Automated object detection of mechanical fasteners using faster region based convolutional neural networks

M. Karthikeyan, T. S. Subashini
Department of Computer Science and Engineering, Annamalai University, Tamil Nadu, India

Article Info

Article history:

Received Dec 15, 2020
Revised May 11, 2021
Accepted Jun 2, 2021

Keywords:

Fasteners
Faster R-CNN
Object detection
RPN

ABSTRACT

Mechanical fasteners are widely used in manufacturing of hardware and mechanical components such as automobiles, turbine & power generation and industries. Object detection method play a vital role to make a smart system for the society. Internet of things (IoT) leads to automation based on sensors and actuators not enough to build the systems due to limitations of sensors. Computer vision is the one which makes IoT too much smarter using deep learning techniques. Object detection is used to detect, recognize and localize the object in an image or a real time video. In industry revolution, robot arm is used to fit the fasteners to the automobile components. This system will helps the robot to detect the object of fasteners such as screw and nails accordingly to fit to the vehicle moved in the assembly line. Faster R-CNN deep learning algorithm is used to train the custom dataset and object detection is used to detect the fasteners. Region based convolutional neural networks (Faster R-CNN) uses a region proposed network (RPN) network to train the model efficiently and also with the help of Region of Interest able to localize the screw and nails objects with a mean average precision of 0.72 percent leads to accuracy of 95 percent object detection.

Corresponding Author:

M. Karthikeyan
Department of Computer Science and Engineering
Annamalai University
Annamalai Nagar, Tamil Nadu, India
Email: karthickrock125@gmail.com

1. INTRODUCTION

AI and machine learning is the current technologies emerging leads to automation of systems. Nowadays, robots are used in automobile industry to assemble a car, these robots are used for multiple operation such as drilling and welding process. A semi-autonomous robot is trained and operates by person to complete the assembly work. This paper proposed the idea about computer vision acts as sensor to detect multiple objects such as type of fasteners to complete the assembly of the vehicle to roll out the manufacturing process to be automated in Industry 4.0. Recently a Deloitte had written an article about Industry 4.0, states that major characteristics are about smart production systems is a vertical networking of smart manufacturing leads to research and development to make smart systems helps to increase productivity of products in an industry. Deep learning techniques involves learning the larger dataset themselves with the help of neural networks and it leads to different types of learning such as supervised learning, unsupervised learning and reinforcement learning. Convolution neural network is used to classify the images and it fails to detect the object in the image. R-CNN is introduced at year 2014 by Ross Girshick which uses selective search algorithm that extracts a region around 2000 and it is called as region proposals. It involves three stages of process such as to find regions in the image that contains an object. Then it will extract cnn features from region proposals. Finally it classifies the object with the extracted features. Fasteners are nuts and bolts used in automobiles.

A robust system to detect tiny metal objects was proposed in [1]. The system is based on hyperspectral imaging technique to analyse the images based on which support vector machine (SVM) is used to classify the objects. The author has made use of hyperspectral camera produced by Wayho Technology to capture the different physical properties of various materials. As a model that could use the full dimensionality of the hyperspectral data captured is required, SVM has been used by avoiding the feature selection process which resulted in a time saving MOD model. Gaussian Kernel with 0.2 for standard deviation and with the value of the penalty parameter as 10 yielded the best results for classification by SVM. Environment for visualizing images (ENVI) software is used for labelling the training images. The coding for the development of the classification system was done using MATLAB. The author was able to achieve an accuracy of 93.4% for ferromagnetic metals and 94.2% for nonferromagnetic metal objects. The present deep-neural network-based object detection models can be divided into 2 categories: Region-based [2], [3] and the region free [4] object detection methods. Wu *et al.* [5] describes that depth information in clear shapes to detect the boundary of objects. An end-to-end multiscale multilevel context and multimodal fusion network is used to learn object identification from RGB value and object boundaries from the data. The method aggregates multiscale multilevel context feature maps which has improved the accuracy on a larger scale. Supervised learning over different modes and scales is applied with a learning rate of 1e-4 for Adam optimization for optimizing MCMFNet. The model is evaluated on NJU2K [6], DES [7], NPLR [8] and SSD [9] datasets. The proposed MCMFNet is also compared with seven RGB-D SOD models: DCMC [10], and CDCP [11], and it was found that the model outperforms others at 88.9% of S-measure.

A new face detection scheme was presented in [12] using faster regional convolution neural network (R-CNN) by combining feature concatenation, hard negative mining, multi-scale training and model pre-training. The tuning of key parameters is also done in addition to this. The lower-level and high-level features are combined using the intermediate results along with the final feature map in region proposed network (RPN). Features from multiple lower-level convolution layers are then RoI-pooled and L2-normalized. The IOU value is set as 0.5 for the hard negative threshold and ratio of the foreground and background is considered to be 1:3. The network is iteratively trained with the hard negatives to improve the efficiency of the model. The model is pre-trained on WIDER FACE dataset and the fine-tuned on FDDB dataset. A pre-trained VGG16 model is trained on these datasets for 110,000 iterations with the learning rate set to 0.0001. Apparently, the model out-performed all the other state- of-art methods and addressing the efficiency and scalability of the proposed method for real-time face detection is been stated as the enhancement of the work. In [13] a deep learning-based approach for bone age assessment is done by faster R-CNN and Inception-v4 networks. It combines the expert knowledge from TW3 methods and feature engineering for detection. Secondly, in the feature extraction process, the features of the input image are extracted to propose the region of interest (ROI). This is done by the Resnet-101 architecture in which the RPN proposes various regions to form a small feature map of fixed size. The final classifier layer detects six kinds of ROIs in the images: dp3, mp3, pp3, radius, ulna, and mc1. These ROIs are then cropped and resized for training the classification network. The network is trained and evaluated on digital hand Atlas (DHA) dataset.

Certain augmentation techniques like random rotation of image between -15 to 15 degrees, zoom in, and flipping, are carried out during the training process to increase the size of the dataset. The mAP achieved for maturity stage classification using Resnet-101 was 81.5% which is greater than the 68% accuracy reported in another study [14]. A model was designed in [15] to perform threat object detection using faster R-CNN and YOLO. The performance is studied on 4 classes of threat objects: i) Gun, ii) Shuriken, iii) Razor-blade, iv) Knife. In addition to GDXray database, images of x-ray image modelling and image transformation methods are used to create the customized dataset. Further, certain augmentations like flip, random distortion, rotation, skew and zoom are also done to increase the size of the dataset [16]. Faster RCNN was evaluated on various architectures like with AlexNet, VGG-16, and ResNet-50,101 architectures. It's observed that faster R-CNN with RESNET at a global learning rate of 0.001, momentum of 0.9 and parameter decay of 0.0005 yields a detection accuracy of 98.4% in terms of positive threat object detections. Apparently, the threat recognition requires only 0.16 sec per image. Small object detection is a challenging problem in computer vision, it suggested that with the help of pascal voc for small object detection task [17]. Faster R-CNN takes feature map for the input image and extracted region features on the feature map that takes shared regions among the different regions [18]. Bshapenet was implemented with one of the small object detection as fasterR-CNN [19]. Real time leak detectetion for automated hydro carbon in industry was proposed with faster R-CNN and the model get compared with SSD [20]. Detection of vulnearable objects in autonomous driving of vehicle with DNN [21]. VGG16 based faster R-CNN was proposed for the detection of vehicles in complex traffic [22]. Overlapped fruit detection for optimized mask R-CNN application in apple harvest is implemented [23]. Facial landmark detection was made through facial patches along with

CNN [24]. Faster R-CNN algorithm is used to detect the tender coconut in the tree and proposed that it is better than SSD and YOLO [25].

2. RESEARCH METHOD

2.1. Proposed object detection method for fasteners

The Architecture diagram for the proposed methodology was illustrated in Figure 1. Schematic diagram of the proposed object detection of fasteners.

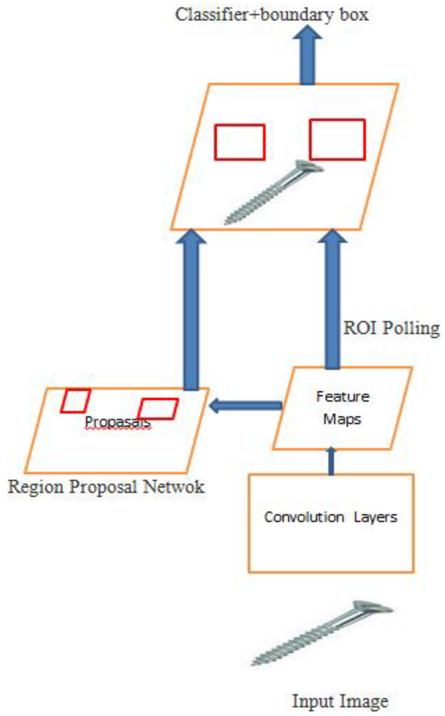

Figure 1. Schematic diagram of the proposed object detection of fasteners

2.2. Faster-RCNN

The first step for object detection in convolution neural network to extract the feature maps with help of relu and pooling layers. This feature maps is shared among all the RPN layers and fully connected layers. In RESNET architecture had conv1 layer as 3x3 with a zero padding,the number of filter 3x3 convolutions in each stage is the same and transitions between stages reduced by the spatial dimensions in a scale of two by double the number of channels. The first convolution layer of resnet-101 architecture uses 7x7 filters with stride 2, downsampling between the stages is done with stride 2 instead of max pooling layers and the final layer is an overall average of pooling layer with connected softmax. Batch normalization is used after each convolution layer. ROI align is applied to these filtered regions of interest that are classified and regressed by a head network. Faster R-CNN is used backbone of Resnet 101 already loaded with predefined weights with the help of pytorch repository.

2.3. RPN

RPN means region proposal network, it is added to the last layer of convolution neural network. It is especially trained with ROI polling along with classifier and with a boundary box regressor to choose the object in the region along with feature maps [2]. RPN place a small slide window on the feature map and it able to build a small network to classify object or not along with regressing boundary box locations. Regression box provides finer localization information with reference to the sliding window. These features are scored with a fully connected layer to define how likely it is that a bounding box of specific size (an anchor) placed at that position fit well an object, for each of the k pre-defined anchors. A sibling layer for regression is also used for each of the anchors. These extra layers are implemented with a fully convolutional network using a 3×3 convolution followed by ReLU and two sibling 1×1 convolutions for classification and regression. RPN architecture is explained in Figure 2. Faster R-CNN architecture diagram.

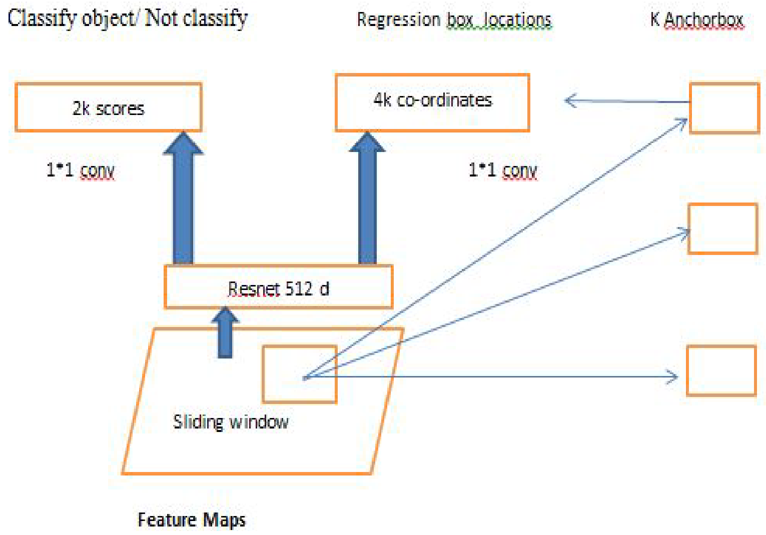

Figure 2. Faster R-CNN architecture diagram

2.4. Object recognition

Object recognition involves identifying objects in images. It involves image classification, object localization, object detection. Image classification method to find the object is belong to bolt class or screw class with the help of faster-RCNN classifier [16]. Then, next step is object localization is used to locate the fasteners object in the images based on the feature maps which is already gets trained with the help of RPN network. Object detection of screw or bolt class gets labelled based on classification and localization of the object.

2.5. Mathematical representation

The dataset is divided in the ration of 80:20 where 80% of the total 800 images are used for training the network and the remaining 200 images is used for the testing purpose. The network is trained with W weights for the I images which is concatenated into a single vector. It is represented as (1).

$$w = (w1;: : : ;wn) \quad (1)$$

Initially, the network is trained with the images from the initial layers to the final layer of the network and based on the predictions of the final layer, the weights of these layers are updated during the back propagation process. The ratio of the change in the output is directly proportional to a small change in the inputs of the network which is represented as (2),

$$\frac{L(x+\alpha)-L(x)}{(x+\alpha)-x} = \frac{L(x+\alpha)-L(x)}{\alpha} \quad (2)$$

where α is the learning rate and x is the value for the input image I. The filter dimensions and the stride values also influence the outputs whose dimensions are finally calculated as (3), (4), (5),

$$W_2 = \frac{W_1-F+2P}{S} + 1 \quad (3)$$

$$H_2 = \frac{H_1-F+2P}{S} + 1 \quad (4)$$

$$D_2 = K \quad (5)$$

where W, H and D represents the width, height and depth of the input whereas K, F, S and P represents the number of filters, width of the filter, the stride and the padding. The network is further optimized for reducing the loss function using stochastic gradient descent technique where the gradient of the cost function of a single image at each iteration is calculated by (6),

$$\delta_b = \delta_b - \propto (Y^{i\prime} - Y^i)X_b^i \quad (6)$$

where $\boldsymbol{\delta}$ is the momentum for the image I in the range m.

The performance of the system was calculated using evaluation metrics are represented in Table 1 and Figure 3. Representation of mean average precision are shown in Figure 4.

Table 1. Computation time for bolt and screw object

S.No	Images	Computation time	Map
1	Bolts	0.14	0.721
2	Screws	0.13	0.712

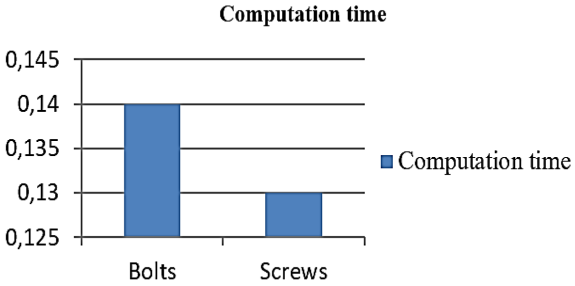

Figure 3. Representation of computation time for bolt and screw object detection

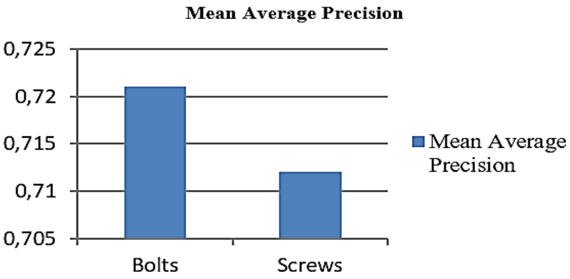

Figure 4. Representation of mean average precision for bolt and screw object detection

3. RESULTS AND DISCUSSION

3.1. Experimental setup

A collection of custom dataset of 1200 images for each bolt and screw is created in which 80 percentages of images is used for training and 20 percentage of images is used for testing. Nearly 120 epochs is used for training with the help of Nvidia GeForce 930-M GPU, faster R-CNN is used much speed execution time for training compared to R-CNN. So, time consuming for training is less compare to R-CNN. An inference on the model is created with the help of ms-coco dataset. Each image annotations is made and it is stored as ms-coco fastener dataset. This labelled ms-coco fastener dataset. The model gets trained is used to test the images will result in detection of the object as screw is shown in below Figure 5. Object detection of screw using faster-RCNN and for the bolt object detection is shown in Figure 6. Object detection of bolt using faster-RCNN.

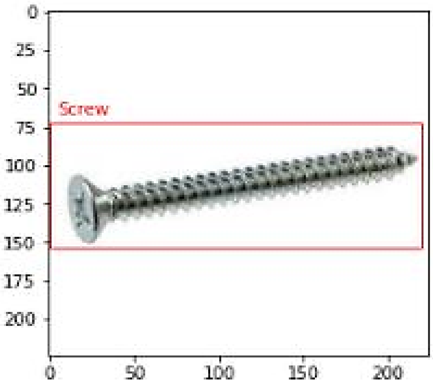

Figure 5. Object detection of screw using faster-RCNN

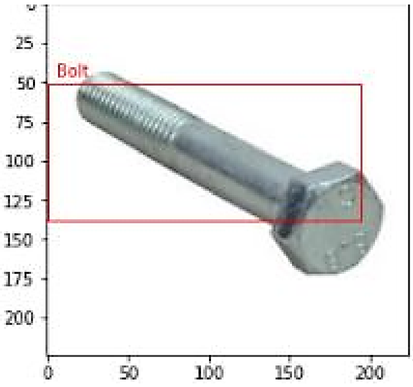

Figure 6. Object detection of bolt using faster R-CNN

4. PERFORMANCE EVALUATION OF PROPOSED METHOD

It is very important to assess the performance of the object detection methods on standard metrics. The performance of the faster R-CNN on the dataset is evaluated on the following metrics:

4.1. Accuracy

It is the ratio of rightly classified objects and the total image under the study. The expression for accuracy is given in (7). The detection accuracy of faster-RCNN on the experimental dataset is found to be 95%.

$$\text{Detection Accuracy} = \frac{\text{True Positive} + \text{False Positive}}{\text{Total of all postive and negative classes}} \tag{7}$$

4.2. Precision

This measure is concerned with assessing the performance based on the true positives among the c lasses of true positives and false positives as given in (8). The work exhibits a promising precision of 95.5%.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \tag{8}$$

4.3. Recall

This score quantifies the ratio of true positive images against the false negatives and true positives. The expression is given in (9). The recall value of the faster-RCNN on the given dataset is found to be 95.5%.

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \tag{9}$$

4.4. F1-Score

F1-Score is an important classification metrics in object detection as it is the geometric mean of precision and recall. This score actually symbolises the balance between the precision and recall. The formulation of F1-score if given in (10). The work portrays an F1-Score of 97.5%.

$$\text{F1} - \text{Score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{10}$$

4.5. Mean average precision (MAP)

In the context of object detection, where there may be multiple classes labelling unique objects in the image, it is crucial to find the average precision of all the classes. MAP is a score that smoothens the average precision among multiple classes and is given in (11). The MAP value of the work is round 72%.

$$\text{Mean Average Precision} = \frac{1}{N}\sum_{i=1}^{N} \text{Average Precision}_i \tag{11}$$

4.6. Specificity

It is the measure of the model’s capability to rightly classify the negative examples as negative. The expression is shown in (12). The faster-RCNN displays the specificity of 50%, which is highly competitive.

$$\text{Specificity} = \frac{\text{True Negative}}{\text{True Negative} + \text{False Positive}} \tag{12}$$

4.7. Sensitivity

This is the ratio of rightly classified examples among the other pure classes and the formulation is given in (13). The faster-RCNN model shows the sensitivity of 99.8%.

$$\text{Sensitivit y} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \tag{13}$$

5. CONCLUSION

In this faster R-CNN fastener object gets detected that is used in manufacturing of automobile industry that help robots to automatically detect the fasteners to pick the fasteners soon to increase the productivity leads to impact the economy in industry revolution. Faster-RCNN is implemented with resnet-101 model achieved with a mean average precision of 0.72 and computation time is less compared to other deep learning model. In future object detection of images with non-max suppression algorithm can be modified according to detect object with angle orientation of fasteners.

REFERENCES

[1] Y. Tian *et al*., "Metal object detection for electric vehicle inductive power transfer systems based on hyperspectral imaging," *Measurement*, vol. 168, 2021, doi: 10.1016/j.measurement.2020.108493.

[2] P. Arbelaez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik, "Multiscale Combinatorial Grouping," *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 328–335, doi: 10.1109/CVPR.2014.49.

[3] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 39, no. 6, pp. 1137-1149, 2015, doi: 10.1109/TPAMI.2016.2577031.

[4] W. Liu *et al.*, "SSD: Single Shot Multi-Box Detector," *European Conference on Computer Vision (ECCV)*, 2016, pp. 21–37.

[5] J. Wu, W. Zhou, T. Luo, L. Yu, and J. Lei, "Multiscale multilevel context and multimodal fusion for RGB-D salient object detection," *Signal Processing*, vol. 178, 2021, doi: 10.1016/j.sigpro.2020.107766.

[6] H. Peng, B. Li, W. Xiong, W. Hu, and R. Ji, "RGBD, salient object detection: a benchmark and algorithms," *Proceedings of the European Conference on Computer Vision,* 2014, pp. 92–109.

[7] Y. Cheng, H. Fu, X. Wei, J. Xiao, and X. CaoDepth, "Enhanced saliency detection method," *Proceedings of International Conference on Internet Multimedia Computing and Service*, ACM, XiaMen, China, 2014, pp. 23–27.

[8] R. Ju, L. Ge, W. Geng, T. Ren, and G. WuDepth, "Saliency based on anisotropic center-surround difference," *Proc. 21th IEEE Int. Conf. Image Process*, Paris, France, 2014, pp. 1115–1119.

[9] C. Zhu and G. LiA, "A three-pathway paychobiological framework of salient object detection using stereoscopic technology," *2017 IEEE International Conference on Computer Vision Workshops*, Venice, Italy, 2014, pp. 3008–3014

[10] R. Cong, J. Lei, and C. Zhang, "Saliency detection for stereoscopic images based on depth confidence analysis and multiple cues fusion," *IEEE Signal Process. Lett.*, vol. 23, no. 6, pp. 819–823, 2016, doi: 10.1109/LSP.2016.2557347.

[11] C. Zhu, W. Zhang, and T. Li, "Exploiting the value of the center-dark channel prior for salient object detection," *ACM Transactions on Intelligent Systems and Technology,* vol. 10, no. 3, pp. 1-20, 2018, Art. no. 32, doi: 10.1145/3319368.

[12] X. Sun, P. Wu, and S. C. H. Hoi, "Face detection using deep learning: An improved faster RCNN approach," *Neurocomputing*, vol. 299, pp. 42–50, 2018, doi: 10.1016/j.neucom.2018.03.030.

[13] T. D. Bui, J. J. Lee, and J. Shin, "Incorporated region detection and classification using deep convolutional networks for bone age assessment," *Artificial Intelligence in Medicine*, vol. 97, pp. 1–8, 2019, doi: 10.1016/j.artmed.2019.04.005.

[14] H. H. Thodberg, S. Kreiborg, A. Juul, and K. D. Pedersen, "The bonexpert method for automated determination of skeletal maturity," *IEEE Trans Med Imaging*, vol. 28, no. 1, pp. 52–66, 2019, doi: 10.1109/TMI.2008.926067.

[15] Dhiraj, and D. K. Jain, "An evaluation of deep learning-based object detection strategies for threat object detection in baggage security imagery," *Pattern Recognition Letters*, vol. 120, pp. 112–119, 2019, doi: 10.1016/j.patrec.2019.01.014.

[16] V. Sharma and R. N. Mir, "Saliency guided faster-RCNN (SGFr-RCNN) model for Object detection and recoginition," *Journal of King Saud University – Computer and Information Sciences*, pp. 1–13, 2019, doi: 10.1016/j.jksuci.2019.09.012.

[17] K. Tong, Y. Wu, and F. Zhou, "Recent advances in small object detection based on deep learning: A review," *Image and Vision Computing*, vol. 97, 2020, doi: 10.1016/j.imavis.2020.103910.

[18] X. Wu, D. Sahoo, and S. C. H. Hoi, "Recent advances in deep learning for object detection," *Neuro Computing*, vol. 396, pp. 39–64, 2020, doi: 10.1016/j.neucom.2020.01.085.

[19] B. R. Kang, H. Lee, K. Park, H. Ryu, and H. Y. Kim, "Bshape Net: Object Detection and Instance segmentation with bounding shape masks," *Pattern Recognition Letters*, vol. 131, pp. 449–455, 2020, doi: 10.1016/j.patrec.2020.01.024.

[20] J. Shi, Y. Chang, C. Xu, F. Khan, G. Chen, and C. Li, "Real-time leak detection using an infrared camera and Faster R-CNN technique," *Computers and Chemical Engineering*, vol. 135, 2020, doi: 10.1016/j.compchemeng.2020.106780.

[21] E. Khatab, A. Onsy, M. Varley, and A. Abouelfarag, "Vulnerable objects detection for autonomous Driving: A review," *Integeration*, vol. 78, pp. 36–48, 2021, doi: 10.1016/j.vlsi.2021.01.002.

[22] C. J. Li, Z. Qu, S. Wang, and L. Liu, "A method of cross-layer fusion multi-object detection and recognition based on improved faster R-CNN model in complex traffic environment," *Pattern Recognition Letters*, vol. 145, pp. 127–134, 2021, doi: 10.1016/j.patrec.2021.02.003.

[23] W. Jia, Y. Tian, R. Luo, Z. Zhang, J. Lian, and Y. Zheng, "Detection and segmentation of overlapped fruits based on optimized mask R-CNN application in apple harvesting robot," *Computers and Electronics in Agriculture*, vol. 172, 2020, doi: 10.1016/j.compag.2020.105380.

[24] R. Hannane, A. Elboushaki, and K. Afdel, "A divide-and-conquer strategy for facial landmark Detection using dual-task CNN architecture," *Pattern Recongnition,* vol. 107, 2020, doi: 10.1016/j.patcog.2020.107504.

[25] S. Parvathi and S. T. Selvi, "Detection of maturity stages of coconuts in complex background using Faster R-CNN model," *Biosystems Engineering*, vol. 202, pp. 119–132, 2021, doi: 10.1016/j.biosystemseng.2020.12.002.

BIOGRAPHIES OF AUTHORS

M. Karthikeyan pursuing doctorate in Annamalai University, Faculty of Engineering and Technology, TamilNadu, India. Currently, he is working as an Assistant Professor, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Tamilnadu, India. His Research Interests include Deep learning, Internet of Things and Computer vision.

T. S. Subashini, Professor, Department of Computer Science and Engineering, Annamalai University, Tamilnadu, India. Her Research Interests include Image Processing, Pattern recognition, Deep Learning.