International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Detection of In-Perceptible Fruits Bearing on Trees from Inter-Spaced Images

Chitra Bhole[1], Chandani Joshi[2] ,Mukesh Kalla[3,] Kamal Hiren[4] Anand Darpan[5]

Submitted: 25/04/2023 **Revised**: 25/06/2023 **Accepted**: 03/07/2023

Abstract: The detection of fruits is a mainstream application taken into account for Single Shot Detectors as well as Region Based Detectors. The problem can be taken on other dimension by making a shift inclined towards the real world application where the detection of fruits when bearing on the trees are done. The dataset for such a problem needs to be gathered in a custom fashion, which we were able to do successfully. Apart from this the annotations for data were also made and famous state of-the-art Object Detection models were used for performance mapping using YOLO.v3, YOLO.v4 and Mask-RCNN. These models are fine-tuned to meet the needs of the dataset and metric of mean average precision comparison is shown. The loss metrics for these models in form of their comparisons is also provided in the results section. The result section also highlights a detailed prediction yielded by the different models on different testing based images respectively

Keywords: *Fine-Tuning; Fruit Detection, Mask-RCNN Object Detection, Region Based Detectors, Single Shot Detectors, YOLO*

1. Introduction

The advent of Deep Learning [1], [2] significantly overtook Machine Learning [3] in a span of years seamlessly as the need for data arose. The applications of deep learning started emerge for computer vision [4], natural language processing [5], time series [6] and audio [7] based data. The prognostication aspect of the deep learning was earlier covered by machine learning by using diversified set of algorithms, viz. logistic regression [8], support vector machines [9], k-nearest neighbors [10], decision trees [11], bagging ensemble [12] methods, boosting ensemble [13] methods, stacked ensemble [14] methods, discriminant [15] learners, clustering [16], regression [17] based methods, etcetera. This was significantly overtaken with deep learningthat uses artificial neural networks inspired from multi-layer perceptron [18]. The process of the deep learning has forward and backward [19] propagation for learning. The neural network learns features from the input by calculating an arbitrarily defined weight and bias which modulates the necessary information learned using an activation function. These activation functions are mathematical function such as softmax [20], ReLU [21] and many more. The entire process learns features and loss is computed which is later optimized using a function such as Adam [22], AdaGrad [23], et cetera. This process extended to areas of computer vision where it all started with convolutional neural network [24]–[26], used for handling the data in form of images.The network uses convolutional layers for learning features and these are reduced using pooling layers with rest of the process as same as regular neural net [27]. This form of network was a breakthrough and was applied in all the areas of computer vision as a baseline network, such as image classification [28], object detection [29], image segmentation [30]–[32], face recognition [33] and image captioning [34] that although uses the essence of recurrent neural networks [35] which are applied in the applied areas of natural language processing. Considering the NLP which is an abbreviation for natural language processing the amount of applications are numerous where the extraction of the features for textual data is done using it and predic- tive analytics is handled using machine learning and deep learning respectively. The use of feature extraction which are known as learning representations from the text initially was performed using bag of words [36] or TF-IDF [37] which significantly changed to word2vec [38] and glove [39] in latter part. The prognostication mechanism for it started with RNN based networks which later used LSTM [40] followed by bidirectional [41] networks too. Parallel to the time period of all these networks was developed one dimensional CNN [42] that was uses. Later many combinations [43] of these parts were made in development perspectives which had some major loopholes that were covered by transformer [44], [45] based architectures. Many advancements in the area of transformers were also done which is completely a different topic but the gist of this paper lies in the object detection aspect.

The object detection methods were prominently used high number of advancement with two approaches that are still used for defining the differences where the two major categories are region based detectors and single shot detectors. The region based algorithms also use the term of selective search and map the object for detection using

1 K. J. Somaiya Institute of Technology, Mumbai , Reseacrh Scholar at Sir Padampat Singhania University,Udaipur, India
2 ,3 ,5 Sir Padampat Singhania University, Udaipur,India
4 Symbiosis University of Applied Science ,Indore, India
** Corresponding Author Email: cbhole@somaiya.edu*

region based boundaries. The region of interest [46] is the main associated term with this type of networks. The uprising of such networks started back in 2014 with R-CNN [47] first. Followed by it in 2015 more advancements were brought in the concept and SPP-Net [48] was discovered. Multi tasking was later introduced in 2015 with FRCN [49], followed by which Faster R-CNN [50] came in same year. The concepts used earlier were taken on a next level in later period when R-FCN [51], FPN [52] and Mask R- CNN [53] were introduced. In the meantime the single shot detectors were also taking the form of advancements where in 2014, multi-box [54] was introduced followed by in 2015 came the Attention-Net [55] and one year later G- CNN [56]. These were all responsible for the development of YOLO [57] in 2016 and followed by it was SSD [58] introduced. Later DSSD [59], YOLOv2 [60] and DSOD [61] were introduced that changed the way people think about single shot detectors. The YOLO was later considered for way ahead of time advancements with YOLO.v3 [62] and YOLO.v4 [63] respectively. This is where we consider only the major aspects of the implementation and the algorithms used in further sections of the paper.

2. Methodology

A. DATASET

The implementation section of the paper has data as its most important part as the data used for mainstream detection of fruits is not used. Many researchers have been emphasizing on the fruits detection [63]–[65] with not base consideration of the fruits related data but a very generalized data based on PASCAL VOC [66] or COCO [67] using a framework like TensorFlow [68] respectively. The data such as ImageNet [69] is very well assigned for the image classification based tasks and not object detection since detection requires annotations. Our task was the collection of data for the fruits that specifically grown in the tropical sections of the world. The fruits that we considered are Mango, Sapodilla, Tomato and Sweet Lime respectively. Our main task was its collection that also considers that the fruits are bearing on the trees. Since most of the stock style images found on the internet do not have any proper background.

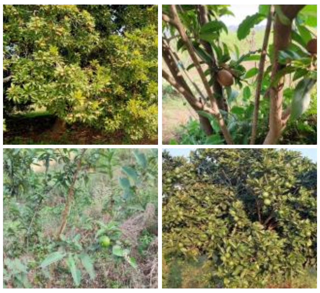

Fig 1. Custom Made Data

The figure 1 depicts some of the fruits that we have clicked their images personally where all the 4 categories are covered. Now the task that arises is pertaining to its annotations. The annotations are most necessary aspect of object detection algorithm as the coordinates of the all the objects that will be trained are considered in an image. The implementation approach that we are considering has use of YOLO.v3, YOLO.v4 and Mask R-CNN algorithms where the annotation procedure for YOLO family and Mask RCNN differ from each other.

B. YOLO.V3

YOLO.v3 [61] is the advanced variant of the YOLO [56] series that came into existence quite later than expected. The scores in the prediction mechanism for earlier versions use softmax [19] that provides the probability of the all the classes and highest probability is the correct detection. In order to overcome this issue of multi-label classification the YOLO.v3 was introduced back in 2018 that works with independent logistic classifier for evaluation of the likelihood. The binary cross-entropy loss function is used in the computation instead of the mean squared error used in earlier versions. The FPN [51], abbreviation of the Feature Pyramid Networks has the similar prediction mechanism and YOLO.v3 was inspired from it. The older variants of the YOLO used Darknet-19, which is a 19 layered network for feature extraction from the annotated images. This variant of YOLO uses a 53 layers deep network known as Darknet53 where the concept of skip connections was used. This concept was introduced in the ResNet [70], where residual blocks [26] are used. Billion Floating Point Operations are lesser in Darknet-53 as compared to Darknet-152, but manages to give 2x faster classification accuracy. Considering the 3x faster computation the YOLO.v3 is able to give similar performance with COCO [67] dataset for average precision metric. The detection of the smaller objects was more emphasized in this variant effectively.

C. YOLO.V4

The more advance variant of YOLO.v3 [61] can be considered as YOLO.v4 [62] which approximately gives 65 frames per second inference speed when used with V100 Tesla GPU. Just like ResNet [70] the concept of skip connections can be also driven from DenseNet [71] which is used in this variant. The DenseNet uses batch normalization for every single layer along with ReLU [20] activation function. The DenseNet feature maps need divisions which can be done using the cross stage partial connections where the task of segregation is conducted. This cross stage partial connections abbreviated as CPS connections is merged with the Darknet-53 for a great reduction in the computational complexity which results in the CPS-Darknet-53 that easily takes over the performance of the ResNet used as a feature extractor. Mosiac Data Augmentation [72] is used as one of the techniques that combines 4 images into 1 for training which improves the object detection beyond the scope of the normal range. It also uses Dropblock [73] regularization which is different from Dropout [74] regularization, that forces the model to drop the information from image intentionally to learn more efficiently from available information. An activation function much more efficient than ReLU known as Mish [75] activation is used which can outperform the performance over many different datasets.

D. MASK R-CNN

The faster R-CNN [49] was developed earlier as compared to the Mask R-CNN [52] which was way good for a stateof-the-art algorithm. All the region based object detection algorithms used earlier emphasized on the region proposal mechanism. This is where the change was made with the advent use of image segmentation that became prevalent in a very short period of time. The region of interest align was the change brought into the mainstream region of interest pooling methods. The interpolation can be leveraged for generating the internal feature maps. The region proposals are visualized with dots instead of lines in this algorithm which does provide an edge over the other over-lapping images solving the problem. The mask uses the image segmentation [29]–[31] process very efficiently. The image segmentation although has types such as instance and semantic segmentation we are only concerned working with the segmentation used in Mask R-CNN. It uses instance segmentation and it is responsible for segmenting the same objects from one class as completely different. This gives the advantage of selecting very specific person or object from the prediction even though many similar are present simultaneously. The memory efficiency is much higher and usually requires less GPU memory while training. The implementation performed by Facebook research team [76] provides the benchmarks for COCO [67] evaluation stating the algorithm is very efficient as compared to other region based variants.

3.Results

A. SINGLE SHOT RESULTS

The single shot detector contains YOLO.v3 and YOLO.v4 in our implementation where have the results with input images and output images. The fruits have been extensively trained on 4 types of fruits where the prediction for each of them is given in the figures.

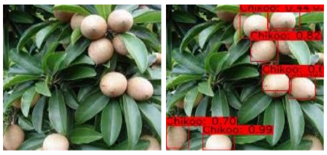

Fig 2. Sapodilla Prediction using YOLO.v3

The figure 2 gives the input image and predicted image of sapodilla where the prediction is made using yolo.v3 with bounding boxes where the fruits are detected on the tree. Every bounding box has also confidence probabilities associated with every fruit. Similarly the prediction can be seen for Mangoes in the figure 3

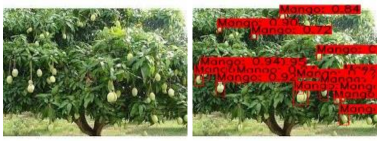

Fig 3. Mangoes on Tree Prediction using YOLO.v3

As figure 3 depicts the mangoes on tree have been predicted very effectively. The confidence probabilities are very good for every bounding box even from a very long distance image. The identification is done irrespective of different training data effectively. The detection for the fruit itself can be performed and it does seamlessly as the data has been trained on the fruit itself with less exposure of tree altogether. The prediction can be seen in the figure 4 respectively.

The similar mechanism can be extended for other fruits too, for instance the Sweet Lime. Now the Sweet Lime and not ripe mango are close enough in context of the color yet the algorithm detects it with no issue.

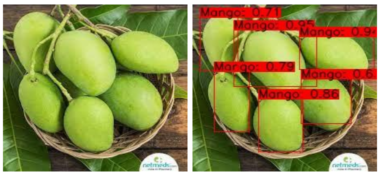

Fig 4. Mangoes in Close Interspace Prediction using YOLO.v3

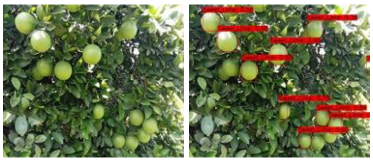

Fig 5. Sweet Lime Prediction using YOLO.v3

The Sweet Lime does in same predictions gets confused with the leaves of the tree. This can be resolved in other models. The Sweet Lime prediction is subtle and now tomato prediction can also be made.

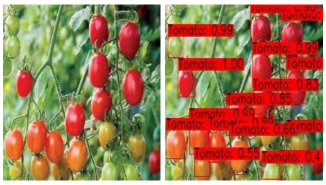

Fig 6. Tomato Prediction using YOLO.v3

The great part about tomato prediction is that even when the tomatoes are not ripe, the algorithm does not give wrong prediction. This is an indication the cause of color of image has less accountability as opposed to the shape of the fruit.

The shape of the fruit is necessary and holds more precedence over the color. Similar context can be extended for predictions related to YOLO.v4 with input and predicted images.

The best results can be obtained when the algorithm is harnessed the most and in the terms of prediction

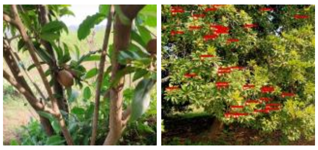

Fig 7. Sapodilla Prediction using YOLO.v4

sapodilla tree the fruits are extremely small and the interspace is very high between the person clicking the image and yet the prediction yielded is very good. The algorithm has detected most of the fruits bearing on the tree.

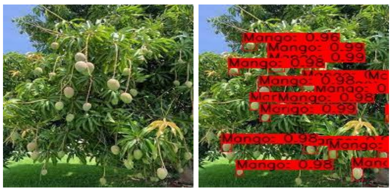

Fig 8. Mango Prediction using YOLO.v4

The image of the mangoes this time for prediction is taken from a medium level interspace that is not too harsh nor too soft on the algorithm where the prediction can be seen at a very good level. The confidence probabilities are very good and most of the fruits bearing on the tree are being detected.

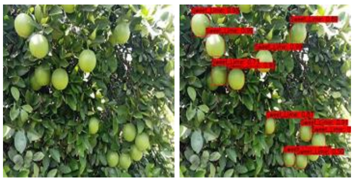

Fig 9. Sweet Lime Prediction using YOLO.v4

The images used for Sweet Lime prediction are as same as the YOLO.v3 where the algorithm gave some mispredictions and this time those errors are solved. The predictions are performed appropriately on the fruit without predicting the leaves as the fruit. This same courtesy can be extended for detecting the Sweet Lime from a very close interspace for prediction of the fruit.

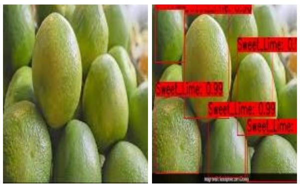

Fig 10. Sweet Lime in close interspace Prediction using YOLO.v4

The figure 10 depicts the fruit in very close interspace and still the prediction is very good to precise. Similarly the prediction for next image can be also done.

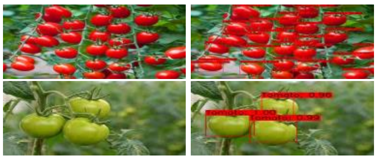

Fig 11. Tomato Prediction using YOLO.v4

The advancements for the algorithm can be seen in detection of the images as nothing is missed at all. The tomatoes in 2 states, rip and unripe make this prediction interesting but both predictions are way good than expected and turn out to be perfect. The images on left are input images of tomatoes on vine and bushes and the predictions suffice more than ever.

B. REGION BASED RESULTS

This section of the Results contains the region based algorithm, Mask R-CNN for implementation where the masks are generated along the training. The data consists of fruits where we train our custom dataset and the fruits detected contain the region boundary which is dotted and masks for every fruit are generated. This is considerably an advanced form of detection system we have implemented. The predictions can seen for every single fruit in this respective sub-section of the paper.

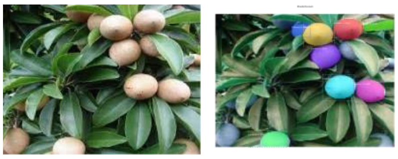

Fig 12. Sapodilla in close interspace Prediction using Mask-RCNN

The predictions yielded are very distinct and accurate for mask R-CNN with dotted region of interest

line and masks with their respective confidence probabilities. The figure 12 gives the prediction for

sapodilla in close interspace where the prediction from a very long distance can also be generated.

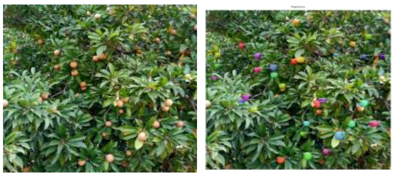

Fig 13. Sapodilla Prediction using Mask-RCNN

The prediction for sapodilla from a long distance is done in the 13 and it is very accurate. Similarly other predictions can also be performed.

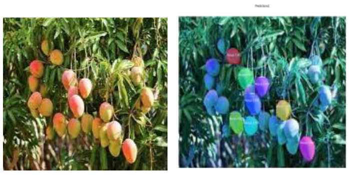

Fig 14. Mango Prediction using Mask-RCNN

The mask based prediction is performed by segmentation but the detection limitations can be seen in some cases where the YOLO.v4 considers itself to be an advanced model. The prediction can be similarly seen for Sweet Lime below.

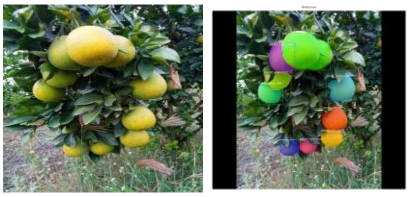

Fig 15. Sweet Lime Prediction using Mask-RCNN

The masks are generated for the Sweet Limes and the predictions are extremely accurate. The model outperforms in terms of detection taking segmentation into consideration

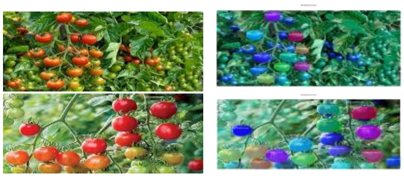

Fig 16. Tomato Prediction using Mask-RCNN

The tomatoes are predicted using the segmentation at a very good level giving extremely distinct predictions to almost every single tomato.

C. RECORDED LOSS

The recorded loss is observation of the loss generated while the training feature extractor for the networks. The feature extractor for YOLO.v3 and YOLO.v4 is darknet and for Mask R-CNN it is Resnet respectively. Considering the training process, the training for every single fruit is done differently where the generated loss is considerably can be depicted in the form of table.

Similarly the loss can be recorded for the Yolo.v4 where the loss is higher than the Yolo.v3 yet the amount of predictions are very good which gives inference that loss is not really everything.

Table 1. Recorded Loss for Yolo.v3

Fruit	Loss
Mango	0.869378
Sapodilla	1.208245
Tomato	0.787936
Sweet Lime	0.633085

Table 2. Recorded Loss for Yolo.v4

Fruit	Loss
Mango	1.092077
Sapodilla	1.509499
Tomato	0.596656
Sweet Lime	1.830130

Finally the loss observations can be done for the Mask R-CNN like the single shot detectors and it is the lowest among all. The network used is Resnet which is a good feature extractor.

Table 3. Recorded Loss for Mask R-CNN

Fruit	Loss
Mango	0.0777
Sapodilla	0.2390
Tomato	0.0987
Sweet Lime	0.0900

4.Conclusion

The mainstream idea of detecting fruits has became prevalent in deep learning research and our idea was an extension to it where the fruits will be bearing from the tree. The idea after taking into consideration, biggest challenge was the collection of the data that is done by us itself. Since the detection is involved the task of data annotation was forthcoming and we performed it manually. Later decided to work with the fine-tuning of state-of-the-art object detection models for best predictions. The models used were from both the sub-divisions, viz. single shot detectors as well as region based detectors. Both gave good object detection prediction making this idea successful. This paper will for sure serve as a reference material for many researchers that promotes the thought of mainstream ideas to fit the real world inclination more than ever.

Conflicts of interest

The authors declare no conflicts of interest.

Data Availability

The nature of the research quantifies the participants of this experimentation for not agreeing the public usage of the data and its availability.

References

[1] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” nature, vol. 521, no.7553, pp. 436–444, 2015.

[2] H. Sarker, “Machine learning: Algorithms, real-world applications and research directions,” SN Computer Science, vol. 2, no. 3, pp. 1–21, 2021.

[3] V. Wiley and T. Lucas, “Computer vision and image processing: A paper review,” International Journal of Artificial Intelligence Research, 2018.

[4] Y. A. Solangi, Z. A. Solangi, S. Aarain, A. Abro, G. A. Mallah, and A. Shah, “Review on natural language processing (nlp) and its toolkits for opinion mining and sentiment analysis,” in 2018 IEEE 5th International Conference on Engineering Technologies and Applied Sciences (ICETAS), 2018, pp. 1–4.

[5] F. Dama and C. Sinoquet, “Time series analysis and modeling to forecast: a survey,” arXiv preprint arXiv:2104.00164, 2021.

[6] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S.-Y. Chang, and T. Sainath, “Deep learning for audio signal processing,” IEEE Journal of Selected Topics in Signal Processing, vol. 13, no. 2, pp. 206–219, 2019.

[7] J. S. Cramer, “The origins of logistic regression,” Econometrics eJournal, 2002.

[8] M. Hearst, S. Dumais, E. Osuna, J. Platt, and B. Scholkopf, “Support vector machines,” IEEE Intelligent Systems and their Applications, vol. 13, no. 4, pp. 18–28, 1998.

[9] P. Cunningham and S. J. Delany, “K-nearest neighbour classifiers-a tutorial,” ACM Computing Surveys (CSUR), vol. 54, no. 6, pp. 1–25, 2021.

[10] J. R. Quinlan, “Induction of decision trees,” Machine learning, vol. 1, no. 1, pp. 81–106, 1986.

[11] N. Kanvinde, A. Gupta, and R. Joshi, “Binary classification for high dimensional data using supervised non-parametric ensemble method,” arXiv preprint arXiv:2202.07779, 2022.

[12] M. Gupta, S. S. Shetty, R. M. Joshi, and R. M. Laban, “Succinct differentiation of disparate boosting ensemble learning methods for prognostication of polycystic ovary syndrome diagnosis,” in 2021 International Conference on Advances in Computing, Communication, and Control (ICAC3). IEEE, 2021, pp. 1–5.

[13] S. Nair, A. Gupta, R. Joshi, and V. Chitre, “Combining varied learners for binary classification using stacked generalization,” arXiv preprint arXiv:2202.08910, 2022.

[14] Gupta, H. Soni, R. Joshi, and R. M. Laban, “Discriminant analysis in contrasting dimensions for polycystic ovary syndrome prognostication,”arXiv preprint arXiv:2201.03029, 2022.

[15] D. Xu and Y. Tian, “A comprehensive survey of clustering algorithms,” Annals of Data Science, vol. 2, no. 2, pp. 165–193, 2015.

[16] D. Maulud and A. M. Abdulazeez, “A review on linear regression comprehensive in machine learning,” Journal of Applied Science and Technology Trends, vol. 1, no. 4, pp. 140–147, Dec. 2020. [Online].

[17] J. Singh and R. Banerjee, “A study on single and multi-layer perceptron neural network,” in 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), 2019, pp. 35–40.

[18] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning internal representations by error propagation,” California Univ San Diego La Jolla Inst for Cognitive Science, Tech. Rep., 1985.

[19] J. S. Bridle, “Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters,” in Proceedings of the 2nd International Conference on Neural Information Processing Systems, ser. NIPS’89. Cambridge, MA, USA: MIT Press, 1989, p. 211–217.

[20] F. Agarap, “Deep learning using rectified linear units (relu),” arXiv preprint arXiv:1803.08375, 2018.

[21] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv preprint arXiv:1412.6980, 2014.

[22] J. Duchi, E. Hazan, and Y. Singer, “Adaptive subgradient methods for online learning and stochastic optimization,” Journal of Machine Learning Research, vol. 12, no. 61, pp. 2121–2159,2011.

[23] Y. LeCun, Y. Bengio et al., “Convolutional networks for images, speech, and time series.”

[24] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. AlShamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, “Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions,” Journal of big Data, vol. 8, no. 1, pp. 1–74, 2021.

[25] Gupta, R. Joshi, and R. Laban, “Detection of tool based edited images from error level analysis and convolutional neural network,” arXiv preprint arXiv:2204.09075, 2022.

[26] Gupta, S. Nair, R. Joshi, and V. Chitre, “Residual-concatenate neural network with deep regularization layers for binary classification,” in 2022

[27] 6th International Conference on Intelligent Computing and Control Systems (ICICCS). IEEE, 2022, pp. 1018–1022.

[28] Li, J. Feng, L. Hu, J. Li, and H. Ma, “Review of image classification method based on deep transfer learning,” in 2020 16th International Conference on Computational Intelligence and Security (CIS), 2020, pp. 104–108.

[29] Z.-Q. Zhao, P. Zheng, S.-t. Xu, and X. Wu, “Object detection with deep learning: A review,” IEEE

transactions on neural networks and learning systems, vol. 30, no. 11, pp. 3212–3232, 2019.

[30] N. R. Pal and S. K. Pal, "A review on image segmentation techniques," Pattern Recognition, vol. 26, no. 9, pp. 1277–1294,1993.

[31] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," IEEE transactions on pattern analysis and machine intelligence, 2021.

[32] R. M. Joshi and D. Shah, "Refactoring faces under bounding box using instance segmentation algorithms in deep learning for replacement of editing tools," in Intelligent Computing and Networking. Springer, 2022, pp. 236–247.

[33] M. Wang and W. Deng, "Deep face recognition: A survey," Neurocomputing, vol. 429, pp. 215–244, 2021.

[34] Elhagry and K. Kadaoui, "A thorough review on recent deep learning methodologies for image captioning," arXiv preprint arXiv:2107.13114, 2021.

[35] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," arXiv preprint arXiv:1406.1078, 2014.

[36] Y. Zhang, R. Jin, and Z.-H. Zhou, "Understanding bag-of-words model: a statistical framework," International journal of machine learning and cybernetics, vol. 1, no. 1, pp. 43–52, 2010.

[37] S. Tambe, R. Joshi, A. Gupta, N. Kanvinde, and V. Chitre, "Effects of parametric and non-parametric methods on high dimensional sparse matrix representations," arXiv preprint arXiv:2202.02894, 2022.

[38] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," arXiv preprint arXiv:1301.3781, 2013.

[39] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1532–1543.

[40] R. C. Staudemeyer and E. R. Morris, "Understanding lstm – a tutorial into long short-term memory recurrent neural networks,"2019.

[41] M. Schuster and K. Paliwal, "Bidirectional recurrent neural networks," IEEE Transactions on Signal Processing, vol. 45, no. 11, pp. 2673–2681, 1997.

[42] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," Advances in neural information processing systems, vol. 28, 2015.

[43] R. Joshi, A. Gupta, and N. Kanvinde, "Res-cnn-bilstm network for overcoming mental health disturbances caused due to cyberbullying through social media," arXiv preprint arXiv:2204.09738, 2022.

[44] Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," Advances in neural information processing systems, vol. 30, 2017.

[45] R. Joshi and A. Gupta, "Performance comparison of simple transformer and res-cnn-bilstm for cyberbullying classification," arXiv preprint arXiv:2206.02206, 2022.

[46] H. Lin, J. Si, and G. P. Abousleman, "Region-of-interest detection and its application to image segmentation and compression," in 2007 International Conference on Integration of Knowledge Intensive Multi-Agent Systems, 2007, pp. 306–311.

[47] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580–587.

[48] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," IEEE transactions on pattern analysis and machine intelligence, vol. 37, no. 9, pp. 1904–1916, 2015.

[49] R. Girshick, "Fast r-cnn," in Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440–1448.

[50] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards realtime object detection with region proposal networks," Advances in neural information processing systems, vol. 28, 2015.

[51] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," Advances in neural information processing systems, vol. 29, 2016.

[52] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2117–2125.

[53] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in Proceedings of the IEEE international conference on computer vision, 2017, pp. 2961–2969.

[54] Erhan, C. Szegedy, A. Toshev, and D. Anguelov, "Scalable object detection using deep neural networks," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 2147–2154.

[55] Yoo, S. Park, J.-Y. Lee, A. S. Paek, and I. So Kweon, "Attentionnet: Aggregating weak directions for accurate object detection," in Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2659–2667.

[56] M. Najibi, M. Rastegari, and L. S. Davis, "G-cnn: an iterative grid based object detector," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2369–2377.

[57] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779– 788.

[58] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in European conference on computer vision. Springer, 2016, pp. 21–37.

[59] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "Dssd: Deconvolutional single shot detector," arXiv preprint arXiv:1701.06659, 2017.

[60] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 7263–7271.

[61] Z. Shen, Z. Liu, J. Li, Y.-G. Jiang, Y. Chen, and X. Xue, "Dsod: Learning deeply supervised object detectors from scratch," in Proceedings of the IEEE international conference on computer vision, 2017, pp. 1919–1927.

[62] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," arXiv preprint arXiv:1804.02767, 2018.

[63] Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," arXiv preprint arXiv:2004.10934, 2020.

[64] C. Foong, G. K. Meng, and L. L. Tze, "Convolutional neural network based rotten fruit detection using resnet50," in 2021 IEEE 12th Control and System Graduate Research Colloquium (ICSGRC), 2021, pp. 75–80.

[65] T. P. Chung and D. V. Tai, "A fruits recognition system based on a modern deep learning technique," Journal of Physics: Conference Series, vol. 1327, no. 1, p. 012050, oct 2019. [Online]. Available: https://doi.org/10.1088/1742-6596/1327/1/012050

[66] Chaudhari, S. S. More, S. Khane, H. Mane, and P. Kamble, "Object detection using convolutional neural network in the application of supplementary nutrition value of fruits," International Journal of Innovative Technology and Exploring Engineering, 2019.

[67] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," International journal of computer vision, vol. 88, no. 2, pp. 303–338, 2010.

[68] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in European conference on computer vision. Springer, 2014, pp. 740–755.

[69] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org.

[70] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255.

[71] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[72] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4700–4708.

[73] Z. Wei, C. Duan, X. Song, Y. Tian, and H. Wang, "Amrnet: Chips augmentation in aerial images object detection," arXiv preprint arXiv:2009.07168, 2020.

[74] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "Dropblock: A regularization method for convolutional networks," Advances in neural information processing systems, vol. 31, 2018.

[75] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," The journal of machine learning research, vol. 15, no. 1, pp. 1929–1958, 2014.

[76] Misra, "Mish: A self regularized non-monotonic activation function," arXiv preprint arXiv:1908.08681, 2019.

[77] Massa and R. Girshick, "maskrcnn-benchmark: Fast, modular reference implementation of Instance Segmentation and Object Detection algorithms in PyTorch," https://github.com/facebookresearch/maskrcnn-benchmark, 2018, accessed: [13-06-2022].

[78] Banerjee, S. ., & Mondal, A. C. . (2023). An Intelligent Approach to Reducing Plant Disease and Enhancing Productivity Using Machine Learning. International Journal on Recent and Innovation Trends in Computing and Communication, 11(3), 250–262. https://doi.org/10.17762/ijritcc.v11i3.6344

[79] Prof. Sharayu Waghmare. (2012). Vedic Multiplier Implementation for High Speed Factorial Computation. International Journal of New Practices in Management and Engineering, 1(04), 01 - 06. Retrieved from http://ijnpme.org/index.php/IJNPME/article/view/8