International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 03 March 2026
Page No: 1646-1652
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0216

Efficient and Accurate Vehicle Detection for Smart Cities Using Deep Learning

Keshavareddy Nagendra Reddy[1], Poladasu Ramesh Goud[2], Jillella Sahithi[3]
[1,2,3]Dept. of CSE (Data Science), Rajeev Gandhi Memorial College of Engineering and Technology, Nandyal, AndhraPradesh, India – 518501.
Emails: *22091a3287@rgmcet.edu.in[1], 22091a32b6@rgmcet.edu.in[2], 22091a32b9@rgmcet.edu.in[3]*

Abstract
Growing urban populations worldwide have intensified the need for automated, intelligent traffic monitoring solutions capable of operating under demanding real-world conditions. This work proposes a six-class vehicle detection system built upon the YOLO11s deep learning architecture, trained exclusively on Indian road imagery sourced from the Indian Vehicle Dataset (5,000 annotated samples spanning Car, Bus, Truck, Motorcycle, Bicycle, and Auto categories). Training was conducted over 30 epochs on an NVIDIA Tesla T4 accelerator, leveraging AdamW optimisation and Automatic Mixed Precision to maximise convergence speed and hardware efficiency. On the held-out test partition, the system recorded mAP@50 of 0.9936, mAP@50–95 of 0.9650, precision of 0.9980, and recall of 0.9928. End-to-end processing latency measured 8.8 ms per frame, confirming real-time deployment viability for smart city traffic management platforms.

Keywords: *Deep Learning; Indian Traffic; Intelligent Transportation Systems; Smart Cities; Vehicle Detection; YOLO11s*

.

1. Introduction

The growing density of motor vehicles on urban roads presents mounting challenges for city planners, traffic en- gineers, and public safety agencies. In countries such as India, where the vehicle population spans a uniquely wide spectrum—from large commercial trucks and intercity buses to small-displacement motorcycles, pedal bicycles, and three-wheeled auto-rickshaws—designing systems ca- pable of reliably distinguishing every vehicle type is consid- erably more demanding than problems addressed by stan- dard Western benchmarks. Automated vision-based monitoring can process data from dozens of simultaneous camera feeds, flag anomalies within milliseconds, and feed real-time analytics to adaptive signal controllers, density estimation modules, and incident response pipelines. Classical image processing pipelines relying on background subtraction or hand-crafted gradi- ent features degrade sharply under the illumination swings, partial occlusions, and overlapping vehicle silhouettes that routinely occur at busy Indian junctions. Learned feature representations obtained through deep convolutional net- works generalise far more robustly to such variability. Within the family of deep object detectors, the YOLOlineage (Redmon et al., 2016) has emerged as a preferred choice for deployment-oriented research owing to its single-pass inference design, which eliminates the region-proposal bottleneck found in two-stage architectures. Building on this foundation, we select the YOLO11s variant (Ultralytics, 2024) as the detection backbone for this study. The specific contributions of this paper are[1]:

- A fine-tuned YOLO11s model achieving mAP@50 of 0.9936 across six heterogeneous Indian vehicle cate- gories.
- A detailed per-class accuracy breakdown revealing con- sistent high-precision detection even for rare classes such as Bicycle.
- Quantitative comparison against SSD (Liu et al., 2016) and YOLOv7 (Wang et al., 2023), showing improvements in both accuracy and inference throughput.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 03 March 2026
Page No: 1646-1652
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0216

- Analysis of training dynamics, failure modes, and archi- tectural design choices for smart city practitioners [2].

2. Related Work

2.1.Two-Stage Detection Frameworks

Girshick (2015) unified feature extraction and classifica- tion into a single end-to-end trainable network, substantially cutting per-image processing time. Ren et al. (2015) sub-sequently embedded region generation within the network through the Region Proposal Network of Faster R-CNN, enabling near-real-time proposals without sacrificing locali- sation fidelity. Although two-stage frameworks offer strong performance on dense scenes, their sequential propose-then- classify pipeline introduces latency that makes frame rates above 30 FPS challenging without specialised hardware [4].

2.2.Single-Stage Detection Frameworks

Liu et al. (2016) introduced SSD as a direct alternative that simultaneously predicts categories and bounding-box offsets from multiple feature-map scales within a single network pass, raising achievable frame rates into the real- time range. Redmon et al. (2016) advanced the single- stage paradigm further with YOLO, framing detection as a unified spatial regression task. EfficientDet (Tan et al., 2020) proposed a scalable BiFPN neck and compound scal- ing for improved accuracy–efficiency trade-offs. YOLOv4 (Bochkovskiy et al., 2020) and YOLOv7 (Wang et al., 2023) introduced cross-stage partial connections and extended ef- ficient layer aggregation, while YOLOX (Ge et al., 2021) contributed an anchor-free prediction scheme with decou- pled heads that improved generalisation to unseen object scales [5].

2.3.YOLO11s and Indian Traffic Detection

The YOLO11s architecture (Ultralytics, 2024) advances its predecessors via C3k2 backbone blocks, an SPPF module for enlarged receptive fields, and a bidirectional PAN-FPN neck enabling richer multi- scale feature fusion— yielding a compact 9.4 M- parameter model that outperforms heavier predecessors. Public benchmarks such as COCO (Lin et al., 2014) predominantly reflect Western traffic pat- terns. The Indian Vehicle Dataset (DataCluster Labs, 2024) addresses this gap with annotated imagery from urban junc- tions, highways, and night scenarios. Our work explicitly trains and evaluates on this domain-matched resource rather than relying on geographically mismatched transfer [3].

3. Dataset Description

The Indian Vehicle Dataset (DataCluster Labs, 2024) comprises 5,000 annotated images captured across city- centre junctions, suburban arterials, and open highways under varied lighting—from bright midday sun to artificial night-time illumination. Annotations follow the YOLO con- vention with normalised bounding-box coordinates. The dataset is partitioned into training (4,000 images, 80%), val- idation (750 images, 15%), and test (250 images, 5%) splits. Six vehicle categories are annotated: Car, Bus, Truck, Mo- torcycle, Bicycle, and Auto (auto-rickshaw), with moderate class imbalance as shown in Table 1—motorcycles and cars are most frequent while bicycles are rarest, providing a natural generalisation stress test.Key annotation challenges include clusters of partially overlapping vehicles with up to 50% occlusion, low signal- to-noise ratios in night-time frames, and motion blur in high-speed highway sequences—collectively ensuring that a well-performing model generalises beyond sterile evalua- tion conditions shown in Figure 1.

Table 1: Approximate Instance Distribution Across the Dataset

Vehicle Class	Relative Frequency
Car	High
Motorcycle	High
Auto	Medium
Bus	Medium
Truck	Medium
Bicycle	Low

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 03 March 2026
Page No: 1646-1652
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0216

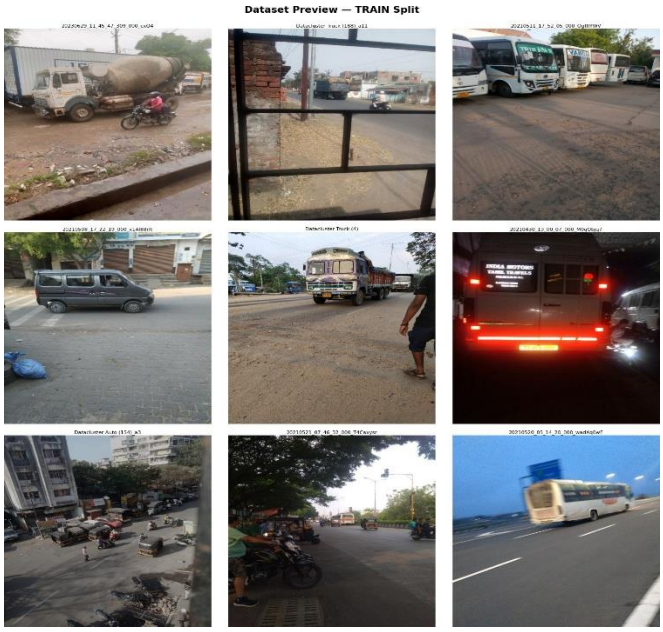

Figure 1 Representative Training Images From The Indian Vehicle Dataset Illustrating Diverse Vehicle Types, Densities, And Lighting Conditions.

4. Methodology

4.1.Yolo11s Network Architecture

The YOLO11s detector processes 640×640-pixel inputs through three functional stages. Feature Extraction Backbone [13]: C3k2 convolutional blocks progressively encode the input into a hierarchy of feature tensors. Shallow layers retain fine-grained spatial detail for localising compact objects such as distant motor- cycles, while deeper layers accumulate semantic informa- tion enabling class discrimination. Feature Aggregation Neck: The PAN-FPN neck per- forms bidirectional lateral feature merging: a top-down pass propagates semantic context downward, and a bottom- up pass returns enhanced spatial information upward. An SPPF module at the backbone–neck junction applies paral- lel max-pooling kernels, extending the effective receptive field without proportional parameter increase.Detection Head [5]: Three parallel prediction branches at strides 8, 16, and 32 pixels (P3, P4, P5) independently predict bounding-box offsets, class log-probabilities, and objectness, specialising each branch for small, medium, and large vehicle scales respectively. The model occupies 101 layers, 9.4 M parameters, and 21.3 GFLOPs per inference pass. Figure 2 illustrates the complete end-to-end detection pipeline.

Figure 2 Yolo11s Detection Pipeline: Indian Vehicle Dataset Input, Data Preprocessing, C3k2 Backbone, Sppf Module, Pan-Fpn Neck, And Three-Scale Detection Head (P3/P4/P5) Producing Bounding Boxes, Class Probabilities, And Objectness Scores.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 03 March 2026
Page No: 1646-1652
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0216

4.2.Objective Function and Evaluation Metrics

The composite training loss is:

$$L_{total} = L_{box} + L_{cls} + L_{dfl} \quad (1)$$

L_{box} is the Complete IoU regression loss penalising centre-point offset, scale mismatch, and aspect-ratio de- viation. L_{cls} is binary cross-entropy per class logit. L_{dfl} is the Distribution Focal Loss (Ultralytics, 2024), modelling each bounding-box edge as a learnable probability distri- bution to capture localisation uncertainty in occluded or motion-blurred boundaries. Standard evaluation metrics:

$$\text{Precision} = \frac{TP}{TP+FP}, \quad \text{Recall} = \frac{TP}{TP+FN}, \quad \text{mAP} = \frac{1}{N}\sum_{i=1} AP_i \quad (2)$$

Weights were initialised from an ImageNet (Deng et al., 2009) pre-trained YOLO11s checkpoint and fine-tuned for 30 epochs:

- Optimiser: AdamW (β1=0.9, β2=0.999, decay= 0.0005)
- Learning rate: lr0=0.001 decayed to lr0×0.01 via co- sine annealing; 3-epoch warmup
- Batch / Resolution: 16 images, 640×640 px
- Early stopping patience: 10 epochs
- NMS IoU / Confidence: 0.45 / 0.25
- Hardware: NVIDIA Tesla T4 (16 GB), AMP enabled
- Augmentation: Mosaic (p=1.0), MixUp (p=0.15), hor- izontal flip (p=0.5), vertical flip (p=0.05), HSV jitter, rotation (±10◦), scale (±50%), translation (±10%)

The rich augmentation pipeline maximises effective train- ing diversity. Mosaic stitches four images into one compos- ite sample, exposing the detector to vehicles at varied scales and occlusion levels per step. MixUp blends image pairs for additional regularisation. Geometric and photometric transforms jointly harden the model against Indian road lighting variability and viewpoint diversity [7].

5. Experimental Results

5.1.Overall Test Set Performance

Table 2 reports metrics on the 250-image test partition. The system attains mAP@50 of 0.9936, confirming near- complete vehicle detection at the standard IoU threshold. The stricter mAP@50–95 of 0.9650 confirms accuracy does not collapse under tighter spatial requirements—essential for lane-occupancy estimation and precise vehicle count- ing. Precision of 0.9980 means fewer than two detections per thousand are spurious, while recall of 0.9928 confirms genuine vehicles are almost never missed [6].

Table 2: Aggregate Performance on the Official Test Set (250 Images)

Metric	Value
mAP@50	0.9936
mAP@50–95	0.9650
Precision	0.9980
Recall	0.9928

5.2.Per-Class Accuracy

Table 3 disaggregates performance across the 750-image validation set. Every class achieves AP@50 ≥ 0.990, con- firming consistent detection across all categories [14]. Bus, Truck, Bicycle, and Auto attain perfect precision and re- call of 1.000. The Car class records the lowest recall at 0.960 due to high instance density and frequent inter- vehicle occlusion. Notably, Bicycle—the rarest class (109 instances, 60 images)—achieves AP@50 = 0.995 with per- fect P&R, validating that the multi-scale PAN-FPN head and mosaic augmentation provide sufficient discriminative capacity even for severely underrepresented categories. The Auto-rickshaw class achieves AP@50 = 0.995, confirming reliable detection of this India-specific vehicle absent from international benchmarks shown in Table 3.

Table 3 Per-Class Detection Performance on the Validation Set (750 Images)

Class	Imgs	Inst.	P	R	AP@50
Car	225	399	0.999	0.960	0.990
Bus	148	233	1.000	1.000	0.995
Truck	499	711	1.000	1.000	0.995
Motorcycle	242	572	0.998	0.983	0.992
Bicycle	60	109	1.000	1.000	0.995

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 03 March 2026
Page No: 1646-1652
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0216

Auto	193	476	0.998	1.000	0.995
Mean	750	2500	0.999	0.990	0.994

5.3.Training Dynamics and F1-Score Stability

Box, classification, and DFL losses all exhibit mono- tonic descent without oscillation, and validation losses track training closely throughout—providing no evidence of over- fitting. The mAP@50 curve saturates around epoch 20. The F1-Confidence curve (Figure 3) peaks at F1 = 0.99 at confidence threshold 0.469, with individual class curves tightly clustered, simplifying threshold selection for real deployments [8] shown in Figure 3.

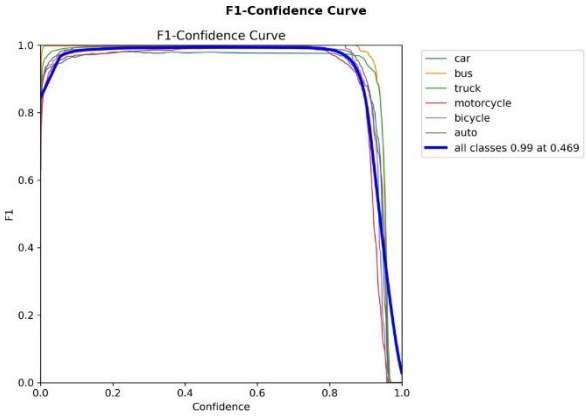

Figure 3 F1-Confidence curve for all six classes. Peak all-class F1 = 0.99 at confidence threshold 0.469.

5.4.Confusion Matrix and Qualitative Results

The normalised confusion matrix (Figure 4) shows strong diagonal dominance across all classes. The only non- trivial off-diagonal entry links Motorcycle and Bicycle— understandable given their shared elongated two-wheeled silhouettes. Background false-positive rates are negligible. Figure 5 presents twelve validation-set predictions span- ning diverse scene types [9]. Multi-vehicle scenes are handled without duplicates or missed detections. Night-time frames present no obstacle—Bus and Car instances are correctly lo- calised under headlight-only illumination. The bottom-right frame confirms simultaneous Truck and dual Auto-rickshaw detection above confidence 0.94, a scene unique to Indian roads.Figure 6 shows detection results on the test split [10], further demonstrating the system's robustness across varied Indian traffic scenarios shown in Figure 4 .

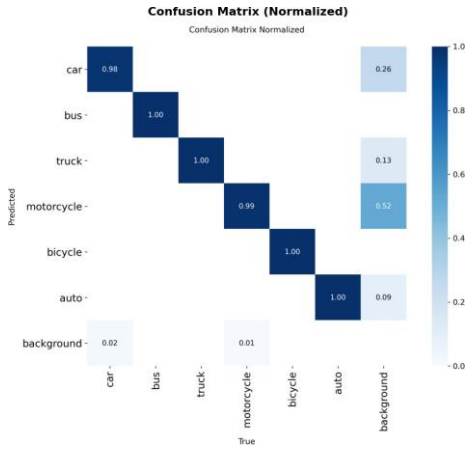

Figure 4 Normalised Confusion Matrix. Dominant Diagonal Entries Confirm Reliable Six-Class Vehicle Discrimination.

FIGURE 5 Qualitative detection results on the validation split. Colour coding: Car (blue), Bus (green), Truck (cyan), Motorcycle (purple), Auto (teal). Confidence scores shown above each box.

5.5. Comparative Evaluation and Inference La- tency

Table 4 benchmarks the proposed system against SSD (Liu et al., 2016) and YOLOv7 (Wang et al., 2023) on the same Indian Vehicle Dataset. SSD trails by 8.4

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 03 March 2026
Page No: 1646-1652
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0216

per- centage points in mAP@50: its fixed-aspect-ratio anchor scheme is not optimised for the elongated bounding boxes of two-wheelers and auto-rickshaws [15], and its unidirectional feature merging fails to propagate semantic context to fine- resolution maps where small vehicles are most detectable. YOLOv7 narrows the gap to 2.4 percentage points via its E-ELAN backbone and auxiliary training heads, but at the cost of roughly double the inference time. YOLO11s achieves superior accuracy with a leaner 9.4 M-parameter model through bidirectional PAN-FPN fusion and DFL- based boundary regression [11]. Per-image latency on the Tesla T4: pre-processing 0.2 ms, network forward pass 5.2 ms, NMS post-processing 3.4 ms shown in Figure 4.

Figure 6 Qualitative detection results on the test split covering day and night scenes with multiple overlapping vehicle classes.

Figure 7 Yolov11 Inference On A Previously Unseen Image Detecting Four Vehicles (Truck, Car, Auto, Motorcycle) With Confidence Scores Between 0.34 And 0.94.

— total ≈8.8 ms (≈113 FPS). At 113 FPS the system op- erates at over four times the 25 FPS threshold for smooth video monitoring, providing substantial headroom for multi- camera smart city deployments [12].

6. Discussion

The results validate that domain-specific training is in- dispensable for high-accuracy vehicle detection on Indian roads. A model fine-tuned on geographically matched imagery readily learns the distinctive proportions of auto- rickshaws, the prevalence of two-wheelers in tight clusters, and night-time lighting gradients—contextual knowledge that cross-domain transfer from COCO-trained weights can- not reliably supply. The PAN-FPN neck emerges as the single most impact- ful architectural component: Indian traffic scenes routinely span vehicles from motorcycles occupying 30% of the im- age to distant buses subtending less than 2%, and bidirec- tional feature fusion equips each prediction branch with both spatial precision and semantic depth simultaneously.

The residual Motorcycle–Bicycle confusion is expected given their shared silhouettes and can be addressed through temporal tracking or targeted dataset augmentation. The compact 9.4 M-parameter footprint positions YOLO11s within the operational envelope of embedded accelerators; INT8 post-training quantisation would roughly halve la- tency, enabling deployment on NVIDIA Jetson Nano-class edge hardware for cost-effective large-scale camera network rollouts.

Table 4 Detection Accuracy and Speed Comparison On The Indian Vehicle Dataset

Model	mAP@50	Inference (ms)
SSD (Liu et al., 2016)	0.910	~15
YOLOv7 (Wang et al., 2023)	0.970	~12
Proposed YOLO11s	**0.9936**	**5.2**

Conclusion

This study demonstrates that YOLO11s, trained on domain-specific Indian road imagery, achieves exceptional multi-class vehicle detection with mAP@50 = 0.9936, preci- sion = 0.9980, and recall = 0.9928. End-to-end inference at 8. 8 ms per frame

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 03 March 2026
Page No: 1646-1652
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0216

(≈113 FPS) far exceeds real-time require- ments. Systematic comparison against SSD and YOLOv7 confirms meaningful advances in both accuracy and speed. Future work will extend the framework to multi-object video tracking, integrate Automatic Number Plate Recogni- tion, and explore model compression for edge deployment.

Acknowledgement

The authors express their sincere gratitude to their project guide Mr. Y. P. Srinath Reddy, Assistant Profes- sor, Department of Computer Science and Engineering (Data Science), Rajeev Gandhi Memorial College of Engi- neering and Technology, Nandyal, Andhra Pradesh, India (srinathreddycseds@rgmcet.edu.in), for his in- valuable guidance, continuous encouragement, and con- structive feedback throughout every phase of this research.

References

[1]. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Ob- ject Detection. Proceedings of IEEE CVPR, Las Vegas, NV, USA, pp. 779–788.

[2]. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). SSD: Single Shot MultiBox Detector. Proceedings of ECCV, Amsterdam, Netherlands, pp. 21–37.

[3]. Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R- CNN: Towards Real-Time Object Detection with Region Proposal Networks. Proceedings of NeurIPS, Montreal, Canada, pp. 91–99.

[4]. Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). YOLOv4: Optimal Speed and Accuracy of Object De- tection. arXiv:2004.10934.

[5]. Wang, C.-Y., Bochkovskiy, A., & Liao, H.-Y. M. (2023). YOLOv7: Trainable Bag-of-Freebies Sets New State-of- the-Art for Real-Time Object Detectors. Proceedings of IEEE CVPR, Vancouver, Canada, pp. 7464–7475.

[6]. Ge, Z., Liu, S., Wang, F., Li, Z., & Sun, J. (2021). YOLOX: Exceeding YOLO Series in 2021. arXiv:2107.08430.

[7]. Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona,P., Ramanan, D., Dolla´r, P., & Zitnick, C. L. (2014). Microsoft COCO: Common Objects in Context. Pro- ceedings of ECCV, Zurich, Switzerland, pp. 740–755.

[8]. Tan, M., Pang, R., & Le, Q. V. (2020). EfficientDet: Scalable and Efficient Object Detection. Proceedings of IEEE CVPR, Seattle, WA, USA, pp. 10781–10790.

[9]. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei- Fei, L. (2009). ImageNet: A Large-Scale Hierarchical Image Database. Proceedings of IEEE CVPR, Miami, FL, USA, pp. 248–255.

[10]. Girshick, R. (2015). Fast R-CNN. Proceedings of IEEE ICCV, Santiago, Chile, pp. 1440–1448.

[11]. Ultralytics. (2024). YOLO11: Ultralytics YOLO11 Documentation. Retrieved from https://docs. ultralytics.com

[12]. DataCluster Labs. (2024). Indian Vehicle Dataset. Kaggle. Retrieved from https://kaggle.com/datasets/dataclusterlabs/ indian-vehicle-dataset

[13]. Jocher, G., Chaurasia, A., & Qiu, J. (2023). Ultralytics YOLO. GitHub. Retrieved from https://github. com/ultralytics/ultralytics

[14]. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. Proceedings of IEEE CVPR, Las Vegas, NV, USA, pp. 770–778.

[15]. Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., Le, Q. V., & Adam, H. (2019). Searching for Mo- bileNetV3. Proceedings of IEEE ICCV, Seoul, South Korea, pp. 1314–1324.