Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 143 (2018) 493–501

Procedia Computer Science

www.elsevier.com/locate/procedia

8th International Conference on Advances in Computing and Communication (ICACC-2018)

A Support Vector Machine with Gabor Features for Animal Intrusion Detection in Agriculture Fields

Saieshwar Radhakrishnan, Ramanathan.R.*

Department of Electronics and Communication Engineering,
Amrita School of Engineering, Coimbatore
Amrita Vishwa Vidyapeetham, India 641112

Abstract

Animal intrusion in agricultural fields has been a pestering problem for farmers, especially during monsoon when they try to maximize their yield. This paper puts forth an image processing and machine learning based approach to classify the animal as threat and hence alert the farmer. The image is segmented into parts using Watershed algorithm. The features are extracted from the training set by using 2D Gabor filter bank. Classification is done using Support Vector Machines algorithm. Percentage accuracy for each test image is analyzed. Training set has been increased in a step wise manner in order to find the minimum possible combination of test images and filter bank and hence increase the efficiency of the model compared to the existing models.

Keywords: Animal intrusion; Watershed Algorithm; Support Vector Machines; Gabor filter.

1. Introduction

Animal intrusion is a serious problem in Bihar, Chhattisgarh, Uttarakhand and especially regions near forest areas. The problem is affecting both farmers and the government. Farmers lose their high yielding crops like rice and wheat, which is their main source of income to support their family for rest of the non-monsoon months. Government in turn has to pay a compensation to the farmers. The primitive solutions practiced in the villages have not been able to be

* Corresponding author.
E-mail address: r_ramanathan@cb.amrita.edu

1877-0509 © 2018 The Authors. Published by Elsevier B.V.

10.1016/j.procs.2018.10.422

overcome the problem completely, thus there is an immediate need for intelligent monitoring system without human requirement to automatically recognize the intruding animal and alert the farmer**.** The approach here uses a camera that acquires images in regular intervals of the field. The images are then sent to the processing unit, in which the image gets classified. First step in classification is segmentation. Segmentation is done to extract the multiple objects in the image, to analyze if that segment is the threat animal. This paper uses the marker based Watershed algorithm. The algorithm is based on the idea of flooding the marked regions and when different markers meet i.e. in case of a maxima, a barrier is created which is the contour. After segmentation, the next step involved is feature extraction using Gabor filters. In frequency domain, Gabor filter is the convolution of Fourier transform of harmonic function and the Fourier transform of Gaussian function. Gabor filters have been extensively used in many image processing applications such as optical character recognition, iris and fingerprint recognition. Since the filter has different frequencies and different orientations in different directions it has been extensively used in extracting text only regions in a page and facial expression recognition. The final step is classification using Support Vector Machine (SVM). Linear SVM which is a supervised learning algorithm is focused in this paper, as the training data is labelled. SVM has huge applications in image processing such as text and hypertext classification, classification of images, hand-written character recognition and it has also been able to classify proteins with up to 90% accuracy. The rest of the paper is organized into five parts. They are as follows. Section 2 contains the problem definition. Section 3 contains the proposed approach. Section 4 discusses the results and inference. Section 5 presents the conclusion and scope for further research. This approach has been used in variety of other applications like Satellite image classification [1], classification of images based on color, texture etc. from a database [2], Heterogeneous Face Detection [3] and in other areas like malware detection [4].

2. Problem Definition

The problem definition can be broken to three parts: Since image classification is the primary goal, there are parameters in each of the processes which needs to be fixed. The first step, segmentation, can be implemented in a more simple and efficient way using thresholding and contour detection, but, when the objects are touching each other, it is not possible to separate them so that is the reason for choosing marker based watershed algorithm. The results thus obtained from watershed from watershed algorithm are segmented perfectly in most of the cases. In the next step, which is feature extraction using Gabor filters, the minimum number of filters for the filter bank needs to be fixed. Different filters are obtained by varying the parameters like wavelength, angle, shape of Gaussian noise etc. Using a minimum number of filter bank which is sufficient enough to classify the image with maximum accuracy will reduce the processing speed drastically.In the final step which is the training and prediction using SVM, size of training set needs to be fixed. There might be instances in which the image taken by the camera has noise added to it, contrast variations during different time in a day, the object (animal) might be bending down or jumping, or in some cases the acquisition percentage may also, still the classifier has to correctly identify the animal, this would require updating the training set of the algorithm in a step wise manner so that it is both efficient and accurate. Thus, analyzing all the parameters and building an efficient classifier will be the primary focus of research.

3. Proposed Approach

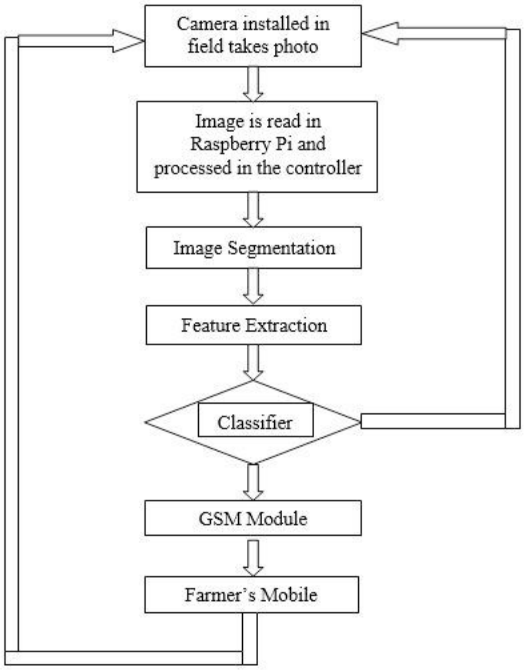


Fig 1: Flowchart of the proposed approach

Fig 1 shows the flowchart of the proposed approach, the model uses an installed raspberry pi or any other standalone processor with camera embedded, in the agricultural field. The camera captures the field in regular intervals. The image is then segmented to separate the foreground alone from the image and then features such as mean and standard deviation are extracted from the image and classified using SVM. The segmentation, feature extraction and classification will be explained briefly in the upcoming parts of this section. Once an animal is classified as a threat animal, the processor being programmed, activates the GSM module connected to it which in turn sends a message to the farmer's mobile. Python was used for coding the algorithms with opencv image processing toolkit.

3.1. Image Segmentation

Before applying the watershed algorithm to the image, the image would require some pre-processing. The first step is Otsu-thresholding. Based on the histogram plot of the image, the algorithm decides the threshold level and converts the image into a binary image. Then, noise removal using Morphological opening. Next, extracting foreground alone, this is done by dilation followed by distance transform and thresholding using the distance transform output as the threshold [4]. Finally, we have to apply the markers i.e. assign labels to background and all the foreground regions. Applying watershed algorithm to the processed image obtained will yield the segmented images [5].

3.2. Feature Extraction using Gabor Filters

Gabor filters have been extensively used in image processing for feature extraction. Gabor filters gives a coefficient matrix which gives multi resolution analysis. Here a 2D Gabor filter has been used for the purpose of feature extraction. A 2D Gabor is a product of Gaussian function and a sinusoidal wave in the time domain and it is a convolution of the transforms of the Gaussian and sinusoid in the frequency domain. It is represented by equations (1.1).

$$g(\gamma,\eta,\varphi,\lambda)=\exp(\frac{x'^2+\gamma 2y'^2}{2\sigma^2}).\cos(\frac{2\pi x'}{\lambda}+\varphi) \tag{1.1}$$

$$x'=x\cos\theta-y\sin\theta \tag{1.2}$$

$$y'=x\sin\theta-y\cos\theta \tag{1.3}$$

The Gabor filter bank can be obtained by varying the parameters like λ, γ, φ and θ. In the above equations, x and y represent image coordinates; s is the standard deviation of Gaussian function which is usually set to 0.56 λ; λ is the wave length of cosine equation; γ characterizes the shape of Gaussian and θ represents the channel orientation and takes values in interval (0, 360). Since it is symmetric, θ varies from zero to 180. The response of this filter is nothing but the convolution given by the equation (1.4) [6].

$$\iint I(\varepsilon,\eta)g(x-\varepsilon,y-\eta)d\varepsilon d\eta \tag{1.4}$$

The value of θ and σ must be taken under some considerations to make the choice of filter to be optimum. The steps to be followed are:

- Fixing values of the parameters, kernel size is taken as 3, λ is 2.7 and σ is taken to be 0.56 times wavelength.
- A bank of 7 Gabor filters are thus obtained by fixing the above parameters and varying θ from $\frac{\pi}{8},\frac{7\pi}{8}$
- Training set and test images are passed through the filter bank.
- The mean and standard deviation are obtained for each individual image, these are taken as features for classification.

3.3. Classification using SVM

SVM is a powerful tool used for classification. It is superior to neural networks as its formulation is in such a way that there is only one minima and it is the global minima. Linear-SVM has been used in this paper. The training data is linearly separable, two hyperplanes are formulated such that the distance between them is maximum. The region bounded by these two hyperplanes is called the margin. With proper dataset, rescaling these hyperplanes can be described by the following equations:

$\vec{w}.\vec{x}-b=1$ (Anything > or = to this boundary belong to one class) (1.5)

$\vec{w}.\vec{x}-b=-1$ (Anything < or = to this boundary belong to one class) (1.6)

Geometrically, the distance between these two hyperplanes is $\frac{2}{\|\vec{w}\|}$, so to maximize the distance between the planes we want to minimize $\|\vec{w}\|$. Our aim is to minimize $\|\vec{w}\|$ subject to $y_i(\vec{w}.\vec{x_i}-b)\geq 1$, for $i=1,\ldots,n$. The $\vec{w}$ and b that solve this problem determine the classifier [7].

The training set which forms the $\vec{x_i}$ of the classifier includes the features obtained from the pictures of dog and Nilgai and 12 edited picture of each of the two animals (classes). Parameters like contrast, brightness, colour and saturation is varied to obtain the edited images. So a set of 26 training images are used for classification algorithm. Here the two classes are: Nilgai (represented by -1) and dog (represented by +1).The test images of this model, which is the $\vec{w}$ includes:

- Images of Nilgai and dog.
- Images with % acquisition of the animal of 50 and 25.
- Images with Noise such as Gaussian noise, Salt and pepper noise, Speckle noise added to it.
- Images rotated by 15, 30, 45, -15, -30, -45 degrees.
- Images with combinations of the above cases.

So, a total of 9877 test images were created using the parameters mentioned above and convoluted with the filter bank and classified using the SVM algorithm.

4. Results and Discussion

The main crux of the result lies in finding the minimum number of filters to be used in the filter bank and in selecting the size of the training set. The method proposed has the advantage of having just 7 filters. To come to this conclusion, we have tried in combinations which include (7, 7), (7, 24), (24, 7), (24, 24) which represents (number of filters for training set, number of filters for test images) and the results are as follows:

Here the training set includes one image each of Nilgai and dog and the test images include brighter, edited, rotated images of Nilgai and dog.

Table 1: Percentage Accuracy in all 4 cases above mentioned for a set of three test images for Nilgai and dog.

Type of image	% accuracy Nilgai	% accuracy dog	% accuracy Nilgai	% accuracy dog	% accuracy Nilgai	% accuracy dog	% accuracy Nilgai	% accuracy dog
Cases	7 filters (training) 7 filters (testing)		7 filters (training) 24 filters (testing)		24 filters (training) 7 filters (testing)		24 filters (training) 24 filters (testing)	
Brighter Image	71.42	0	61.9	95.32	71.43	85.71	90.47	95.24
Edited Image	28.57	100	9.5	100	14.28	100	4.76	100
Rotated Image	100	71.42	95.23	90.47	100	100	100	100

From Table 1, (7, 7) case has comparatively same accuracy, except in two cases which is brighter and edited image of dog. To improve the results the training set of the model is increased. Now, the training set contains brighter image, 10 types of edited images each with different contrast and saturation for both dog and Nilgai. As mentioned in Section 2, the test images include the combinations of rotated, acquisition percentage 50, 25 of the animal, noisy images. The accuracy bar graph is plotted.

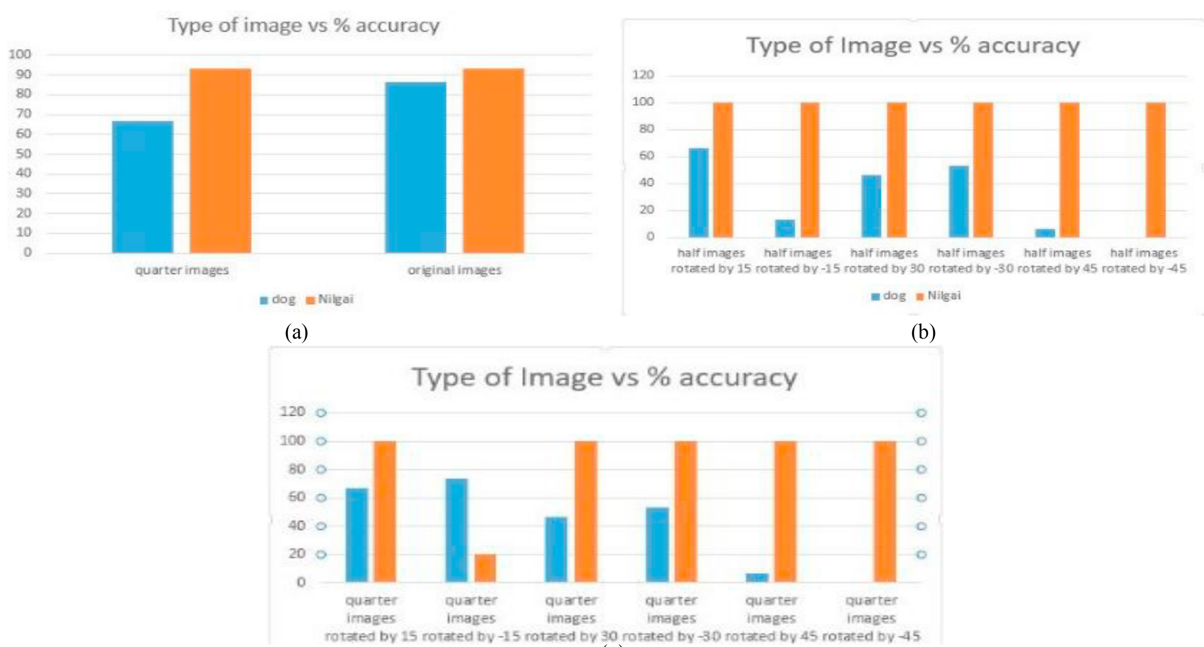


Fig. 2: Accuracy plot for (a) Original images, 25% available, (b) 50% available, rotated, (c) 25% available, rotated.

From the bar graphs in Fig 4, percentage accuracy of Nilgai is consistently above 80%, the model detects Nilgai even if the acquisition of the animal is one fourth available or if it is rotated by 45 degrees, whereas, for dog the percentage accuracy falls with decreasing % acquisition of the dog

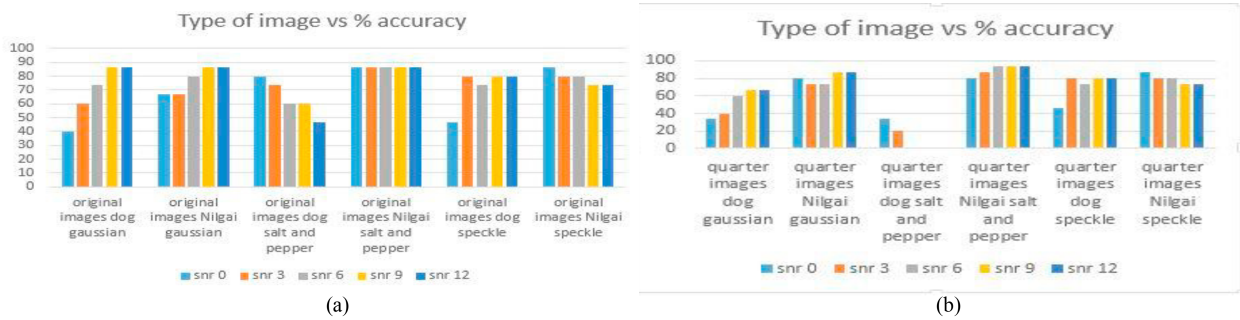
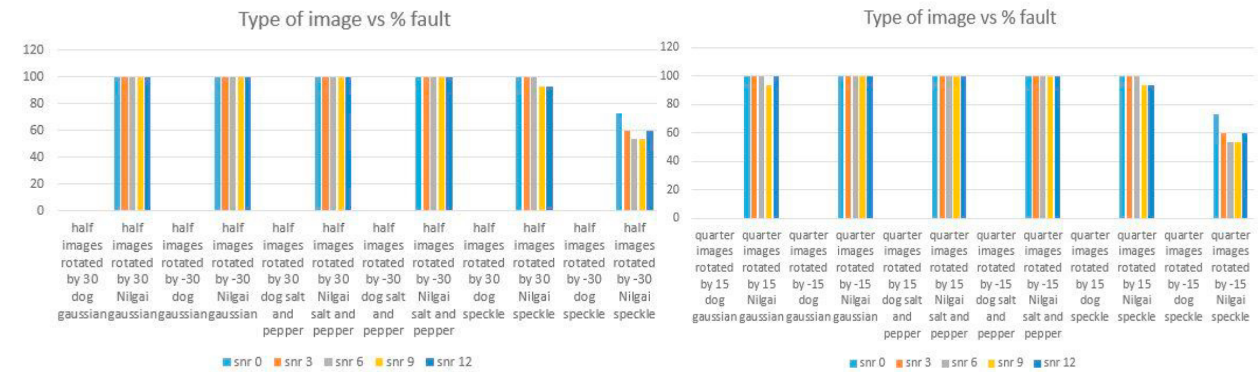


Fig. 3: Accuracy plot with different noise for (a) original images (b) acquisition of 25% available.

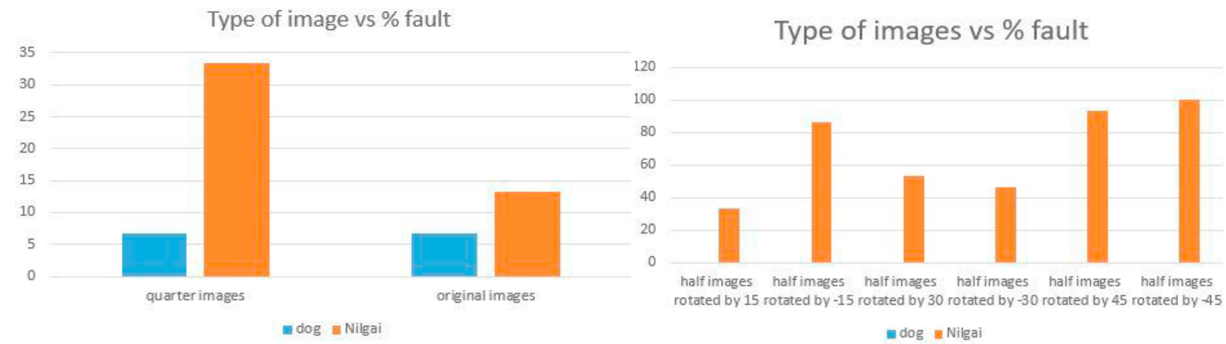
In Fig 5, the image is tested with adding Gaussian, salt and pepper and speckle noise with varying SNR, we observe that, in the case of Nilgai, the percentage accuracy although it falls when snr decreases, is consistently above 75%, hence it is highly likely to get detected even in presence of noise. But in case of dog, when only noise is added the detection rate is above 60% for most of the cases but as the acquisition percentage of the animal decreases, the detection percentage reduces drastically.



(a) (b)

Fig. 4: Accuracy plot with different noise for (a) acquisition of 50% and rotated (b) acquisition of 25% and rotated.

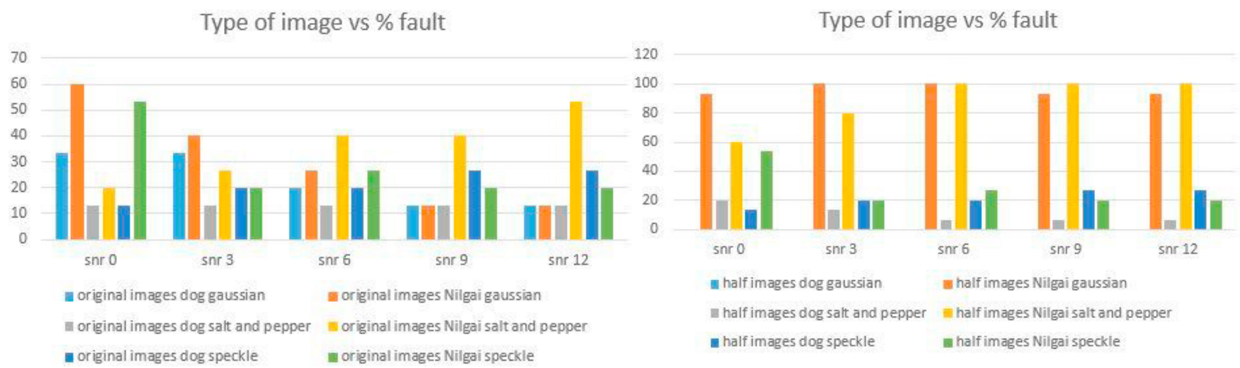
In Fig 6, the image is subjected to both rotation as well as noise. The results shows that, Nilgai gets classified correctly in all the cases, whereas, for dog, in case of acquisition of 50%, it gets classified to a maximum of 45% of the cases when varying SNR in Gaussian and speckle noise and that gets reduced when acquisition decreases to 25%.



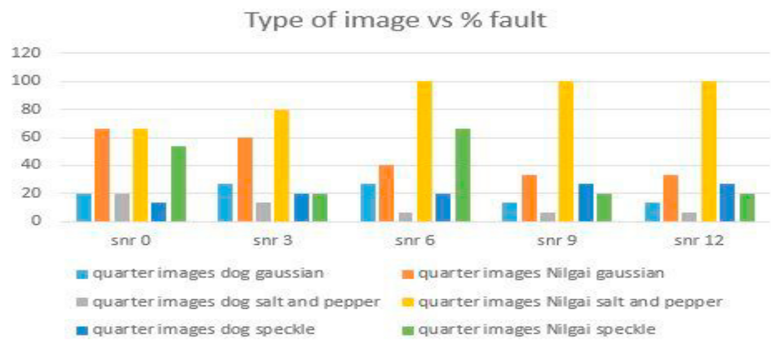
(a) (b)

Fig. 5: % fault for dog and Nilgai (a) original images, acquisition of 25% available, (b) 50% available and rotated.

Fig 7 deals with similar cases as Fig 4, but checks for the case when dog is classified as Nilgai and vice versa. The % fault increases with decrease in acquisition percentage as in (a). In (b) and (c), as rotation angle increases, dog getting classified as Nilgai increases.



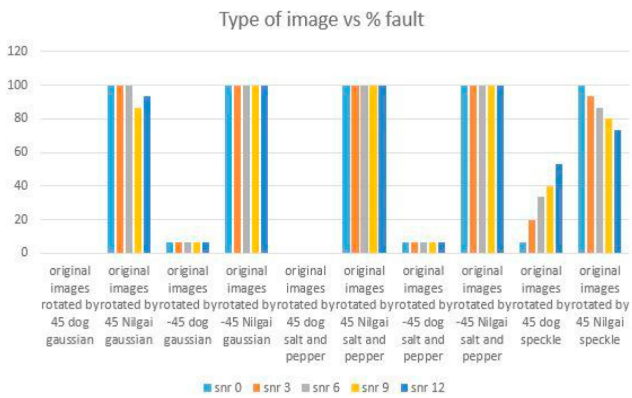
(a) (b)



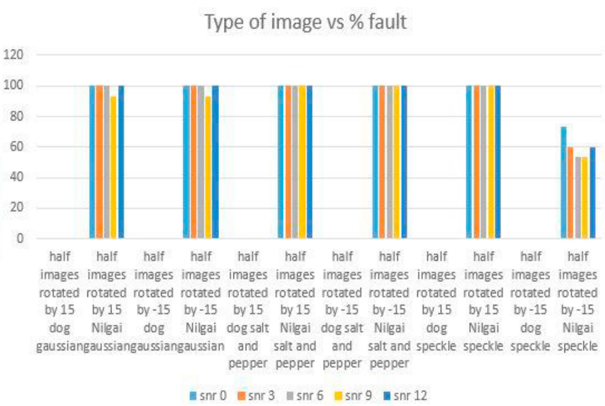
(c)

Fig. 6: % fault plot with different noise for (a) original images (b) acquisition of 50% (c) acquisition of 25%.

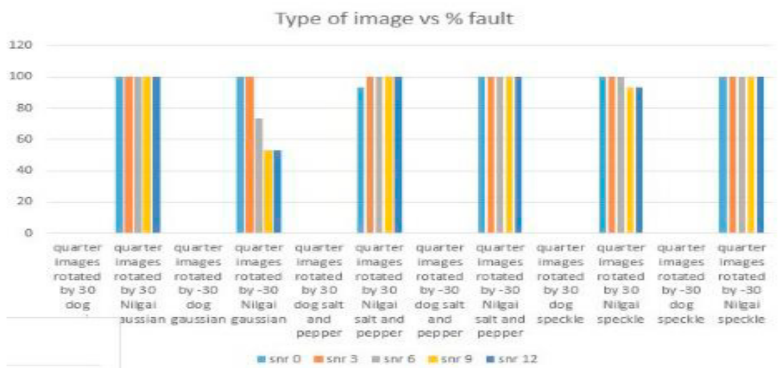
Fig 8 deals with similar cases as Fig 5, Nilgai being detected as dog in most of the cases are less than 25%. But, the case when dog being detected as Nilgai increases with increase in SNR. This is analogous to Fig 7 because as SNR increases the signal component increases and it follows the same trend as without noise.



(a)



(b)



(c)

Fig. 7: % fault plot with noise for (a) original images, rotated (b) 50% available, rotated (c) 25% available rotated

Fig 9 deals with similar cases as Fig 6, percentage of Nilgai being classified as dog in most of the case is less than 20%. But dog being classified as Nilgai is consistently above 90% for different noise with varying SNR.

4.1. Comparison with existing methods

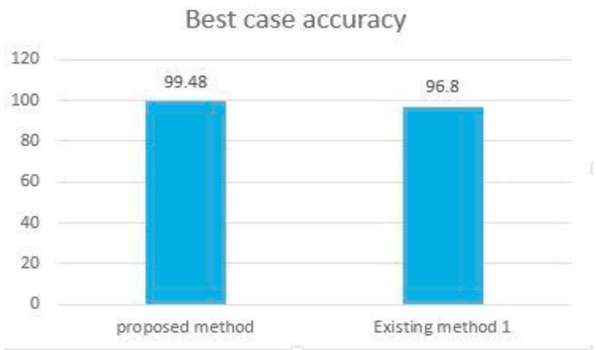


Fig. 8: Percentage accuracy for an animal to be detected correctly in proposed method and an existing method

The existing method [8] uses 1.4 million images as training set and the test image contains 105,000 images. But, the test image in that model does not contain noise added to it. The proposed method tests images with three types noise: Gaussian, salt and pepper, speckle and image acquisition of 25% and 50% of the animal available, rotation of images by 15, 30, 45, -15, -20, -45 and the combinations of all these properties as shown in the previous sections. Thus, ensuring all the possibilities of the image captured by the camera installed in the field. This model uses 26 training images and 9877 test images, the ratio if scaled can improve the overall efficiency. As shown in Fig. 10. The best case accuracy in classification of an animal is 99.48% in this model compared to 96.8% in the existing method. The model has an overall accuracy in classifying an animal with all the above constraints as 54.32%.

5. Conclusion

Thus, an efficient image classifier was developed by segmenting the objects of an image using watershed algorithm and extracting features like mean and standard deviation using Gabor filters and using those features for classification using Support Vector Machines. The model has an overall average percentage accuracy of 54.32. Scope of further research is to improve the training set to classify the animal in a more efficient way with higher accuracy and hence could possibly be implemented to prevent animal intrusion in agriculture fields.

References

[1] Manali Jain, Amit Sinha, Classification of Satellite Images through Gabor Filter using SVM, International Journal of Computer Applications April 2015; (0975 – 8887) Volume 116 – No. 7.

[2] M.Janani, K.Nandhini, K.Senthilvadivel, S.Jothilakshmi, Digital Image Technique using Gabor Filter and SVM in Heterogeneous Face Recognition, International Journal of Engineering and Science April 2014, Vol.4, Issue 4 PP 45-52 Issn (e): 2278-4721, Issn

[3] P. Poornachandran, Praveen, S., Ashok, A., Krishnan, M. R., and Dr. Soman K. P., Drive-by-download malware detection in hosts by analyzing system resource utilization using one class support vector machines, Advances in Intelligent Systems and Computing 2017, vol. 516, pp. 129-137.

[4] Image Segmentation with Watershed algorithm [Online]. Available: http://opencv-python-tutorials.readthedocs.io/en/latest/py_tutorials/py-imgproc/py_watershed/

[5] Jos B.T.M. Roerdink, Arnold Meijster, The watershed Transform: Definitions, Algorithms and Parallelization Strategies, Fundamenta Informaticae 41 (2001) 187–228.

[6] R.Ramanathan, Arun.S.Nair, L.Thaneshwaran, S.Ponmathavan, N.Valliappan, Dr. K.P.Soman, Robust Feature Extraction Technique for Optical Character Recognition, International Conference on Advances in Computing, Control, and Telecommunication Technologies 2009, 573-575.

[7] R.Ramanathan, Arun.S.Nair, V.Vidhya Sagar, N.Sriram, Dr. K.P.Soman, A Support Vector Machines Approach for Efficient Facial Expression Recognition, International Conference on Advances in Recent Technologies in Communication and Computing 2009, 850-854.

[8] Mohammed Sadegh Norouzzadeh, Anh Nguyen, Margaret Kosmala, Ali Swanson, Meredith Palmer, Craig Packer and Jeff Clune, Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning, arXiv:1703.05830v5 [cs.CV] 15 Nov 2017.