JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

T-YOLO: Tiny Vehicle Detection Based on YOLO and Multi-Scale Convolutional Neural Networks

[1]Gottemukula Hadassah Mercy, [2]Maggidi Prathyusha, [3]Pendoti Kavya, [4]Mr.Monis Tariq
[1,2,3]UG Scholars, [4]Asst.Professor
[1,2,3,4] Department of Computer Science Engineering (Internet of Things)
Guru Nanak Institutions Technical Campus (Autonomous), Hyderabad, India

Abstract : Addressing real-world challenges in various smart city applications, such as parking occupancy detection, necessitates fine-tuning deep Neural Networks. For expansive parking areas, utilizing a high-placed cenital plane camera facilitates comprehensive monitoring with a single device. Leading object detection models like YOLO offer commendable precision and real-time speed, yet leveraging proprietary data provides significant room for customization beyond general-purpose datasets like COCO and ImageNet. This study proposes a modified, lightweight deep object detection model based on the YOLO-v5 architecture, capable of detecting objects of all sizes. Specifically, a multi-scale mechanism is introduced to learn discriminative features across different scales and automatically select the optimal scales for object detection, particularly vehicles. This multi-scale module reduces the number of trainable parameters compared to the original YOLO-v5 architecture. Experimental results demonstrate a substantial improvement in precision, with only a minor reduction in parameters from the YOLO-v5-S profile to our model. Moreover, the detection speed is reduced by inferring 30 fps compared to YOLO-v5-L/X profiles, while the performance in detecting tiny vehicles sees a significant 33% enhancement compared to the YOLO-v5-X profile.

***IndexTerms* -** Real-world challenges, Parking occupancy detection, Fine-tuning, Deep Neural Networks, High-placed cenital plane camera , Comprehensive monitoring, Object detection models, YOLO, Proprietary data, General-purpose datasets, COCO, ImageNet Modified model, Lightweight, YOLO-v5 architecture, Multi-scale mechanism, Optimal scales, Vehicle detection, Improvement Detection speed, Tiny vehicles

I. INTRODUCTION

The burgeoning urban population has escalated the urgency of managing city resources, prompting the emergence of smart cities to exploit resource data effectively. Enhancing driving experiences, traffic control, surveillance, and parking guidance pose significant challenges in large urban areas, crucial for improving overall mobility. Addressing the issue of parking spot availability, which notably impacts drivers' time and contributes to environmental pollution due to extended search distances, is paramount. Traditional sensor-based methods face limitations such as battery life depletion and compatibility issues with modern vehicles, prompting exploration into computer vision and deep learning for cost-effective solutions using smart cameras. While previous studies have attempted parking spot detection based on visual data, challenges persist in generalization and adaptation across different parking lots. Most existing techniques focus on spot classification rather than vehicle localization, thereby missing out on potential information like road congestion and human interactions with vehicles. Vehicle detection methods offer a more comprehensive approach, enabling the extraction of crucial data that spot classification techniques cannot provide.

II. LITERATURE SURVEY

S. C. K. Tekouabou, E. A. A. Alaoui, W. Cherif, and H. Silkan present "Impactful research in 'Improving parking availability prediction in smart cities with IoT and ensemble-based model (2022) addresses urban traffic congestion and parking challenges. They proposes an innovative system integrating IoT and ensemble modeling to predict parking space availability in smart cities.This underscores the potential of IoT integration and ensemble methods in optimizing parking prediction, promising improved urban mobility and quality of life."

M. A. Merzoug, A. Mostefaoui, G. Gianini, and E. Damiani provide insights into "Smart connected parking lots based on secured multimedia IoT devices" (2021) introduces an innovative IoT-based solution for automating parking spot counting and driver notification in major cities. Experimentation with classical and lightweight deep learning techniques validates the system's efficiency, ease of deployment, and suitability for resource-constrained embedded systems, highlighting the effectiveness of lightweight solutions in terms of inference time, size, and accuracy.

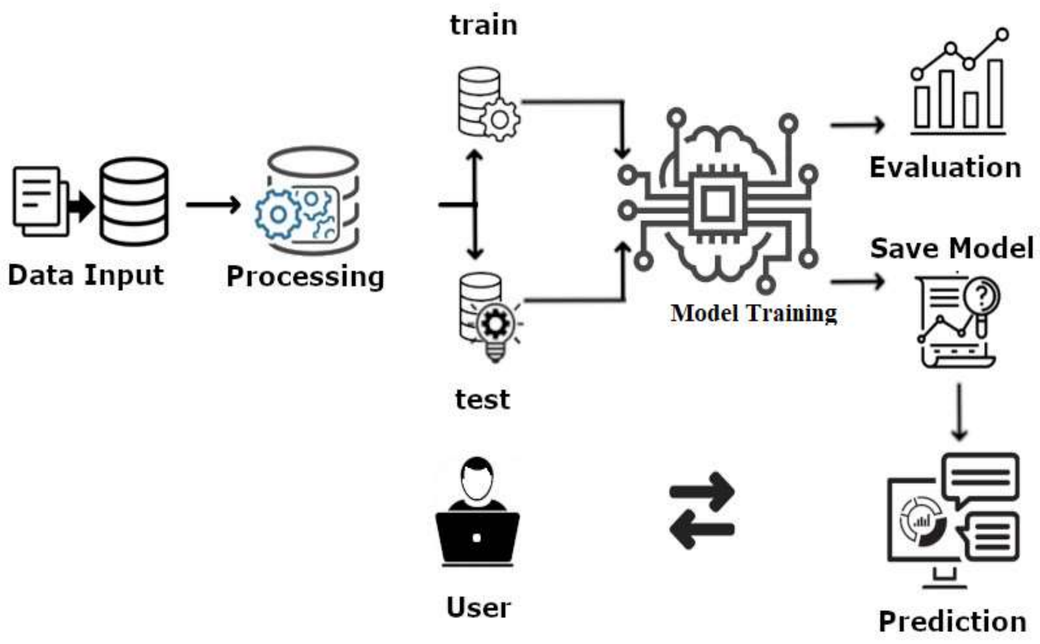
M. dos Santos de Arruda, L. P. Osco, P. R. Acosta, D. N. Gonçalves, J. M. Junior, A. P. M. Ramos, E. T. Matsubara, Z. Luo, J. Li, J. de Andrade Silva, and W. N. Gonçalves propose "Counting and locating high-density objects using convolutional neural network". (2021) introduces a novel Convolutional Neural Network (CNN) approach for counting and locating objects in dense imagery. By leveraging feature map enhancement and Multi-Stage Refinement of the confidence map, the proposed method achieves superior performance in object counting and localization. This innovative method exhibits great potential for accurately counting and locating objects in high-density environments.

M. Farag, M. Din, and H. Elshenbary focus on The paper "Deep learning versus traditional methods for parking lots occupancy classification" (2020) that compares Convolutional Neural Network (CNN) approaches with traditional methods for parking lot occupancy classification. By leveraging feature map enhancement and Multi-Stage Refinement of the confidence map, the CNN method demonstrates superior performance in object counting and locating.

III. Proposed System

The YOLO-v5 model is designed for efficient detection with a focus on minimizing parameters and achieving high frames-per-second (fps) inference suitable for low-end terminals. It's crucial to ensure that any modifications to the model do not significantly impact this aspect. The YOLO network consists of numerous interconnected layers, which can be categorized into three main sections. The first section, known as the backbone (referred to as cspdarknet in YOLO-v5), comprises typical CNN operations such as convolutions, concatenations, and max-pooling, along with a straightforward forwarding mechanism to extract multiple features for subsequent sections. The backbone concept is widely employed across various deep learning networks for object detection and serves as a fundamental base network.

IV. System Architecture



V. HARDWARE REQUIREMENTS

The hardware requirements form the foundation for a contract outlining the system implementation and must thus provide a comprehensive and cohesive specification of the entire system. They serve as the initial reference point for software engineers in designing the system, focusing on defining what the system should accomplish rather than dictating its implementation details.

- Processor: Dual Core 2 Duos.
- RAM: 4GB DD RAM
- Hard Disk: 250 GB

VI. SOFTWARE REQUIREMENTS

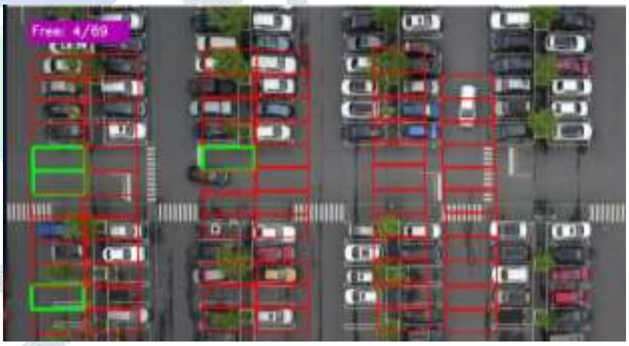
The software requirements document serves as the system's specification, encompassing both the definition and specification of requirements. It outlines what the system should accomplish rather than how it should achieve it. These requirements form the foundation for creating the software requirements specification. They play a vital role in estimating costs, planning team activities, executing tasks, and monitoring the team's progress throughout the development process.

- Operating System : Windows 7/8/10
- Platform : Spyder3
- Programming Language : Python
- Front End : Spyder3

VII. FUTURE ENHANCEMENT

In the realm of T-YOLO: Tiny Vehicle Detection based on YOLO and Multi-Scale Convolutional Neural Networks, future enhancements are poised to propel the system's capabilities to new heights. Foremost among these is the pursuit of improved accuracy, refining the model to excel in detecting vehicles even amidst challenging scenarios such as occlusions or low-light conditions. Moreover, optimization efforts will target real-time performance, ensuring swift and seamless vehicle detection for timely decision-making. Robustness to environmental changes will be prioritized, with enhancements aimed at fortifying the model against varying weather conditions, lighting fluctuations, and occlusions. Multi-object tracking capabilities will be integrated to enable the tracking of vehicles across consecutive frames, offering valuable insights for traffic flow analysis and surveillance. Adaptability to different domains will be emphasized, broadening the model's applicability across diverse environments and scenarios. Furthermore, efforts will be directed towards hardware efficiency, ensuring that the model can run efficiently on resource-constrained devices without compromising performance. Integration with IoT and smart city systems will unlock new avenues for leveraging additional data sources and enhancing overall functionality for traffic management and urban planning. Semantic segmentation capabilities will be incorporated to precisely delineate vehicle boundaries and improve detection accuracy, particularly in crowded or complex scenes. Continuous learning mechanisms will enable the model to adapt to evolving traffic patterns and scenarios over time, ensuring sustained performance improvements. Finally, user-friendly interfaces and visualization tools will be developed to facilitate easy deployment, monitoring, and management of the T-YOLO system by end-users and administrators.

VIII. SNAPSHOTS



IX. CONCLUSION

In conclusion, T-YOLO: Tiny Vehicle Detection based on YOLO and Multi-Scale Convolutional Neural Networks represents a significant advancement in the field of vehicle detection systems. By leveraging the YOLO architecture and incorporating multi-scale convolutional neural networks, T-YOLO achieves impressive accuracy and efficiency in detecting vehicles of varying sizes in diverse environmental conditions. Its ability to accurately identify vehicles in real-time, while consuming minimal computational resources, makes it a valuable tool for a wide range of applications, from traffic management to surveillance and urban planning. Continued research and development efforts aimed at enhancing T-YOLO's performance, adaptability, and usability hold promise for further revolutionizing the field of vehicle detection and contributing to the advancement of smart city technologies.

X. REFERENCES

[1] S. B. Atitallah, M. Driss, W. Boulila, and H. B. Ghézala, ‘‘Leveraging deep learning and IoT big data analytics to support the smart cities development: Review and future directions,’’ Comput.

[2] T. Lin, H. Rivano, and F. Le Mouël, ‘‘A survey of smart parking solutions,’’ IEEE Trans. Intell. Transp. Syst., vol. 18, no. 12, pp. 3229–3253, Dec. 2017.

[3] S. J. Faraji and M. J. Naozar, ‘‘Smart parking: An efficient approach to city’s smart management and air pollution reduction,’’ J. Air Pollut. Health, vol. 4, no. 1, pp. 53–72, Feb. 2019.

[4] O. Cats, C. Zhang, and A. Nissan, ‘‘Survey methodology for measuring parking occupancy: Impacts of an on-street parking pricing scheme in an urban center,’’ Transp. Policy, vol. 47, pp. 55–63, Apr. 2016.

[5] G. Amato, F. Carrara, F. Falchi, C. Gennaro, C. Meghini, and C. Vairo, ‘‘Deep learning for decentralized parking lot occupancy detection,’’ Expert Syst. Appl., vol. 72, pp. 327–334, Apr. 2017.
[6] S. C. K. Tekouabou, E. A. A. Alaoui, W. Cherif, and H. Silkan, ‘‘Improving parking availability prediction in smart cities with IoT and ensemble-based model,’’ J. King Saud Univ.-Comput. Inf. Sci., vol. 34, no. 3, pp. 687–697, Mar. 2022.
[7] M. A. Merzoug, A. Mostefaoui, G. Gianini, and E. Damiani, ‘‘Smart connected parking lots based on secured multimedia IoT devices,’’ Computing, vol. 103, no. 6, pp. 1143–1164, Jun. 2021.
[8] M. dos Santos de Arruda, L. P. Osco, P. R. Acosta, D. N. Gonçalves, J. M. Junior, A. P. M. Ramos, E. T. Matsubara, Z. Luo, J. Li, J. de Andrade Silva, and W. N. Gonçalves, ‘‘Counting and locating high-density objects using convolutional neural network,’’ 2021, arXiv:2102.04366.
[9] M. Farag, M. Din, and H. Elshenbary, ‘‘Deep learning versus traditional methods for parking lots occupancy classification,’’ Indonesian J. Elect. Eng. Comput. Sci., vol. 19, pp. 964–973, Aug. 2020.
[10] S. Nurullayev and S.-W. Lee, ‘‘Generalized parking occupancy analysis based on dilated convolutional neural network,’’ Sensors, vol. 19, no. 2, p. 277, Jan. 2019.
[11] A. Farley, H. Ham, and Hendra, ‘‘Real time IP camera parking occupancy detection using deep learning,’’ Proc. Comput. Sci., vol. 179, pp. 606–614, Jan. 2021.
[12] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, ‘‘Microsoft COCO: Common objects in context,’’ 2014, arXiv:1405.0312.
[13] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, ‘‘ImageNet: A large-scale hierarchical image database,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2009, pp. 248–255.
[14] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, ‘‘The PASCAL visual object classes (VOC) challenge,’’ Int. J. Comput. Vis., vol. 88, no. 2, pp. 303–338, Jun. 2010, doi: 10.1007/ s11263-009-0275-4.
[15] S. Ren, K. He, R. Girshick, and J. Sun, ‘‘Faster R-CNN: Towards real-time object detection with region proposal networks,’’ 2015, arXiv:1506.01497.
[16] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, ‘‘SSD: Single shot MultiBox detector,’’ in Computer Vision— ECCV, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham, Switzerland: Springer, 2016, pp. 21–37.
[17] J. Redmon and A. Farhadi, ‘‘YOLOv3: An incremental improvement,’’ 2018, arXiv:1804.02767.
[18] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, ‘‘YOLOv4: Optimal speed and accuracy of object detection,’’ 2020, arXiv:2004.10934.
[19] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, ‘‘Scaled-YOLOv4: Scaling cross stage partial network,’’ 2020, arXiv:2011.08036.
[20] G. Jocher, A. Stoken, J. Borovec, L. Changyu, A. Hogan, A. Chaurasia, L. Diaconu, Doug, Durgesh, F. Ingham, Frederik, Guilhen, A. Colmagro, H. Ye, Jacobsolawetz, J. Poznanski, J. Fang, J. Kim, K. Doan, and L. Yu, ‘‘Ultralytics/yolov5: V4.0—Nn.SiLU() activations, weights & biases logging, PyTorch hub integration,’’ Ultralytics, Los Angeles, CA, USA, Jan. 2021, doi: 10.5281/zenodo.4418161.
[21] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, ‘‘Feature pyramid networks for object detection,’’ 2016, arXiv:1612.03144.
[22] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, ‘‘ENet: A deep neural network architecture for real-time semantic segmentation,’’ 2016, arXiv:1606.02147.
[23] C. Kumar B., R. Punitha, and Mohana, ‘‘YOLOv3 and YOLOv4: Multiple object detection for surveillance applications,’’ in Proc. 3rd Int. Conf. Smart Syst. Inventive Technol. (ICSSIT), Aug. 2020, pp.
[24] K. Simonyan and A. Zisserman, ‘‘Very deep convolutional networks for large-scale image recognition,’’ 2014, arXiv:1409.1556.
[25] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, ‘‘Path aggregation network for instance segmentation,’’ in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 8759–8768.
[26] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, ‘‘Dual attention network for scene segmentation,’’ 2019, arXiv:1809.02983.
[27] T. Ridnik, H. Lawen, A. Noy, E. B. Baruch, G. Sharir, and I. Friedman, ‘‘TResNet: High performance GPU-dedicated architecture,’’ 2020, arXiv:2003.13630.
[28] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, ‘‘Mixup: Beyond empirical risk minimization,’’ 2017, arXiv:1710.09412.