Multi-Target Classification using Deep Learning Models for Automotive Applications

Soumya. A[1], Linga Reddy Cenkeramaddi[2]*, Chalavadi Vishnu[1], Yash Vinod Lanjewar[3], and Krishna Mohan C[1]
[1]*Department of Computer Science and Engineering, Indian Institute of Technology, Hyderabad, India*
[2]*Department of Information and Communication Technology, University of Agder, Grimstad, Norway*
[4]*Department of Architecture and Regional Planning, Indian Institute of Technology, Kharagpur, India*
Corresponding author: linga.cenkeramaddi@uia.no

2023 11th International Conference on Control, Mechatronics and Automation (ICCMA) | 979-8-3503-1568-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICCMA59762.2023.10375062

***Abstract*— This paper presents a multi-perspective convolutional neural network (CNN) that extracts the class of objects supporting intelligent transportation systems. The proposed model is the visual geometry group (VGG) backbone network with custom feature extraction blocks, that use multilayer prediction heads. The model addresses both multi-class and multi-object classification tasks utilizing the automotive object detection dataset. The model is designed with multiple prediction heads to classify different objects in an image and enable object count prediction. A publicly available automotive object detection dataset with 19800 images and labels has been utilized. The dataset consists of five primary types of objects: Persons, Trucks, Motorbikes, Cars, and Cyclists. On the dataset, pre-trained models such as VGG, Resenet, EfficientNet, and DenseNet were tested and their classification performance was evaluated. The experimental results illustrate the superiority of the proposed VGG backbone deep learning CNN model in comparison to other pre-trained models.**

***Index Terms*—Deep Learning, Convolutional Neural Network, Image Classification, Multi-target Classification, Multi-object Prediction.**

I. Introduction

In recent years, the discipline of computer vision has seen exponential growth and transformative advancements, transforming how we interact with technology, analyze data, and interpret the world around us. In an era where visual data is abundant, computer vision has emerged as a cornerstone technology, playing a pivotal role in a wide range of industries, from healthcare and autonomous vehicles to entertainment and agriculture covering a wide range of applications. Image classification with a convolutional neural network (CNN) is a fundamental idea of how the model learns to perform feature extraction and it convolves images and filters to provide the invariant features that are passed on to the next layer in the model. The features in the following layer are combined with various filters to provide more invariant and abstract features, and the process is continued until the final feature, which is occlusion-invariant, is acquired. Earlier, CNN used to perform poorly when faced with complex issues like classifying high-resolution images due to a lack of sufficient training data, a lack of improved regularization techniques, and insufficient computational power. But today, CNN excels at image classification tasks even with larger datasets like ImageNet [1], due to the development of powerful GPU processors and enhanced regularisation methods. This paper describes a multi-class object classification for camera images based on deep learning. Numerous pre-trained deep learning models are available in the literature for classifying image datasets. The pre-trained model's assessment of the new image data serves as a good benchmark for the recently suggested architectures. The pre-trained models available are DenseNet [2], MobileNet [3], ResNet [4], VGGNet [5], EfficientNet [6], and InceptionNet [7] etc.

Image classification tasks encounter several challenges, such as scale variation, which makes it difficult to accurately classify objects of different sizes and scales. Another challenge is object counting, where models struggle to estimate the number of specific objects present in an image. Additionally, task specialization poses a problem, as training models to excel in specific classification tasks can be challenging. To tackle these issues, the proposed model incorporates a multilayer prediction head, allowing separate branches to learn task-specific parameters and specialize in classifying objects of various scales. Furthermore, the model goes beyond traditional image classification by enabling object count prediction, providing a comprehensive solution to the challenges in image classification tasks.

In this paper, we introduce a deep learning model, namely, the VGG backbone convolutional neural network, multi-class image classification inclusive of multi-object prediction. In the proposed model, we resized the original images into the input image size of size $224 \times 224 \times 3$ and passed through the feature extraction layer. The feature extraction layer extracts hierarchical representations, capturing essential low-level and high-level features for object recognition. The extracted features are then fed into separate prediction branches, each responsible for classifying objects at different scales. The model generates ten predictions, corresponding to potential objects in an image. If an image contains fewer than ten objects, the remaining prediction boxes are detected but suppressed during result analysis. We conducted image classification testing for these classes: Person, Truck, Motorbike, Car, and Cyclist.

The contributions of this paper are as follows: We propose an image classification model that addresses both multi-class and multi-object prediction tasks.

- A multi-layer prediction strategy, our approach enables separate prediction branches to acquire dedicated parameters for learning.
- Enabling accurate object classification across varying scales.
- Crafted with precision to effectively categorize objects spanning diverse classes.
- Predicting the number of objects within each class.

Moreover, the study showcases the effectiveness of the suggested CNN model in contrast to various alternative architectures developed for similar classification tasks. Overall, the contributions in this paper offer valuable insights and advancements in the field of multi-class classification along with multi-object prediction, highlighting the capabilities of the proposed innovative model architecture and multilayer prediction approach.

The structure of this paper is as follows: Section II introduces other works related to image classifications with CNNs; Section III describes the proposed CNN model that we use as the multi-class multi-object image classifier; Sections IV and V cover the dataset details and the evaluation of the state-of-the-art CNNs with automotive dataset. In Section VI, we present two experiments, involving frozen and trainable weights. We present evaluation results through comparisons to demonstrate the enhanced performance of the proposed image classification model. Finally, Section VII covers the conclusion of the paper.

II. Related Works

This section summarizes the existing works on multi-class image classification. Traditional machine learning and deep learning approaches for image classification are presented in [8]. Several works were proposed for image classification with convolutional neural networks (CNN) classifiers in [9]. A deep study to understand the CNNs is presented in [10]. The analysis of image feature extraction in CNN is presented in [11]. Applying a supervised deep learning model for classifying satellite images using a CNN reported in [12], helps in remote sensing applications. A CNN-based hyperspectral image classification is designed in [13] using 2D spatial features with multiple scales and 1D spectral features. A CNN was trained in [14] to classify the dental problems. An ensemble of CNNs is used in [15] to get the predictions of both CNNs to classify the satellite images at various views. A model of ensemble CNNs is developed in [16] for vehicle-type classification while maximizing the accurate predictions on the images consisting of unbalanced data. In [17], feature extraction-based CNN is designed for vehicle type and vehicle color classification. A model in [18] presents the scene classification using unlabeled samples. The authors in [19], presented the multiple-view deep convolutional neural network design for automatic target recognition in synthetic aperture radar (ATR-SAR) applications by enhancing detection rates through selective topographies and fusing features from multiple SAR images. An approach in [20], that combines fine-tuning EfficientNets models with power mean support vector machine for classifying insect images across various life cycle stages. In [21], A deep CNN is designed with AlexNet for extracting discriminative information, support vector machine, and softmax classifiers for improving prediction performance.

III. The proposed work

We have trained multiple convolutional neural network models in addition to the proposed model for classification, aiming to discern and assess the variations in prediction outcomes resulting from alterations in the feature extraction layer and prediction head. The proposed CNN model includes:

A. Model: CNN with multi-layer prediction and VGG backbone

We propose a model with a multilayer prediction head with a VGG16 feature extraction backbone. The model named VGG backbone CNN model for multi-class, multi-label classification is shown in Fig. 1 utilizes a pre-trained VGG16 feature extractor backbone with pre-trained weights on the ImageNet dataset. The VGG backbone is frozen, ensuring that the pre-trained weights remain unchanged during training. Applying a pre-trained VGG backbone for feature extraction offers several advantages in image classification tasks. Firstly, VGG is a well-established and widely-used architecture with proven performance in various image recognition competitions and benchmarks. Due to its depth and simplicity, it is capable of capturing both high-level and low-level features, making it effective in recognizing intricate patterns and details in images.

Following the feature extraction backbone, additional layers are introduced to further process the extracted features. These layers include convolutional (Conv2D) and fully connected layers (Dense), with Rectified Linear Unit (ReLU) activations applied after each layer. This combination of layers enables the model to capture and refine the features extracted from the backbone, enhancing the accuracy of the model. The model's output is structured to support multi-class, multi-label classification. It comprises ten output branches, each responsible for making a distinct classification decision.

An important aspect of the model architecture is its ability to extract features at multiple scales. This is accomplished by feeding the features extracted from the backbone into separate branches of the model. For instance, features from the layer after the first Conv2D layer are used for predicting three output classes (output_class1, output_class2, and output_class3). By leveraging features from subsequent layers, the model in Table I can effectively capture and classify objects of various scales, utilizing features at different levels of abstraction.

We acknowledge that object prediction can encounter challenges in distinguishing between multiple instances of the same class and a single object captured by multiple heads. This aspect of our model is carefully addressed during the training phase to enhance the specialization of prediction layers. When supplying specific class labels to various prediction layers, we

TABLE I: VGG backbone model

Layer Name	Input Layer	Output Size
Input		$224 \times 224 \times 3$
VGG16 Backbone	Input	$7 \times 7 \times 512$
Conv 1	VGG16 Backbone	$5 \times 5 \times 512$
Flatten1	Conv 1	12800×1
Dense 1A	Flatten1	512×1
Dense 1B	Dense 1A	256×1
Conv 2	Conv 1	$5 \times 5 \times 512$
Conv 3	Conv 2	$3 \times 3 \times 256$
Flatten 2	Conv3	2304×1
Dense 2A	Flatten 2	256×1
Dense 2A	Dense 2A	128×1
Conv 4	Conv 3	$1 \times 1 \times 128$
Conv 5	Conv 4	$1 \times 1 \times 128$
Flatten 3	Conv 5	128×1
Dense 3	Flatten 3	128×1
3 x Dense 4	Dense 1B	7×1
3 x Dense 5	Dense 2B	7×1
3 x Dense 6	Dense 3	7×1

arrange the labels for each image in a meticulously designed order. This order ensures that each layer is fed with labels possessing distinct characteristics, thereby enabling the layers to specialize in the prediction of those particular classes. Furthermore, in arranging these labels, we take into account the scale at which each class predominantly occurs in the dataset. The labels are organized in decreasing order of scale, aligning with the flow from shallower prediction heads to deeper ones. This strategic arrangement aids each layer in developing an inclination for predicting classes of varying scales.

Additionally, we employ a data-feeding strategy that encourages each layer to slightly specialize in predicting specific classes. For instance, within the dataset, the "person" class exhibits a particular characteristic. Most instances of this class are situated closer to the camera. This arrangement allows the initial prediction layers to become more adept at predicting persons as well as large-scale objects. By structuring the training process in this manner, the model optimizes its capacity to classify objects with varying characteristics and scales, leading to enhanced accuracy and robustness in object recognition and counting tasks.

IV. DataSet and Image collection

A publicly available automotive object detection dataset in IEEE Dataport [22] is utilized. This dataset comprises camera images corresponding to five classes with varied dimensions. The dataset contains 19800 images and corresponding labels. Among these, we randomly selected 15777 images for training, 1973 images for validation, and 1972 for testing. The camera image of size $1440 \times 1080 \times 3$ is resized to $224 \times 224 \times 3$. There may be one or more objects in one image, so the location of each object is pre-annotated. All the objects in the dataset are divided into five categories: person, truck, pedestrian, car, and cyclist.

V. Evaluation of the State-of-the-Art CNNs

In this part, pre-trained models such as EfficientNet, DenseNet, ResNet, and VGG are trained and tested with the automotive image dataset available in IEEE Dataport [22]. The models are evaluated to determine their accuracy, and the results and overall accuracy are shown in Table II. Comparatively, the proposed model gives high accuracy with fewer parameters and helps in real-time road scene classifications. The proposed model can increase mean precision to some extent, compared to the baseline models. Concretely, the proposed CNN improves accuracy by more than 20% in contrast with the individual pre-trained models.

VI. Experiments

A. *Comparison of VGG backbone using trainable and frozen weights:*

The proposed VGG backbone model is compared by setting all the frozen layers as trainable, and we observed distinct training dynamics between a model with frozen weights and the model with trainable weights initialized with ImageNet weights. The model with frozen weights exhibited a smooth loss epoch curve shown in Fig. 2, while the model with trainable weights displayed a more variable loss epoch curve shown in Fig. 3, despite both models achieving similar loss values at the end. The smooth loss epoch curve observed in the model with frozen weights can be attributed to the stability provided by the pre-trained weights. These weights are optimized using an extensive dataset such as ImageNet, capturing important information about general visual features. The model was equipped with a solid initialization by leveraging these pre-trained weights as a starting point, leading to a more stable training process.

On the other hand, the model with trainable weights initialized with ImageNet weights underwent a process of fine-tuning, allowing it to adapt the pre-trained weights to the specific task or dataset at hand. Consequently, the loss epoch curve exhibited variability compared to the model with frozen weights. Although the loss epoch curve exhibited fluctuations, the final loss value achieved by the model with trainable weights was comparable to that of the model with frozen weights. In contrast, this comes with the disadvantage of having twice the number of trainable parameters and more converging epochs. These findings emphasize the trade-off between stability and adaptability when working with pre-trained models.

B. *Evaluation Metrics*

Classification assessment metrics were utilized to measure the performance of the proposed classification model in various contexts to understand better the classification process and the results obtained in this study.

Accuracy: The proportion of correctly predicted outcomes among all positive outcomes. Precision is the ratio of true positive predictions to the total number of instances predicted as positive (true positives and false positives), indicating the accuracy of positive predictions. The recall is computed by dividing the number of true positive (TP) predictions by the total count of actual positive instances (true positives plus false negatives). Precision and Recall are mathematically

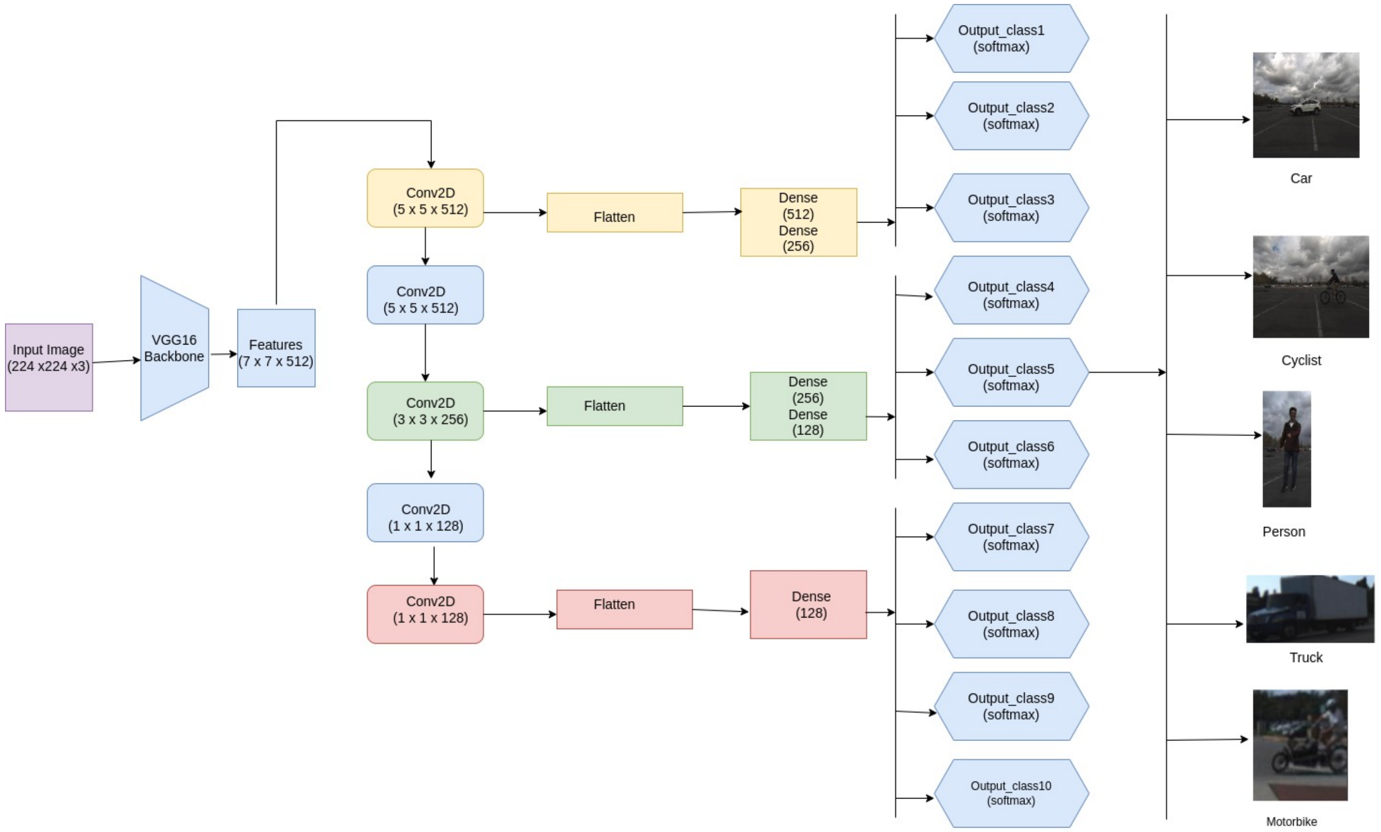

Fig. 1: CNN architecture with multi-layer prediction and VGG backbone

TABLE II: Performance of several state-of-the-art models

Model	Average Accuracy(%)	Average Weighted F1-Score(%)	Trainable Parameters
VGG11	70.278	64.925	8008
VGG13_bn	70.29	64.642	8008
Resnet101x64V2	70.44	64.012	16392
Resnet101V2	70.44	64.08	16392
Resnet50V2	70.44	63.99	16392
Efficientnet_b1	70.41	64.133	10248
Efficientnet_b2	70.44	64.04	11272
Efficientnet_b3	70.42	64.050	12296
Efficientnet_b5	70.44	63.99	16392
Efficientnet_b4	70.44	63.92	14344
Resnet34	70.30	64.48	4104
Resnet101	70.45	64.46	16392
Densenet169	70.18	64.47	8008
Densenet201	70.27	64.56	8008
VGG backbone model	99.46	98.20	$1,14,50,566$

shown in Eq. (1). To assess performance, we use the average precision (AP), average recall (AR), and F1-score. F1-score mathematically shown in Eq. (2), reaches its optimum when precision and recall are equal.

$$Precision = \frac{TP}{TP+FP}, Recall = \frac{TP}{TP+FN} \tag{1}$$

$$F1{-}Score = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN} \tag{2}$$

We first sorted the classifications according to their confidence levels, and we computed the precision and recall for each aggregated classification.

C. Evaluation Results

We Analysed the performance of the proposed model on the Automotive dataset with the 15777, 1973, and 1972 images for training, validation, and testing images. Few sample images are with multiple objects belonging to different classes and few images with a single object. The numbers of objects per class type are 1473, 52, 1177, 14, and 977, for person, truck, car, motorbike, and cyclist classes respectively. The confusion matrix of the VGG backbone model is shown in Fig. 4 and precision-recall values are shown in Table III. The model

TABLE III: Class-wise performance evaluation table for the proposed CNN model

Class	True Positives	False Positives	False Negatives	True Negative	Precision (%)	Recall(%)	F1-Score(%)
Person	1473	2	1	2230	99.86	99.93	99.89
Truck	52	3	2	3649	94.54	96.29	95.41
Car	1177	0	4	2525	100	99.66	99.83
Motorbike	14	1	0	3691	93.33	100	96.55
Cyclist	977	7	6	2716	99.28	99.38	99.33

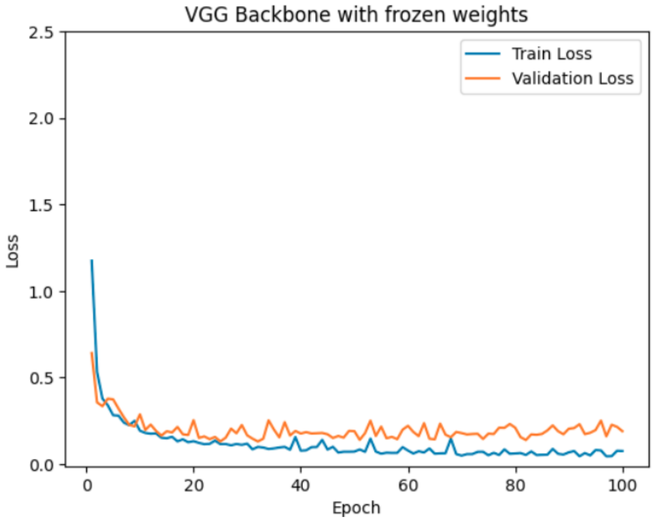

Fig. 2: Loss curves of VGG backbone network with frozen weights

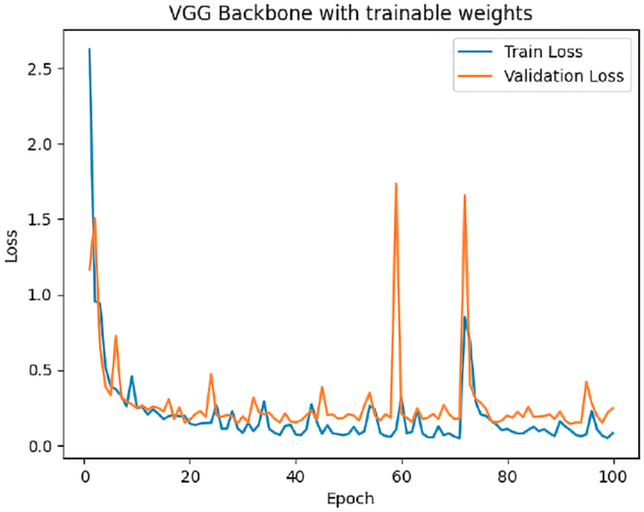

Fig. 3: Loss curves of the VGG backbone network with trainable weights

achieves 99.64% accuracy, and the corresponding accuracy plot is shown in Fig. 5. Table IV illustrates the comprehensive model performance, including class-specific results and the total number of parameters, along with the accuracy score. In addition, we used sparse categorical cross-entropy loss. The sparse categorical cross-entropy loss function is particularly effective when the target labels are presented as integers, as opposed to vectors that have undergone one-hot encoding. The target label, an integer indicating the class, is compared to the projected probability distribution of the model in this loss function.

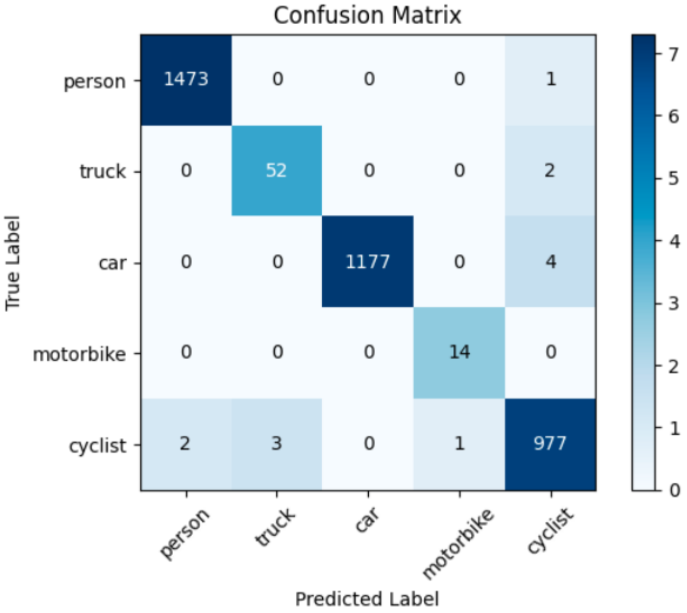

Fig. 4: Confusion matrix of the VGG backbone model with 5x5 class accuracies

The proposed VGG-based classification model achieved higher accuracy than the existing image classification models. The primary reason is that The proposed CNN can learn discriminative features for image classification and can classify objects of different sizes and scales.

The model with the multilayer prediction head excelled in accurate classification at various object scales, providing a significant advantage over the model without the multilayer head. It exhibited faster convergence, a smoother loss versus epoch curve, and more stable training. In contrast, the model without the multilayer prediction head showed irregular loss epoch curves, leading to training instability, reduced generalization performance, slower convergence, and higher sensitivity to hyperparameter tuning. These results underscore the effectiveness of the multilayer prediction head in our model.

VII. Conclusion

In this paper, we have effectively constructed a convolutional neural network model with the purpose of training the automotive image dataset taken from dynamic scenarios. The

TABLE IV: Accuracy evaluation of the proposed model

Model Name	Person	Truck	Car	Motorbike	Cyclist	Total parameters	Trainable params	Non-trainable params	Accuracy(%)
$VGGbackbonemodel$	99.95	99.9	99.87	99.97	99.66	$2,61,65,254$	$1,14,50,566$	$1,47,14,688$	99.64

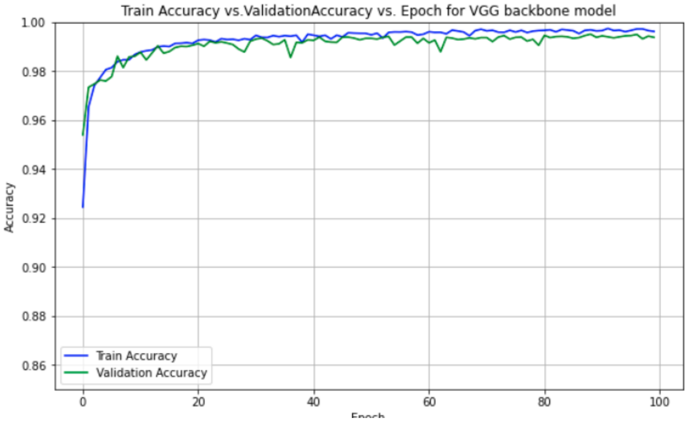

Fig. 5: Accuracy plot for the VGG backbone model

inclusion of the multilayer prediction head in the proposed model leads to better performance. The presence of separate branches in a model with multilayer feature extraction allows for the specialization of parameters for each predicted output. Each branch learns and updates its parameters independently, allowing it to specialize in extracting and predicting relevant specific features of various sizes and scales. This enables it to handle the multi-target classification along with multi-object predictions of varying scales. While pre-trained models are achieving an accuracy of 70%, the proposed custom network with VGG backbone showcases an accuracy of 99.64%. The proposed model underwent rigorous testing with previously unseen data, showcasing better precision.

In summary, the proposed model architecture leverages the feature extractor's strengths by incorporating extra layers and multiple output branches. This enables it to handle multi-class classification and make predictions for objects of varying scales.

Acknowledgment

This work was supported by the Indo-Norwegian Collaboration in Autonomous Cyber-Physical Systems (INCAPS) project: 287918 of the International Partnerships for Excellent Education, Research and Innovation (INTPART) program and the Low-Altitude UAV Communication and Tracking (LUCAT) project: 280835 of the IKTPLUSS program from the Research Council of Norway.

References

[1] 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*. Ieee, 2009, pp. 248–255.

[2] 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.

[3] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[4] 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.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[6] M. Tan and Q. Le, "Efficientnetv2: Smaller models and faster training," in *International conference on machine learning*. PMLR, 2021, pp. 10 096–10 106.

[7] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[8] P. Wang, E. Fan, and P. Wang, "Comparative analysis of image classification algorithms based on traditional machine learning and deep learning," *Pattern Recognition Letters*, vol. 141, pp. 61–67, 2021.

[9] C. Coman *et al.*, "A deep learning sar target classification experiment on mstar dataset," in *2018 19th international radar symposium (IRS)*. IEEE, 2018, pp. 1–6.

[10] A. A. M. Al-Saffar, H. Tao, and M. A. Talab, "Review of deep convolution neural network in image classification," in *2017 International conference on radar, antenna, microwave, electronics, and telecommunications (ICRAMET)*. IEEE, 2017, pp. 26–31.

[11] M. Jogin, Mohana, M. S. Madhulika, G. D. Divya, R. K. Meghana, and S. Apoorva, "Feature extraction using convolution neural networks (cnn) and deep learning," in *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, 2018, pp. 2319–2323.

[12] M. A. Shafaey, M. A.-M. Salem, H. M. Ebied, M. N. Al-Berry, and M. F. Tolba, "Deep learning for satellite image classification," in *International Conference on Advanced Intelligent Systems and Informatics*. Springer, 2018, pp. 383–391.

[13] M. He, B. Li, and H. Chen, "Multi-scale 3d deep convolutional neural network for hyperspectral image classification," in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 3904–3908.

[14] M. P. Muresan, A. R. Barbura, and S. Nedevschi, "Teeth detection and dental problem classification in panoramic x-ray images using deep learning and image processing techniques," in *2020 IEEE 16th International Conference on Intelligent Computer Communication and Processing (ICCP)*, 2020, pp. 457–463.

[15] M. Pritt and G. Chern, "Satellite image classification with deep learning," in *2017 IEEE applied imagery pattern recognition workshop (AIPR)*. IEEE, 2017, pp. 1–7.

[16] W. Liu, M. Zhang, Z. Luo, and Y. Cai, "An ensemble deep learning method for vehicle type classification on visual traffic surveillance sensors," *IEEE Access*, vol. 5, pp. 24 417–24 425, 2017.

[17] Z. Tan, "Vehicle classification with deep learning," *Master's thesis, Fırat University, Institute of Science, Elazığ, 63p*, 2019.

[18] A. J. Joshi, F. Porikli, and N. Papanikolopoulos, "Multi-class active learning for image classification," in *2009 ieee conference on computer vision and pattern recognition*. IEEE, 2009, pp. 2372–2379.

[19] S. Chakraborty, T. Choudhury, R. Sille, C. Dutta, and B. K. Dewangan, "Multi-view deep cnn for automated target recognition and classification of synthetic aperture radar image," *Journal of Advances in Information Technology Vol*, vol. 13, no. 5, 2022.

[20] T.-N. Doan, "Large-scale insect pest image classification," *Journal of Advances in Information Technology*, vol. 14, no. 2, 2023.

[21] J. A. Villaruz, "Deep convolutional neural network feature extraction for berry trees classification," *Journal of Advances in Information Technology Vol*, vol. 12, no. 3, 2021.

[22] X. Gao, Y. Luo, G. Xing, S. Roy, and H. Liu, "Raw adc data of 77ghz mmwave radar for automotive object detection," https://dx.doi.org/10.21227/xm40-jx59, 2022.