

DEEP LEARNING AND STATISTICAL MODELS FOR DETECTION OF WHITE STEM BORER DISEASE IN ARABICA COFFEE

Sanat Bhandarkar^{1,*}, Rutha Prasad¹, Vaibhav Agarwal¹, R. Hebbar², D. Uma¹, Y. B. Venkata Reddy³, Y. Raghuramulu³ and K. Ganesha Raj²

¹ PES University, BSK III Stage, Bengaluru, Karnataka, India (umaprabha, sanatbhandarkar, ruthaprasad, vaibhavagarwal)@pesu.pes.edu

² RRSC-South, NRSC/ISRO, Marathahalli, Bengaluru, India (hebbar_kr, ganeshraj)@nrsc.gov.in

³ CCRI, Balehonnur, Karnataka, India (dr.coffeeboard@nic.in, venkatcoffee@gmail.com)

KEY WORDS: White stem borer, Arabica coffee, detection, deep learning, transfer learning, Auto encoders, Environmental factors, Multiple Regression

ABSTRACT:

Early detection of crop pest and disease is very critical for taking up suitable control measures to reduce the loss of economic yield. Coffee is an important commercial crop in India which is affected by pests and diseases every year resulting in major yield loss. White stem borer (*Xylotrechus quadripes*) is the most serious pest of coffee (*Arabica* sp.) in India causing substantial loss of yield every year. Detection of the infestation in its early stage is quite challenging. In this regard, image pattern recognition techniques offer cost effective and scalable solutions. An image library was created representing different stages of the plant infestation using camera/mobile devices. Our Convolutional Neural Network (CNN) models use these images of healthy and infested plants for early detection of white stem borer infestation. The overall methodology included image processing, machine learning, supervised transfer learning and unsupervised auto-encoding techniques to solve the problem of early detection and severity of the infestation. Using the Inception v3 transfer learning model, we obtained average accuracy of 85.5% which is quite encouraging with limited image datasets. We explore Unsupervised Autoencoder models, which can work with limited image datasets. In addition, statistical analysis of long-term climatic factors such as temperature, rainfall, humidity and luminescence is explored for reliable detection and diagnosis of the infestation. Based on the encouraging results, a mobile application is proposed for near real time monitoring of WSB infestation to help the coffee planter's community

1. INTRODUCTION

Xylotrechus quadripes is a species of beetle in the family of Cerambycidae. Due to its habit of boring through the stems of coffee arabica plants, it is commonly known as the Coffee White Stem Borer (CWSB), and is considered to be one of the most lethal pests for coffee. The impact of WSB infestation on coffee yield is extremely pronounced, reducing the annual yield in India by almost 60%, and resulting in an annual financial loss of around \$20 million. Farmers today employ many techniques to tackle this infestation, namely manual collection of adult pests and uprooting of affected plants, swabbing stems with repellent chemicals and usage of pheromone traps, which is one of the best solutions available currently. These solutions, however, are incapable of dampening the effect of the infestation, which raises the need for a more robust solution. We propose a Deep Learning and Image Processing based approach to detect the infestation in its early stages, to enable adoption of effective control and remedial measures to save the crop yield. Successful adoption of these measures can result in a global reduction in coffee prices by approximately 150 %, which would impact coffee farmers in a significantly positive manner, as Coffee is a popular cash crop.

*Corresponding author

2. LITERATURE REVIEW

2.1 White Stem Borer pest

Xylotrechus quadripes beetle affects coffee arabica plants by boring through the stem on reaching adulthood. It is a diurnal insect, which means bright sunlight favours its activity. The adult beetle lays its eggs inside the plant through crevices present in the stem. These eggs hatch into grubs, which penetrate the plant through the stem making small tunnels through the plant. The emergence of ridges on the stem is in accordance with the grubs eating through the bark and reaching the hard wood. These grubs' tunnel through the hardwood by cutting the hardwood into a fine whitish powder, which can be observed on internal inspection of affected plants. They then pupate inside the plant, and pupating larvae exit the plants on reaching adulthood through the stems of the plant. These exits create holes, or bores, on the plant which render the plant dead in most cases. The emergence of the pest peaks in two distinct periods called flight periods - the summer flight, which lasts from April to May, and the winter flight which lasts from October to December. It had initially been observed that winter flight was significantly longer than summer flight, however further research indicates that both flight periods have an almost equal contribution to infestation. It was also found that predominantly higher number of males emerged during both flight periods. Infested plants demonstrate drooping, wilting and yellowing of

leaves and emergence of holes on the stems of the plant. Ringing of the barks of main roots occurs below ground level and even propagates higher up above the ground. Infestation by the pest often results in death of the plant, and those trees which survive the attack become highly susceptible to termite attacks. The cumulative effect of pest infestation is extremely detrimental to coffee farmers, resulting in an average economic loss of yield of around 2 to 20% annually. Previous work and Limitations

Extensive studies have stated that more than 850 different kinds of insects have been found to attack coffee, out of which the coffee leaf miner, the coffee berry borer and the coffee White Stem Borer are the most prominent (Ziska, 2018). All previous studies have primarily dealt with the detailed study of the infestation causes and effects, and subsequently preventive and management measures to ensure reduction in loss of yield. Research has been done on detection and prediction of onset of other diseases using Machine Learning, DL (Mohanty Sharada P., 2016) and ImgProc (Barbedo, 2013), but none of those techniques have been applied to the detection of White Stem Borer infestation (Beyene, 2018). The major aim of our research work is to provide an ensemble approach to detection and prediction of WSB infestation using Deep Learning (DL), Transfer Learning (TL) and Image Processing (ImgProc). Although previous solutions have been proven to work to a certain extent, there are some inherent limitations, which impose a restriction on obtaining a concise solution. Previous solutions involving traditional Machine Learning techniques were found to perform well for training data, but could not deliver same accuracy results on real-time data and these models could not be deployed easily due to lack of portability. The effect of environmental parameters like temperature, pressure, humidity, sunlight and rainfall have been hypothesized to have a significant correlation (Kutwayo 2013; Magina 2009; Reddy, 2011; Santaram, 2008) with emergence of adult CWSB beetles, but all prior solutions have been image data driven, and fail to take these factors into account.

3. METHODS AND METHODOLOGY

3.1 Data Collection and Dataset

The extensive dataset collected by Coffee Board, was used for training and testing of all our models which comprised of 2425 images of healthy (400), infested (800) and discarded/dead (400) crops from coffee plantations in Belur, Chickmagalur, Mudigere, Sakleshpur, Somwarpet and Yeslur regions of Karnataka. The images are acquired in uncontrolled environments. The disadvantages include non-uniform lighting, contrast differences, external noise and blurriness. This further increases the challenge to detect the infestation using images. Our approach to tackle these drawbacks, and develop a solution that can work on these images, reinstates that it can be used for other similar situations where lab-controlled image acquisition method takes immense time and effort.

3.1.1 Environmental Factors Systematic environmental data collected by Central Institute Coffee Research in the research farm at Balehonnur, Karnataka

during 1999-2006, was analyzed in the present study. The dataset included temperature (minimum maximum), sunlight, rainfall and humidity along with number of total beetles emerged.

3.2 Pre-processing

A salient feature of non-regularized Convolutional Neural Networks is that there is a high capacity for learning. This means that noise and legitimate data can be learnt equally well. Thus, there arises the need for an adequately large dataset which has been cleaned and structured. Data Augmentation addresses the two possible shortcomings which may arise- lack of data and presence of noisy data. The potency of any model has a direct relationship with the quality of input data. Augmenting an existing dataset with proper instances of the domain being dealt with, can increase accuracy substantially.

The skewed distribution of image samples for a particular infected class may induce bias in feature learning and hence, sample datasets require augmentation in terms of both quality and quantity. Autoencoder approach using Artificial Neural Networks (ANNs) can be implemented for augmenting the datasets. These models possess the ability to efficiently learn the encoding or representation of data in an unsupervised fashion. Autoencoders consist of two sub-models- an encoder and a decoder. The encoder sub-model reduces the dimensions of the input image by performing compression. This compressed image is then passed to the decoder sub-model, which tries to reconstruct the original image from the compressed representation. Variational Autoencoders can be used as potent generative models, owing to the need of the user to explore variations in existent data in a specific direction, and not entirely randomly. These models can all be used to generate new, distinct data from existent data, with the required variations needed to be the same kind as the previous existent data. This effectively leads to augmentation of the existing dataset with new, valid instances of data. A small dataset can be increased in size exponentially without the need for collecting more data from external sources. For below implementations, simple data augmentation techniques of rotating, inverting, sheering has been used, to increase the lesser numbered classes.

For Environmental data, Maximum Temperature, Minimum Temperature, Sunshine and Humidity were averaged and Rainfall and Total Beetles were summed over a period of seven days and outliers were removed. The number of total beetles was normalized for all years to maintain the scale for Multiple Regression models. The data was split into Summer season and Winter season for better modelling. Multi-variable Non-linear Regression models were built using the first five years of data and the last two years' data was used for checking model accuracy. Non-Linear regression was used as statistical tests showed that linear regression will not be able to model the weather trends properly.

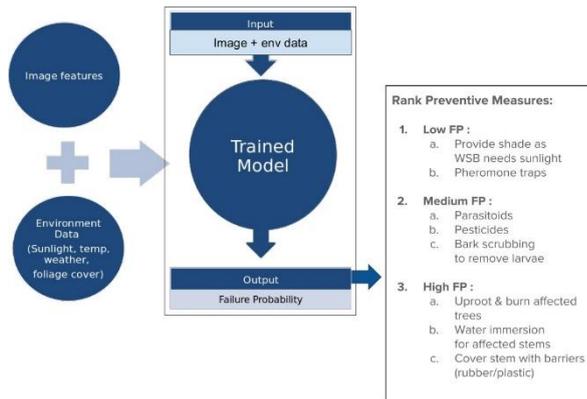


Figure 1. System Design

3.3 Methodology

3.3.1 Image Processing The dataset used for training and testing of all our models was given to us by the Coffee Board of India, which comprises 2425 images of healthy, infested and dis- carded (dead) crops directly from coffee plantations, and not in controlled lab environments. These wild raw geo-tagged images were given as input, and processing of images was done to enhance the quality of the input. ImgProc techniques used include contrast enhancement, background removal, denoising, edge detection, texture-based clustering and filtering. Experimentation was done to implement these techniques Artificial Neural Net- works, which yielded better results when compared to brute force feature engineering algorithms. Features considered for extraction were estimated leaf size, yellowing and wilting stems and branches, surrounding foliage cover, and holes, ridges or cracks on the stem.



Figure 2. Healthy Plants. Image is prominently filled with white leaf shaped patches

Enhancing contrast, converting into grey scale, performing Otsu’s line segmentation, thresholding, Healthy plants exhibit prominent leaf size and more foliage cover. Unhealthy plants show lesser foliage. Converting to greyscale is advantageous in highlighting morphology but at the risk of losing subtle color variations (Fig 3, Fig 4). Intuition suggests abundance of green and brown in all class of images may trick DL models, and decrease prediction accuracy, but subtle colour variations are picked up very well during feature extraction in deep models and thus aids better learning. Since previous methods involve converting images to grey scale, to retain color enhancing contrast, k-means clustering to find dominant color (Fig 4), thresholding specific colors has been applied to achieve segmentation. Thresholding dark brown and shadow colors

after clustering, to achieve segmentation. (Fig 5, Fig 6)



Figure 3. Unhealthy plant. Observable shrunken leaf size and reduced foliage cover. Image is prominently filled with black patches and shadows



Figure 4. Brighter shades of Light Green for Healthy plants, Deeper shades of Brown and Dark Green for Unhealthy plants.



Figure 5. Healthy plants show bushy green stems and berries.



Figure 6. Unhealthy plants show prominent yellowing stems and leaner foliage.

3.3.2 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are DL models which possess the ability to extract image features from

input data, without the need for feature engineering. Since there is no need for feature engineering, CNNs are extremely adept at handling highly complex and convoluted features, thereby making them extremely capable image classifiers. Deep Convolutional Neural Networks (DCNNs) are CNNs with a large number of convolutional layers. These multiple layers are trained to work together to construct a vast and complex feature space. The complexity of features learnt increases on traversal of the network, which means that lower order features like blob or edge detection are learnt at the initial few layers, and higher order features are progressively learnt over other layers. The final layer features are fed into the classifier, which may consist of one or more layers.

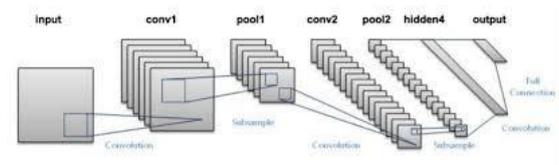


Figure 7. Convolutional Neural Network Architecture

As it can be observed from Figure 7, a CNN model comprises layers which apply local filters, and these filters are stacked in a particular order.

Convolution: Each convolutional layer has a filter, which is a rectangular matrix of values. If the image matrix is larger than the filter, the filter slides over sections of the image matrix, each time producing a new pixel value. The weights for the convolutional layer are the same for each neuron present in the layer, and these weights indicate the convolutional channel.

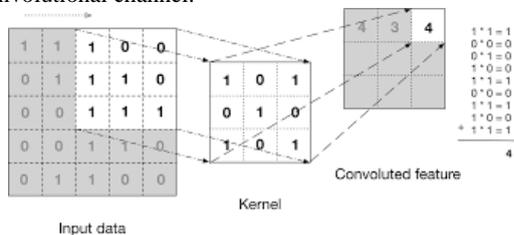


Figure 8. Convolution

Pooling: The pooling layer performs non-linear down sampling of data. It partitions the data into multiple non-overlapping blocks or rectangles and outputs the maximum value in each block. The driving idea is that the exact location of a feature in the matrix is not as important as its location relative to other features. These layers perform dimensionality reduction by reducing the number of parameters to be considered, thereby decreasing computational requirements. These also help in controlling over-fitting of data, which is why these layers are usually present in between successive convolutional layers.

Fully Connected: These layers are usually present at the end of the network-after multiple convolutional and pooling layers. A fully connected layer forms connections between all of its neurons, and all neurons of the previous network. High level feature extraction and reasoning occurs in these layers.

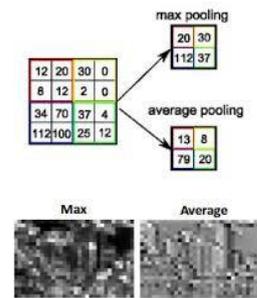


Figure 9. Pooling

3.3.3 Transfer Learning The complexity of detection of pest infestation can be handled by DL techniques, namely Convolutional Neural Networks (CNNs). However, the overhead of model training time increases exponentially with increase in the dataset size, leading to the requirement of a faster solution. TL utilizes the complexity of DL without the overhead of training time. A Deep Convolutional Neural Network is pre-trained on a very large dataset, and the weights from this model are used either as initialization or for fixed feature extraction based on the requirement. This discards the need to train the entire model from scratch, thereby making prototyping and deployment much faster. Google's Inception v3 model was used to implement TL.

3.3.4 Convolutional Neural Network with Transfer Learning The features extracted and learnt by CNNs on a particular source can be transferred to augment learning features of another distinct, but related target. Low level features can be transferred to the target for learning new complex features in the target domain, but this is applicable only if there is sufficient data available. In the absence of sufficient data, the task of learning high level features for the target domain becomes more difficult. However, if there is a significant similarity between source and target domains, TF can be applied to use the feature space generated for the source domain to learn complex features in the target domain.

3.3.5 Inception v3 model GoogleNet, also known as Inception v3, is one of the most robust Deep Convolutional Neural Network (DCNN) models designed to date. The principal focus of this model is to create a good local network topology, and then stack these topologies one above another to create a complex network of networks. The model has 22 layers and each layer is a Convolutional Neural Network called an Inception module. This extremely complex architecture is capable of identifying extremely intricate details in input data. A pre-trained Inception v3 model was used to classify our image input, only by retraining the final classifier layer on our input data. The idea of avoiding training all layers from scratch and simply retraining the final layer exponentially reduced training time.

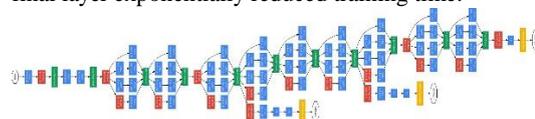


Figure 10. GoogleNet Architecture

3.3.6 Multi-Variable Non-Linear Regression Non-Linear Regression is a regression model in which the dependent variable is modelled by a function which is a non-linear combination of the model parameters and one or more dependent variables. LASSO regularization was used to improve model accuracy. It performs variable selection (setting the coefficient of non-contributing variables to 0) and performs regularization to improve prediction accuracy and improve the interpretability of the model.

3.3.7 Support Vector Regression Support Vector Machines (SVMs) can also be used as a regression method. A linear learning machine learns a non-linear function by the learning machine being mapped into a high dimensional kernel induced feature space. Support Vector Regression (SVR) is the regression algorithm. The value of the parameter C determines the margin size. A larger C results in a smaller margin and vice versa. It is chosen in a way that it minimizes the misclassification rate of the model on testing data.

3.3.8 Kernel Ridge Regression Kernel ridge regression (KRR) combines Ridge Regression (l_2 -norm regularized linear least squares) with the kernel trick. It thus learns a linear function in the space induced by the respective kernel and the data. For non-linear

kernels, this corresponds to a non-linear function in the original space. The form of the model learned by KRR (loss function: squared error loss, combined with l_2 regularization) is identical to support vector regression (SVR) (loss function: insensitive loss, combined with l_2 regularization).

3.4 Implementation

The Inception v3 model has been used as it was designed to perform image classification on our image datasets. The entire 22-layer model was trained with the values of weights being learnt from scratch. Due to the extreme complexity of this model, training time was exponentially higher than TL. For from-scratch, the Inception v3 model achieved a final training and validation accuracies achieved were higher at 87% and 75% respectively for 296 epochs using the Adam optimizer with a learning rate of 0.001. Fine-tuning the model and tweaking hyper-parameters like learning rate, optimizer used, batch size and activation functions was found to increase overall validation accuracy by 23.7% initially.

For transfer learning approach, the same Inception v3 model architecture is used. The pre-trained model classifies the Imagenet dataset, a huge generalized group of over 2000 classes. In general, when approaching TL, we first need to compare the similarity in the dataset and the similarity in the model outputs. Since the WSB plant dataset is very specific and not as generalized as Imagenet, the model might require extensive retraining to understand and extract both, high level and low-level features during training. The default number of output classes is over 2000, the top classification layer is popped off, and a new softmax classification layer of 4 classes is reattached. This change in the architecture can be considered as drastic, and thus we presumed that the training required a few layers of the model to be made trainable. We froze the initial layers and

made the last 7 trainable. We experimented with multiple optimizers and loss functions, such as categorical cross-entropy, MSE, SGD and Adam, to name a few. Categorical cross-entropy, Adam optimizer and a 0.2 dropout resulted in the best predictions.

Observing the performance of supervised training models and limitations in interpreting the final predictions (it offers only discrete classes instead of intuitive continuous predictions which can be extrapolated) (Fig 13,14,15), we aimed at developing a more balanced solution, by using unsupervised feature learning. Even though the dataset is exhaustive and classified, we considered learning the images without relying on the labels, through unsupervised learning. Autoencoders, unlike supervised models, the output is the same as the input, and the model focuses on learning features from the images to achieve maximum reconstruction accuracy. As features are extracted in each layer, from input images to produce the same images as output, the model learns to represent the same data in multiple ways, and is unbiased by input labels, unlike supervised feature learning, as reconstruction loss is used to increase accuracy during back-propagation instead of classification loss.

We built a CNN-based autoencoder with symmetric stacking of filter layers (16,8,4,8,16), with Reluactivation and uniform padding for each layer. The output layer has sigmoid activation, and the whole model is compiled with adam optimizer and categorical cross-entropy as the loss function. SGD and categorical cross-entropy performed equally.

Using the features extracted by the autoencoder, we train a classifier ANN. Combining the predictions of both approaches, we

can expect a better overall unbiased output. We pop off the decoder of the autoencoder, and get the extracted features from the encoder layers. Feeding in the encodings of each input image and its respective label, to train the ANN.

The ANN is fully connected with 2 layers (8,4) of Reluactivations, and a softmax output layer. A global average 2D pooling layer is used on the flattened input encodings. Compilation is done using Adam optimizer and Categorical cross-entropy.

For Environmental Data models, Maximum Temperature, Minimum Temperature, Sunshine, Rainfall and Humidity were used as the independent/predictor variables and Total Beetles was the dependent variable. The models were trained on first five years of data (1999-2004) and tested on the last two years of data (2005-2006) for both Summer and Winter. Root Mean Squared Error (rmse) was used as the model accuracy metric.

The degree of the Non-Linear Regression Model was chosen as 2 for it to be able to model the variance in the dependent variable properly. Another Non-Linear Regression Model was trained with the same degree and LASSO regularization was used to improve the model

performance and improve the prediction accuracy.

The Support Vector Regression models for both Summer and Winter were trained using the Radial Basis Function (rbf) kernel with the C value set as 100 so that the model can learn and adapt to the higher beetle emergence values.

The Kernel Ridge Regression models for both Summer and Winter were trained using the Radial Bias Function (rbf) kernel. The L2 norm term in ridge regression is weighted by the regularization parameter alpha. Larger the alpha value, the smoothness constraint will be higher. Lower the alpha value, the model is closer to becoming just a plain Least Squares Regression Model. A large value of alpha would lead to a stricter fit and a small value of alpha would lead to a high value of coefficients. The alpha value was set to 0.1 for both Summer and Winter models.

4. RESULTS AND DISCUSSION

4.1 Results on Image Data

Using TF for supervised learning, the highest accuracy attainable was 87% in 60 epochs, for 512x512 input image sizes. Further training and tweaking, resulted in a drop of accuracy, as the model tends to overfit on the training data.(Fig 11)

Epochs	Image Size	Accuracy
80	256 x 256	0.55
120	256 x 256	0.68
30	512 x 512	0.72
60	512 x 512	0.84

Figure 11. Accuracies of Transfer Learning on Raw images

We approximate, the model will perform better for higher resolutions images. On processed images, the model achieves accuracy faster than on raw images, but stagnates later (Fig 12). About 250 raw images were taken from each class and converted to processed images for training and testing (1000 images in total). We approximate the model will gain much higher accuracy if more processed images are used.

Epochs	Image Size	Accuracy
50	256 x 256	0.61
100	256 x 256	0.72
40	512 x 512	0.77
60	512 x 512	0.87

Figure 12. Accuracies of Transfer Learning on Processed images



Figure 13. Ranked Stage Predictions on Raw Images



Figure 14. Ranked Predictions on Raw Images. False predictions give hints on how the features extracted by the model, from the image, correlate to other classes, even pick out false labels. Such intuitions can be extrapolated and correlated.

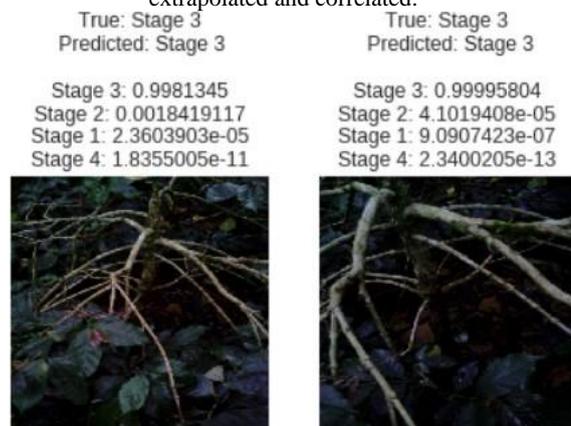


Figure 15. Ranked Stage Predictions on Processed Images

True/Predicted	Stage 1	Stage 2	Stage 3	Stage 4
Stage 1	20	0	0	0
Stage 2	0	82	2	2
Stage 3	0	0	12	0
Stage 4	2	2	52	26

Figure 16. Confusion Matrix of 200 Raw Images, of 84% TF model

Based on the similar results on both raw and processed

images, we conclude that our solution works well on raw images equally and thus can get predictions faster during real-time usage, as no image processing needs to be done once the images are captured.

Using the CNN-based autoencoder, the highest reconstruction accuracy attainable was 93% in 60 epochs (for 512x512) and 50 epochs (for 256x256) input image sizes, both raw and processed. Using CNN-based layers, ensure shift invariance and low overfitting. Using pure layers might have given better accuracy, but at the risk of overfitting. (Fig 17)

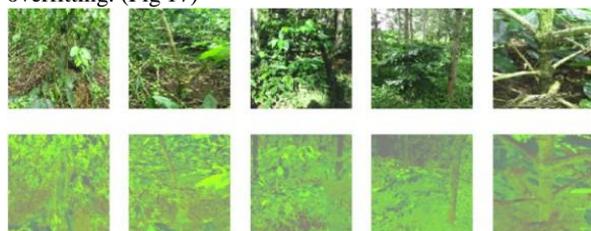


Figure 17. Reconstruction of Raw Images using Autoencoder

Using ANNs as the final classifier, the highest accuracy attainable was 75% in 80 epochs, for 512x512 input image encodings. Further training and tweaking, resulted in stagnation. (Fig 18)

Model Combination	Image Size	Raw	Processed
AED + ANN classifier	256 x 256	0.68	0.72
AED + ANN classifier	512 x 512	0.72	0.75

Figure 18. Accuracies of training ANNs on AutoEncoder Encodings

4.2 Results of Environmental Data

Among the Non-Linear Regression models, the model with Lasso Regularization performed better for Summer Data as the normal model was overfitting a lot. For Winter data, neither of the models could show satisfactory performance (Fig 19).

Model	Summer	Winter
NLR ^a	99.601	111.311
NLR(LR) ^b	104.495	119.065
SVR ^c	111.786	110.634
KRR ^d	131.459	131.049

^aNLR = Non-Linear Regression

^bNLR (LR) = Non-Linear Regression with Lasso Regression

^cSVR = Support Vector Regression

^dKRR = Kernel Ridge Regression

Figure 19. Results (RMSE) of Environmental Data Models

The Kernel Ridge Regression (KRR) models were also either overfitting or not capturing any trend at all. The KRR model for Summer was relatively better than all the other models trained on

Summer data as it was best able to capture the pattern for beetle emergence. (Fig 20)

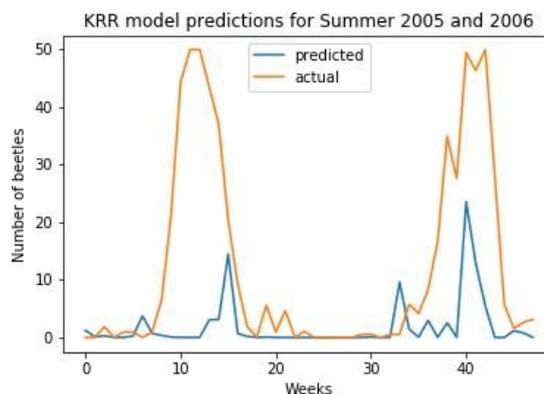


Figure 20. KRR Model predictions for Summer 2005 and 2006

The Support Vector Regression (SVR) models were also either overfitting or not capturing any trend at all. The SVR model trained on Summer data did manage to capture the beetle emergence pattern to some extent. The SVR model for Winter data was relatively better than all other models trained on Winter data as it was best able to capture the pattern for beetle emergence. (Fig 21)

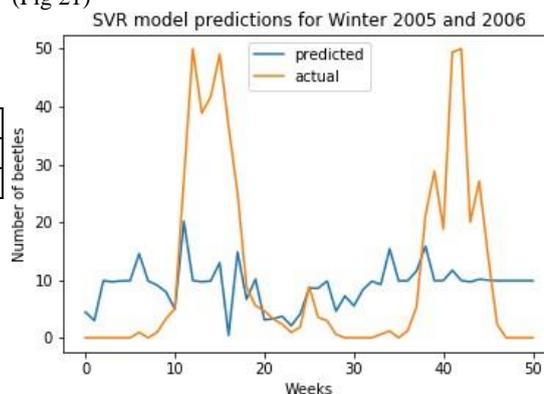


Figure 21. SVR Model predictions for Summer 2005 and 2006

Although RMSE values for any of the models are not good enough for the models to be taken as robust statistical models. But seeing their performance with less data and the fact that some models are able to capture the pattern of beetle emergence to some extent, we approximate, the models will work well with larger amounts of data

5. FEASIBILITY AND LIMITATIONS

As an insight, we advocate that laboratory tests are ultimately always more reliable than diagnoses based on visual symptoms alone, and oftentimes early-stage diagnosis via visual inspection alone is always challenging. Our solution revolves around 2 types of data, images and weather data, and we chose DL approaches for both. In order to develop accurate image classifiers for the purposes of diagnosis, we needed a large, verified dataset of images of infested and healthy plants. For similar diagnosis problems on other datasets, particularly smaller datasets, we estimate that the same solutions would work efficiently, but with customization and experimentation on epochs and classifier tweaking.

Further, most previously implemented approaches depended on plant datasets which were collected in a controlled environment (lab environments, with standard lighting and acquisition parameters), our solution is built to work on both controlled and uncontrolled image acquisition scenarios. Additionally, it is important to note that although training large neural networks can be very time-consuming, the trained models can classify images very quickly, which makes them also suitable for consumer applications on smartphones. Supplemented with location and time information for additional improvements in accuracy, these DL models can be easily deployed in smartphones, unlike computation-heavy and hardware-heavy solutions such as pure image feature algorithms. With ever improving number and quality of sensors on mobile devices, and the ease of scaling models, we consider it likely that highly accurate diagnoses via the smartphone are only a question of time. The focus of our study has been to understand WSB infestation and build a model to help detect probable infestation with a well defined lead period. However, the scope of our study is not absolute. The statistical models built by the analysis of the environmental data of Balehonnur provided did not produce results which were conclusive enough to conclude that extrapolation could be done to larger and more number of areas. Such an outcome can partly be attributed to the scarcity of data, the data available was not sufficient enough to train a robust statistical model. Further research must be done to build a sufficiently robust statistical model. The dataset was compiled by including images captured in a few districts of Karnataka, making it susceptible to skewness. Consideration of nature of the environment in which these pictures were captured also affect implementation methodology. If the environment is proven to be deterministically controlled, traditional Machine learning techniques can be used to perform feature engineering to extract the required features from the images. Since our dataset was not compiled in a fully controlled environment, feature engineering could not be done owing to increased complexity of the features. DL techniques needed to be implemented to discard tedious feature engineering. ImageProc techniques like segmentation were not applied for symptoms depicted by individual parts. The entire plant image was considered, making the focus on the entire plant rather than on specific parts of the plant.

6. CONCLUSION

WSB infestation is extremely fatal to the coffee planter's community, and easily poses to be one of the biggest threats known to coffee yield. Traditional methods failed due to the peculiarity of the infestation and the need for a better, automated diagnostic solution was established. Previous solutions proposed in this direction were a considerable improvement over existing solutions, but these solutions still came with some inherent limitations and could not solve the problem entirely. Our solution to the early detection of pest infestation has been established to theoretically overcome the limitations of previous solutions by 2 points. The first being it has reliable performance on even raw uncontrolled images and the second being, it

includes external features like environmental data points to improve accuracy of prediction and successfully determine the probabilistic occurrence of pest infestation in the near future with a significant enough lead time. However, the scope of our study and feasibility of our current solution is limited by the points mentioned in the Limitations section. We intend to continue working to expand the currently proposed solution to overcome the stated limitations and to integrate the statistical model along with the ImageProc and DL model to provide a concise prediction.

ACKNOWLEDGMENTS

The authors are grateful to Dr. Shantanu Chowdhury, Director, National Remote Sensing Centre, ISRO for providing us an opportunity to carry out the study at RRSC-South/NRSC. We sincerely thank officials of Coffee Board for providing relevant field photographs, environmental parameters and valuable suggestions for successful completion of the study. We are thankful to Dr. S.S Shylaja, Chairperson, Department of Computer Science, PES University, for giving us the opportunity and her support for carrying out the study. We thank PES University for providing facilities to carry out the study.

REFERENCES

- A 2017 Guide to Semantic Segmentation with Deep Learning. <http://blog.qure.ai/notes/semantic-segmentation-deep-learning-review>
- A Brief History of CNNs in Image Segmentation: From R-CNN to Mask R-CNN. <https://blog.athelas.com/a-brief-history-of-cnns-in-image-segmentation-from-r-cnn-to-mask-r-cnn-34ea83205de4>
- Barbedo, Arnal Garcia, Jayme, 2013. Digital image processing techniques for detecting, quantifying and classifying plant diseases. SpringerPlus. <https://doi.org/10.1186/2193-1801-2-660>
- Beyene, Hailay Narayan A Joshi, DR Kotecha, Ketan, 2018. Plant Diseases Prediction Using Image Processing And Machine Learning Techniques: Survey. International Journal of Computer Application. <https://dx.doi.org/10.26808/rs.ca.i8v1.23>
- Kutywayo D, Chemura A, Kusena W, Chidoko P, Mahoya C, 2013. The impact of climate change on the potential distribution of agricultural pests: the case of the coffee white stem borer (*Monochamus leuconotus* P.) in Zimbabwe. <https://doi.org/10.1371/journal.pone.0073432>
- Machine Learning, 2012, A Probabilistic Perspective Murphy, K.
- Magina F. L.K, 2009. Spatial distribution and temporal abundance variations of three important insect pests of coffee in Kilimanjaro region, Tanzania. Morogoro. University of agriculture.
- Mohanty Sharada P., Hughes David P., Salath Marcel, 2016. Using Deep Learning for Image-Based Plant Disease Detection. Frontiers in Plant

Science. <https://doi.org/10.3389/fpls.2016.01419>

N, Aswathanarayana Reddy, 2011. Studies on behaviour and management of coffee white stem borer, *xylotrechus quadripes chevrolat* (Cerambycidae: coleoptera)

P. - chapter 14.4.3, pp. 492-493, The MIT Press, 2012

Santaram, Akundi RK, Sabir SR, Mythrasree Seetharama, H RV, Rao, 2008. White Stem Borer Resistance in Coffee: Perspectives on Breeding, Management and Consumption. XXII International Colloquium on Coffee Science.

Thapa, Sushil Lantinga, Egbert, 2016. Infestation by Coffee White Stem Borer, *Xylotrechus quadripes*, in Relation to Soil and Plant Nutrient Content and Associated Quality Aspects. *Southwestern Entomologist*. <https://doi.org/10.3958/059.041.0209>.

Thapa, Sushil Lantinga, Egbert, 2017. Growing Coffee in the Shade: A Strategy to Minimize the Prevalence of Coffee White Stem Borer, *Xylotrechus quadripes*. *Southwestern Entomologist*. <https://doi.org/10.3958/059.042.0205>.

Venkatesha, Melally Dinesh, Anegunda, 2012. The coffee white stem borer *Xylotrechus quadripes* (Coleoptera: Cerambycidae): bioecology, status and management. *International Journal of Tropical Insect Science*. 32. 10.1017/S1742758412000331.

Ziska, Lewis Bradley, Bethany D. Wallace, Rebekah Barger, Charles H. LaForest, Joseph Choudhury, Robin A. Garrett, Karen Vega, Fernando, 2018. Climate Change, Carbon Dioxide, and Pest Biology, Managing the Future: Coffee as a Case Study. <https://doi.org/10.3390/agronomy8080152>.