Vision-based Human Detection by Fine-Tuned SSD Models

Tang Jin Cheng[1], Ahmad Fakhri Ab. Nasir[2]*, Mohd Azraai Mohd Razman[4]
Innovative Manufacturing, Mechatronics and Sports Laboratory[1,2,4]
Faculty of Computing[2]
Faculty of Manufacturing and Mechatronic Engineering Technology[1,4]
Universiti Malaysia Pahang, 26600 Pekan
Pahang, Malaysia

Anwar P. P. Abdul Majeed[3]
School of Robotics, XJTLU Entrepreneur College (Taicang), Xi'an Jiaotong-Liverpool University
Suzhou, 215123, P. R. China

Thai Li Lim[5]
TT Vision Holdings Berhad, 11900 Plot 106, Sungai Hilir Keluang 5, Bayan Lepas FIZ.4, Bayan Lepas
Pulau Pinang, Malaysia

***Abstract*—Human-robot interaction (HRI) and human-robot collaboration (HRC) has become more popular as the industries are taking initiative to idealize the era of automation and digitalization. Introduction of robots are often considered as a risk due to the fact that robots do not own the intelligent as human does. However, the literature that uses deep learning technologies as the base to improve HRI safety are limited, not to mention transfer learning approach. Hence, this study intended to empirically examine the efficacy of transfer learning approach in human detection task by fine-tuning the SSD models. A custom image dataset is developed by using the surveillance system in TT Vision Holdings Berhad and annotated accordingly. Thereafter, the dataset is partitioned into the train, validation, and test set by a ratio of 70:20:10. The learning behaviour of the models was monitored throughout the fine-tuning process via total loss graph. The result reveals that the SSD fine-tuned model with MobileNetV1 achieved 87.20% test AP, which is 6.1% higher than the SSD fine-tuned model with MobileNetV2. As a trade-off, the SSD fine-tuned model with MobileNetV1 attained 46.2 ms inference time on RTX 3070, which is 9.6 ms slower as compared to SSD fine-tuned model with MobileNetV2. Taking test AP as the key metric, SSD fine-tuned model with MobileNetV1 is considered as the best fine-tuned model in this study. In conclusion, it has shown that the transfer learning approach within the deep learning domain can help to protect human from the risk by detecting human at the first place.**

Keywords*—Human detection; deep learning; transfer learning; SSD; fine-tuning; human-robot interactions

I. Introduction

Robotics and automation systems has become a key technology that can help in creating an idealistic future [1]. According to the worldwide trend, it is reported that although the deployment of concepts related to human-robot collaboration (HRC) and human-robot interaction (HRI) has increase progressively, yet it shows that the adoption of the market on these concepts is still in the early stage [2]. As much as the HRI concepts are concerned, the introduction of robots can be a stumbling block to the industries as it possesses potential risks to the human workers when it comes to sharing of working space between the robots and the human workers [3]. Perhaps, both human workers and robots must ensure appropriate communication to promote safe collaboration [4].

Insufficient safety devices can be the most remarkable barrier in forming the trust of HRI concept [5]. It has been found out that the occupational safety has ranked as the most important factor in bringing successful HRI applications. Enormous amount of research has been carried out to improve the safety of HRI [6]–[8]. In light of this, the major safety mechanisms can be generally categorized into four senses, which are vision, tactile, audition and distance [9]. It is noted that even though vision-based sensor seems to be the trickiest and most computationally extensive, it is still considered to have the most unique sense owing to its irreplaceable richness of the features extracted from this sense. With that in mind, vision-based sensors were incorporated in the present study.

Recent advances in deep learning and transfer learning have garnered great success in various domains [10]–[12]. In particular to HRI applications, limited number of studies has exploited on the usage of deep learning in ensuring a safe interaction and collaboration, not to mention about using transfer learning approach. Within this context, this study intended to examine the applicability of deep learning-based object detection approach via transfer learning approach to detect human at the first place and evaluate on the performance of such approach.

II. Related Works

Mohammed et al. [13] presented a novel online collision avoidance approach that only relies on inputs from Microsoft Kinect sensors. The depth images were used with the background subtraction approach to obtain the virtual model of the human operators. Distance between human operators and the robot was then computed for the collision detection based on a threshold value. Instead of solely rely on the Microsoft Kinect sensors, Magrini et al. [14] established a layered control architecture to improve the safety of HRC applications by integrating multiple sensors and controllers. Microsoft Kinects sensors were utilized to compute the

*Corresponding Author.

distance between two parties while laser scanners were deployed to ensure in which area the human worker is located. Both studies used dual-camera settings as to obtain the depth information of the workers as well as to detect the presence of the human operators within the robot cell.

A study has claimed that it is the first time a deep neural network is utilized for developing a real-time collision detection system and solid results are yielded with a highly satisfied detection speed [15]. Joint signals were used as the input of the training of the deep neural networks. The authors highlighted that the study is limited to cyclic motion due to the fact that cycle-normalization technique is applied. Another study integrated both tactile and visual perception for safety purposes [16]. 1D-CNN was used for the physical contact detection while 3D-CNN was meant for the visual perception to perform human action recognition. Joint signals and images are used as the input for the networks, respectively.

As for the advance of deep learning, transfer learning seems to be a favourable approach in different research fields. The effectiveness of transfer learning approach are investigated to extract the wink-based EEG signals that is converted by means of continuous wavelet transform (CWT) method [17]. The findings were prominent which can potentially be used for controlling the rehabilitation devices. A deep learning-based pre-trained object detection model were adopted in the research with huge amount of data to diagnose the rice leaf disease in a real-time manner [12]. The results further justify the applicability of deep learning technologies, might as well for the transfer learning approach in yielding promising results across different domain.

III. Methodology

In this section, a general description of the overall research workflow is provided. First, the image acquisition process is explained and the dataset that is used in this work is reported, followed by the image annotation stage. Next, appropriate files are generated for training the models under specific deep learning framework. The training strategies and procedures are then clarified and visualized in a flowchart for better glance of the process. Lastly, the performance metrics that is used to evaluate the performance of the models are described.

A. Image Acquisition

The surveillance cameras that are mounted in TT Vision Holdings Berhad were used to acquire the image dataset. In particular, the recorded video footages were obtained from the surveillance database to extract relevant images. Since it is always beneficial to introduce variation to the deep learning models, the location of the surveillance cameras was carefully chosen. Among all the surveillance cameras, it was observed that the people working in the production area will have higher possibility of moving around compared to the office due to their work nature. Therefore, only the surveillance cameras that are in the production area were selected.

In total, 1463 images were acquired from the recorded video footages. Example of the obtained images was depicted in Fig. 1. In compliance to the standard data splitting procedure, the dataset was separated into three portions, which are training, validation, and testing. The ratio of the data splitting used in this study was set to be 70:20:10. In addition, data augmentation methods such as horizontal flipping and cropping were utilized to further increase the size of the image dataset and allow more variation of the dataset.

Fig. 1. Examples of Training Images.

B. Image Annotation

All the human workers that shown up in the images were annotated manually with a tight bounding box and labelled with the “Person” class name. Not to confuse the model, it is noted that in this study the occluded part of the human workers was not annotated. With respect to the image annotation procedure, a popular image annotation tool known as LabelIMG [18] was used to perform the annotation. The output annotation files with the format of PASCAL VOC were generated after the bounding box annotation.

C. TensorFlow Records (TFRecords) Generation

Throughout the study, TensorFlow is used as the deep learning framework to develop the deep learning-based object detection models. Thus, TFRecords files were required to generate because this is the only format that can be translated by TensorFlow library for loading the datasets. For instance, TFRecord files can be understood as a simple format that stores the dataset as sequence of binary strings for efficiency purposes. A script was used to iterates through all the annotations in the XML files so that the annotations can be converted into TFRecord annotation files. Since TFRecords only stores binary strings, the label class are stored in binary value and a label map is required for the annotations to have a reference on the class name. For this reason, a one-class label map was developed for mapping the class integer presented in the TFRecords file and the class name.

D. Transfer Learning: Fine-tuning

An open-source deep learning framework that is developed on top of TensorFlow known as TensorFlow Object Detection API was leveraged since it can be considered as a ready to use toolkit in developing, training and inferencing object detection models. Particularly, the Single Shot Multibox Detector (SSD) within the TensorFlow 2 Detection Model Zoo was trained to perform human detection tasks [19]. The meta-architecture of the fine-tuned SSD models is shown in Fig. 2. In this study, both MobileNetV1 and MobileNet V2 that were pretrained on

COCO 2017 dataset were used as the backbone networks [20], [21]. The feature pyramid network was deployed as the neck to extract richer semantic features [22]. Whereas the single shot convolutional prediction head was used correspond to the class prediction and the bounding box regression.

Instead of training the object detection models from scratch, transfer learning approach known as fine-tuning was applied as it is beneficial from the perspective of training time as well as the requirement of a large dataset. By mean of fine-tuning strategy, it is indicated that only the detection head was subject to the training of human detection task, while the backbone network and the neck was remained.

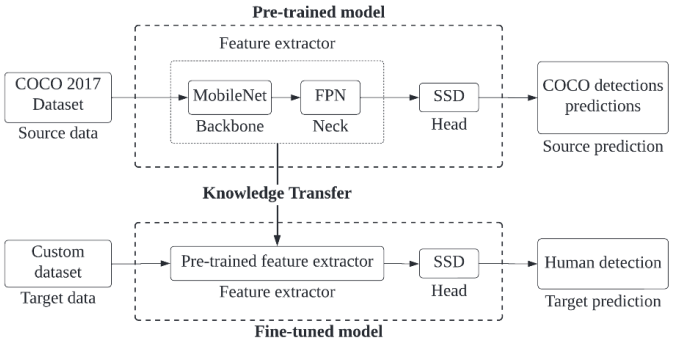

Fig. 2. Meta-architecture of Fine-tuned SSD Models.

E. Learning Rate and Loss Inspection

Various hyperparameters can be adjusted in order to improve the learning performance of the models. Learning rate acts an important role due to the fact that it can affects the scale of how much the weights are updated within the networks. Instead of using a mere value for the learning rate, this hyperparameter was scheduled via the cosine annealing and warm restart techniques [23]. The motivation behind is to improve the learning behaviour and avoid the occurrence of exploding gradient and vanishing gradient. Both SSD models were configured to undergo a 2.5k warmup steps with learning rate of 0.001. In total, the models were fine-tuned for 100k training steps with learning rate of 0.01.

As for deep learning approaches as well as transfer learning, the learning behaviour has to be monitored throughout the training process. Despite that the scheduling techniques applied can help in improving the model learning, still it is possible for exploding gradient and vanishing gradient to take place. In this sense, loss graphs were utilized to further ensure the learning behaviour of the SSD models goes well throughout the fine-tuning process.

F. Performance Evaluation

In the field object detection, Average Precision (AP) is commonly used as a standard evaluation metric to evaluate the robustness of an object detection model. With regards to the object detection task, precision is computed based on the Intersection over Union (IoU) threshold. IoU can be defined as the ratio of the overlap area between the predicted bounding box and the ground-truth box to the union area of these two boxes. The concept of IoU is visualized in Fig. 3.

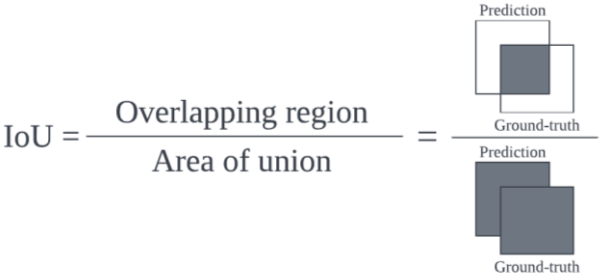

Fig. 3. Intersection over Union (IoU).

According to the predefined IoU threshold value, the predictions were then be classified as true or false. As such, the IoU threshold value is set to 0.5 in the present study. After the calculation of precision and recall, AP was then calculated by computing the area under the precision-recall curve with the expression as below:

$$AP = \sum_n (Recall_n Recall_{n-1}) Precision_n \quad (1)$$

IV. Result and Discussion

In this section, the results were reported throughout the fine-tuning process. The total loss of the models over the fine-tuning process was visualized and discussed, followed by the performance evaluation of the SSD models after the fine-tuning process. Then, the performance of the SSD models, particularly the AP was reported together with several considerable performance details.

A. Loss Inspection via Loss Graphs

To ensure the fine-tuning of the object detection models are neither underfitting nor overfitting, the learning behaviour of the models were monitored throughout the fine-tuning process via the training and validation loss graph [24], [25]. The training loss and validation loss graphs were plotted as in Fig. 4 and 5. As described in the graph legends, the orange line is the training loss while the grey line is the validation loss. Each of them corresponds to the training set and the validation set of the image dataset.

From both learning curves, it can be clearly seen that the training and the validation losses greatly decrease at the beginning and end up converge at the end of the fine-tuning process. It is reasonable for the losses to be high in the initial stage because the knowledge learned from the previous domain is not specifically cater for the human detection task. However, as the training steps increase, the models had learned the important features for the human detection tasks, hence the losses decrease over the iterations. Towards the end of the fine-tuning process, the losses have become more stable suggesting that the models have converge. In addition, it is shown that the validation loss curve stays above the training loss curve for both SSD models, in turn indicates that the fine-tuned models are neither underfitting nor overfitting.

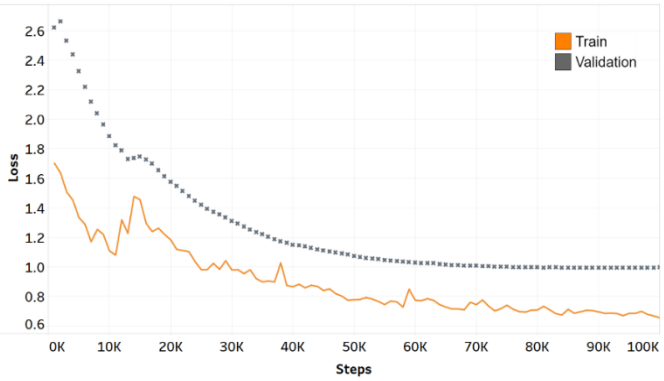

Fig. 4. Learning Curve of Fine-tuned SSD_MobileNetV1.

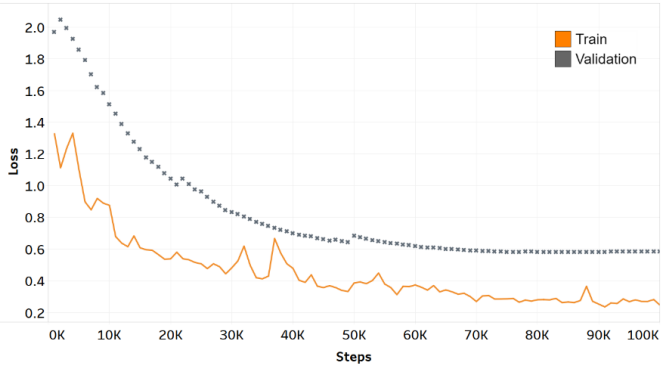

Fig. 5. Learning Curve of Fine-tuned SSD_MobileNetV2.

B. Performance Evaluation

The AP of the fine-tuned models is depicted as in Fig. 6 and the performance details of the fine-tuned models are tabulated in Table I [26]. The SSD_MobileNetV1 is referring to the SSD model with MobileNetV1 as the base network while SSD_MobileNetV2 is referring to the SSD model with MobileNetV2 as the backbone network. In the performance evaluation stage, the SSD_MobileNetV1 achieves 94.10% train AP, 86.90% validation AP and 87.20% test AP with an inference speed of 46.2 ms. Whereas SSD_MobileNetV2 achieves 89.40% train AP, 82.40% validation AP and 81.10% test AP with an inference speed of 36.6ms. For instance, SSD_MobileNetV1 has a model size of 4.62MB and 10.89 million parameters, while SSD_MobileNetV2 has a model size of 6.46MB and 2.60 million of parameters.

Discussing with regards to the model size and number of parameters, noted that although the SSD_MobileNetV1 fine-tuned model has greater number of parameters than SSD_MobileNetV2, but it has a smaller model size than the SSD_MobileNetV2 fine-tuned model. This is because in the implementation of SSD_MobileNetV2, depthwise separable convolution technique is used for reduction of parameters. Still, the model size is not reduced given the fact the depth of SSD_MobileNetV2 is more than SSD_MobileNetV1.

Despite the comparison of model size and number of parameters, AP and inference speed are the key factors to evaluate the performance of the fine-tuned models. From Fig. 6 and Table I, it can be observed that the SSD_MobileNetV1 has higher AP for all three train, validation, and test AP with a slower inference speed on RTX 3070 GPU as compared to SSD_MobileNetV2. In fact, the only difference between these two fine-tuned models is the backbone network, hence the differences in AP and inference speed are most probably attributed to the architecture of the base network used in the fine-tuned models [27]–[30]. In a better context, SSD_MobileNetV2 is said to be the recommended fine-tuned model if inference speed comes into consideration before AP. Since this study has considered AP to be more important than inference speed as it is concerned with the HRI safety, SSD_MobileNetV1 is a better choice among these two SSD fine-tuned models. In summary, SSD_MobileNetV1 is proposed as the best fine-tuned model in this case with respect to the human detection task.

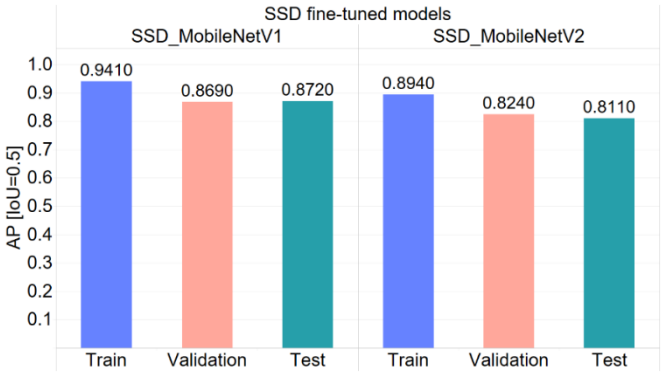

Fig. 6. Average Precision of SSD Fine-tuned Models with Respect to Train, Validation and Test Dataset.

TABLE I. PERFORMANCE DETAILS OF THE SSD FINE-TUNED MODELS

SSD Fine-Tuned Models	Performance Details		
	Model Size	No. of Parameters	Inference speed
SSD_MobileNetV1	4.62 MB	10.89M	46.2 ms
SSD_MobileNetV2	6.46 MB	2.60M	36.6 ms

V. CONCLUSION

In the present study, the surveillance system was used to acquire the image dataset for human detection task. The human workers presented in the images were annotated with relevant annotation tools. According to the deep learning framework that has been used in this study, the dataset was developed to specific format that suits the framework. The transfer learning strategy called fine-tuning was leveraged to decrease the training time. In particular, the pre-trained weights of the base network were restored, only the prediction head was subjected to the fine-tuning by using the custom dataset. The result has shown that the SSD_MobileNetV1 fine-tuned model has the highest AP with tolerable sacrifice of inference speed.

ACKNOWLEDGMENT

The authors would like to thank TT Vision Holdings Berhad for providing the image dataset to make this evaluation possible as well as for funding the study in

collaboration with Universiti Malaysia Pahang via UIC200816 and RDU202405.

REFERENCES

[1] L. Onnasch and E. Roesler, "A Taxonomy to Structure and Analyze Human–Robot Interaction," Int. J. Soc. Robot., vol. 13, no. 4, pp. 833–849, 2021, doi: 10.1007/s12369-020-00666-5.

[2] International Federation of Robotics, "World Robotics Report 2019," 2020. [Online]. Available: https://ifr.org/downloads/press2018/2020-09-24_IFR_press_release_WR_industrial_robots.pdf.

[3] M. Vasic and A. Billard, "Safety issues in human-robot interactions," in 2013 IEEE International Conference on Robotics and Automation, May 2013, pp. 197–204, doi: 10.1109/ICRA.2013.6630576.

[4] V. V. Unhelkar, S. Li, and J. A. Shah, "Decision-making for bidirectional communication in sequential human-robot collaborative tasks," ACM/IEEE Int. Conf. Human-Robot Interact., pp. 329–341, 2020, doi: 10.1145/3319502.3374779.

[5] T. Kopp, M. Baumgartner, and S. Kinkel, "Success factors for introducing industrial human-robot interaction in practice: an empirically driven framework," Int. J. Adv. Manuf. Technol., vol. 112, no. 3–4, pp. 685–704, Jan. 2021, doi: 10.1007/s00170-020-06398-0.

[6] Z. M. Bi, C. Luo, Z. Miao, B. Zhang, W. J. Zhang, and L. Wang, "Safety assurance mechanisms of collaborative robotic systems in manufacturing," Robot. Comput. Integr. Manuf., vol. 67, no. January 2020, p. 102022, Feb. 2021, doi: 10.1016/j.rcim.2020.102022.

[7] D. Kim et al., "Design of a sensitive balloon sensor for safe human–robot interaction," Sensors, vol. 21, no. 6, pp. 1–12, 2021, doi: 10.3390/s21062163.

[8] S. Robla-Gomez, V. M. Becerra, J. R. Llata, E. Gonzalez-Sarabia, C. Torre-Ferrero, and J. Perez-Oria, "Working Together: A Review on Safe Human-Robot Collaboration in Industrial Environments," IEEE Access, vol. 5, pp. 26754–26773, 2017, doi: 10.1109/ACCESS.2017.2773127.

[9] A. Cherubini and D. Navarro-Alarcon, "Sensor-Based Control for Collaborative Robots: Fundamentals, Challenges, and Opportunities," Front. Neurorobot., vol. 14, no. January, pp. 1–14, Jan. 2021, doi: 10.3389/fnbot.2020.576846.

[10] O. P. Toon, M. A. Zakaria, A. F. Ab. Nasir, A. P.P. Abdul Majeed, C. Y. Tan, and L. C. Y. Ng, "Autonomous Tomato Harvesting Robotic System in Greenhouses: Deep Learning Classification," MEKATRONIKA, vol. 1, no. 1, pp. 80–86, Jan. 2019, doi: 10.15282/mekatronika.v1i1.1148.

[11] J. L. Mahendra Kumar et al., "An Evaluation of Different Fast Fourier Transform - Transfer Learning Pipelines for the Classification of Wink-based EEG Signals.pdf," MEKATRONIKA, vol. 2, no. 1, pp. 1–7, 2020, doi: https://doi.org/10.15282/mekatronika.v2i1.4881.

[12] B. S. Bari et al., "A real-time approach of diagnosing rice leaf disease using deep learning-based faster R-CNN framework," PeerJ Comput. Sci., vol. 7, pp. 1–27, 2021, doi: 10.7717/PEERJ-CS.432.

[13] A. Mohammed, B. Schmidt, and L. Wang, "Active collision avoidance for human–robot collaboration driven by vision sensors," Int. J. Comput. Integr. Manuf., vol. 30, no. 9, pp. 970–980, Sep. 2017, doi: 10.1080/0951192X.2016.1268269.

[14] E. Magrini, F. Ferraguti, A. J. Ronga, F. Pini, A. De Luca, and F. Leali, "Human-robot coexistence and interaction in open industrial cells," Robot. Comput. Integr. Manuf., vol. 61, no. June 2018, p. 101846, Feb. 2020, doi: 10.1016/j.rcim.2019.101846.

[15] Y. J. Heo, D. Kim, W. Lee, H. Kim, J. Park, and W. K. Chung, "Collision Detection for Industrial Collaborative Robots: A Deep Learning Approach," IEEE Robot. Autom. Lett., vol. 4, no. 2, pp. 740–746, Apr. 2019, doi: 10.1109/LRA.2019.2893400.

[16] F. M. Amin, M. Rezayati, H. W. van de Venn, and H. Karimpour, "A Mixed-Perception Approach for Safe Human–Robot Collaboration in Industrial Automation," Sensors, vol. 20, no. 21, p. 6347, Nov. 2020, doi: 10.3390/s20216347.

[17] J. L. Mahendra Kumar et al., "The classification of EEG-based wink signals: A CWT-Transfer Learning pipeline," ICT Express, vol. 7, no. 4, pp. 421–425, 2021, doi: 10.1016/j.icte.2021.01.004.

[18] L. Tzu Ta, "LabelImg. Git code." 2015, Accessed: Jun. 28, 2021. [Online]. Available: https://github.com/tzutalin/labelImg.

[19] W. Liu et al., "SSD: Single Shot MultiBox Detector," in Eccv, vol. 9905, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 21–37.

[20] A. G. Howard et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," arXiv, Apr. 2017, [Online]. Available: http://arxiv.org/abs/1704.04861.

[21] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Jun. 2018, pp. 4510–4520, doi: 10.1109/CVPR.2018.00474.

[22] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," Proc. - 2019 IEEE Intl Conf Parallel Distrib. Process. with Appl. Big Data Cloud Comput. Sustain. Comput. Commun. Soc. Comput. Networking, ISPA/BDCloud/SustainCom/SocialCom 2019, pp. 1500–1504, Dec. 2016, doi: 10.1109/ISPA-BDCloud-SustainCom-SocialCom48970.2019.00217.

[23] I. Loshchilov and F. Hutter, "SGDR: Stochastic Gradient Descent with Warm Restarts," Aug. 2016, doi: 10.48550/arxiv.1608.03983.

[24] C. Zheng et al., "Detecting glaucoma based on spectral domain optical coherence tomography imaging of peripapillary retinal nerve fiber layer: a comparison study between hand-crafted features and deep learning model," Graefe's Archive for Clinical and Experimental Ophthalmology, vol. 258, no. 3. pp. 577–585, 2020, doi: 10.1007/s00417-019-04543-4.

[25] S. Du et al., "The connectivity evaluation among wells in reservoir utilizing machine learning methods," IEEE Access, vol. 8, pp. 47209–47219, 2020, doi: 10.1109/ACCESS.2020.2976910.

[26] H. J. Lee, I. Ullah, W. Wan, Y. Gao, and Z. Fang, "Real-Time vehicle make and model recognition with the residual squeezenet architecture," Sensors (Switzerland), vol. 19, no. 5, 2019, doi: 10.3390/s19050982.

[27] A. F. Nurfirdausi, S. Soekirno, and S. Aminah, "Implementation of Single Shot Detector (SSD) MobileNet V2 on Disabled Patient's Hand Gesture Recognition as a Notification System," 2021 Int. Conf. Adv. Comput. Sci. Inf. Syst. ICACSIS 2021, pp. 19–24, 2021, doi: 10.1109/ICACSIS53237.2021.9631333.

[28] W. Rahmaniar and A. Hernawan, "Real-time human detection using deep learning on embedded platforms: A review," J. Robot. Control, vol. 2, no. 6, pp. 462-468Y, 2021, doi: 10.18196/jrc.26123.

[29] X. Gao, J. Xu, C. Luo, J. Zhou, P. Huang, and J. Deng, "Detection of Lower Body for AGV Based on SSD Algorithm with ResNet," Sensors, vol. 22, no. 5, 2022, doi: 10.3390/s22052008.

[30] B. Mathurabai, V. P. Maddali, C. Devineni, and I. Bhukya, "Object Detetcion using SSD-MobileNet," pp. 2668–2671, 2022.