures with extreme scale variations, initially, searching scale proposals and then discarding errors in scale proposals by exploring dominant texture primitives. Furthermore, method is efficient since the scale proposal searching and reducing methods are very fast. A novel ACO for better texture feature learning. Almost certainly because of the extreme scale variations present in those datasets, relative to the more modest gains in KTHTIPS2b, which has less extensive scale variations. As the field of interest and the results of this study turned out to be rich and broad, there are several ways to extend it in various directions.

Few possible ways to examine this work in the near future are discussed below. Recently, bring-in the number of multi resolution analyses by many researchers such as Contourlet transform, Framelet Transform, Shearlet Transform. The extracted features from the texture region must be clear to get maximum classification accuracy and the performance of any texture classification system.

References

1. L. Liu, P. Fieguth, Y. Guo, X. Wang and M. Pietikäinen, Local binary features for texture classification: Taxonomy and experimental study, Pattern Recognition, Vol.62, Pp.135-160, 2017.
2. L. Liu, S. Lao, P.W. Fieguth, Y. Guo, X. Wang and M. Pietikäinen, Median robust extended local binary pattern for texture classification, IEEE Transactions on Image Processing, Vol.25, No.3, Pp.1368-1381, 2016.
3. A. Depeursinge, Z. Püspöki, J.P. Ward and M. Unser, Steerable wavelet machines (SWM): learning moving frames for texture classification, IEEE Transactions on Image Processing, Vol.26, No.4, Pp.1626-1636, 2017.
4. V. Andrearczyk and P.F. Whelan, Using filter banks in convolutional neural networks for texture classification, Pattern Recognition Letters, Vol.84, Pp.63-69, 2016.
5. X. Dai, J. Yue-Hei Ng and L.S. Davis, Fason: First and second order information fusion network for texture recognition, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Pp.7352-7360, 2017.
6. I. Hadji and R.P. Wildes, A spatiotemporal oriented energy network for dynamic texture recognition, Proceedings of the IEEE International Conference on Computer Vision, Pp.3066-3074, 2017.
7. G. Sharma, S. ul Hussain and F. Jurie, Local higher-order statistics (LHS) for texture categorization and facial analysis, European conference on computer vision, Pp.1-12, 2012.
8. M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed and A. Vedaldi, Describing textures in the wild, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Pp.3606-3613, 2014.

9. M. Cimpoi, S. Maji and A. Vedaldi, Deep filter banks for texture recognition and segmentation, Proceedings of the IEEE conference on computer vision and pattern recognition, Pp.3828-3836, 2015.
10. K. Simonyan and A. Zisserman, Very deep convolutional networks for large-scale image recognition, Computer Vision and Pattern Recognition, Pp.1-14, 2014.
11. A. Gómez-Ríos, S. Tabik, J. Luengo, A.S.M. Shihavuddin, B. Krawczyk and F. Herrera, Towards highly accurate coral texture images classification using deep convolutional neural networks and data augmentation, Expert Systems with Applications, Vol.118, Pp.315-328, 2019.
12. L.G. Hafemann, L.S. Oliveira and P. Cavalin, Forest species recognition using deep convolutional neural networks, 22nd IEEE International Conference on Pattern Recognition, Pp.1103-1107, 2014.
13. G. Akbarizadeh, A New Feature Extraction Method Using Genetic Optimization for Texture Recognition of Satellite Images, Journal of Agricultural Engineering and Biotechnology, Vol.1, No.3, Pp.68-73, 2013.
14. L. Liu, Y. Long, P.W. Fieguth, S. Lao and G. Zhao, BRINT: binary rotation invariant and noise tolerant texture classification, IEEE transactions on Image Processing, Vol.23, No.7, Pp.3071-3084, 2014.
15. B. Zhou, D. Bau, A. Oliva and A. Torralba, Interpreting deep visual representations via network dissection, IEEE transactions on pattern analysis and machine intelligence, Vol.41, No.9, Pp.2131-2145, 2018.
16. L. Liu, J. Chen, G. Zhao, P. Fieguth, X. Chen and M. Pietikäinen, Texture classification in extreme scale variations using GANet, IEEE Transactions on Image Processing, Vol.28, No.8, Pp.3910-3922, 2019.
17. M.M. Cheng, Y. Liu, W.Y. Lin, Z. Zhang, P.L. Rosin and P.H. Torr, BING: Binarized normed gradients for objectness estimation at 300fps, Computational Visual Media, Vol.5, No.1, Pp.3-20, 2019.
18. H. Zheng, L. Fang, M. Ji, M. Strese, Y. Özer and E. Steinbach, Deep learning for surface material classification using haptic and visual information, IEEE Transactions on Multimedia, Vol.18, No.12, Pp.2407-2416, 2016.
19. C.H. Liao, J.M. Wu, J. Du and L. Zhao, Ant colony optimization inspired resource allocation for multiuser multicarrier systems, 9th International Conference on Wireless Communications and Signal Processing (WCSP), Pp.1-6, 2017.
20. S. Peng, Q. Hu, Y. Chen and J. Dang, Improved support vector machine algorithm for heterogeneous data, Pattern Recognition, Vol.48, No.6, pp.2072-2083, 2015.
21. Senthil, P. "Image Mining In ranking Approach under Interval-Valued Hesitant Fuzzy Set Gr Selection." International Journal of Scientific Research in Computer Science, Engineering and Information Technology 1, no. 2 (2016): 105-114.
22. Senthil, P., and M. Suganya. "Exchanged Nonlinear Third Order Differential Equation Ordinary Differential Equation." Journal for Research| Volume 4.05 (2018).
23. Senthil, P., Suganya, M., Baidari, I. and Sajjan, S.P., 2020. Enhancement Sushisen algorithms in images analysis technologies to increase computerized tomography images. International Journal of Information Technology, pp.1-13.

24. Senthil, P. "Enhanced of image mining techniques the classification brain tumor accuracy (ENCEPHALON)." Int J Comput Sci Mobile Comput 5.5 (2016): 110-115.
25. Senthil, P. "An Improved Gradient Boosted Algorithms Based Solutions Predictive Model (Trade)." Asian Journal of Managerial Science 5.1 (2016): 30-40.