ROAD SCENES UNDERSTANDING IN AUTONOMOUS DRIVING

B C Sai Bhanu Kiran*, Jalluri Uday†, Rayanki Moksha Pranavi‡
*Department of CSE, VIT-AP (Amaravati), kiran.22bce8310@vitapstudent.ac.in
†Department of CSE, VIT-AP (Amaravati), uday.22bce8177@vitapstudent.ac.in
‡Department of CSE, VIT-AP (Amaravati), pranavi.22bce7273@vitapstudent.ac.in

***Abstract*—Object detection is a key application in computer vision, especially in autonomous driving and surveillance. This paper compares pre-trained models, YOLOv3 and YOLOv5, with a custom model pipeline. The custom model aims to combine speed, accuracy, and advanced visualization techniques to improve object detection performance. The results indicate that the custom pipeline outperforms YOLOv3 in complex scenarios and offers enhanced accuracy, particularly for small and occluded objects.**

***Index Terms*—Object Detection, Convolutional Neural Networks (CNNs), YOLO, YOLOv3, YOLOv5, YOLOv8, Faster R-CNN, Feature Extraction, Real-Time Systems, Computer Vision, Visualization Techniques, Transfer Learning, Intersection over Union (IoU).**

I. Introduction

Object detection is pivotal in computer vision applications like autonomous Object detection is a rapidly expanding field across various domains, including autonomous driving, surveillance, and real-time analytics. Its importance has driven the development of ever-faster and more accurate object detection models with advancing technologies. This report reviews the potentials of two widely used pre-trained models, YOLOv3 and YOLOv5, and proposes a more advanced custom model pipeline designed to upgrade the pipeline of detection efficiency and accuracy. This custom pipeline gathers various state-of-the-art detection techniques and does involve advanced visualization tools to offer a full performance comparison.

II. Literature Survey

Object detection has been one of the core research areas in computer vision and has led to quite a lot of influence in what occurs in autonomous driving, surveillance, as well as real-time analytics. This shift of traditional methods toward advanced machine learning models depicts the measures by which precision, speed, and adaptability have been improved in detecting complex scenarios.

One of the prominent improvements is made through integration with pre-trained models, such as YOLO (You Only Look Once) and Faster R-CNN. Balancing speed and accuracy of general-purpose object detection tasks, YOLOv3, by Redmon and Farhadi, used a grid-based prediction system to enable real-time object detection but left a lot to be desired concerning detecting smaller and occluded objects in crowded environments.

Building upon YOLOv3, YOLOv5 improved further on efficiency while increasing the accuracy by utilizing the improvements in the amplitude in the layers for the feature extraction that increase processing times and reduce the overhead of computation. This makes YOLOv5 a preferred framework for real-time application as it is optimized architecture with compact size.

Further improvements were done with the YOLOv8 which provided state-of-the-art backbone networks as well as achieved the highest accuracy of detections within the YOLO family. It is centered on more powerful detection on small objects and even overlapping ones, which makes it well fitting for complex real-world scenarios.

Ren et al. proposed the region-based detection approach called Faster R-CNN, which improves the accuracy of object detection in dense and occluded environments. Its two-stage approach, region pro-

posal and classification, enables high-quality localization of objects but performs slower inference speeds compared to the YOLO models. This is suitable for application use cases that require detection of smaller and partially occluded objects more accurately.

A. Model Architectures and Evaluations on MSCOCO Dataset

Model	Architecture	Evaluation	Comment
You (2016)	CNN+RNN	BLEU-4: 30.4, METEOR: 24.3	Visual attributes.
Fu (2017)	VGG/ResNet	BLEU-4: 31.3, METEOR: 24.8, CIDEr: 53.2	Region+scene attention.
Dai (2017)	GAN	CIDEr: 84.6, SPICE: 1.027	G-GAN aligns well with human eval.
Aneja (2018)	ResNet152	BLEU-4: 29.9, CIDEr: 97.2	Improved ResNet encoding.
Luo (2018)	ResNet101	BLEU-4: 37.2, CIDEr: 102.3	Discriminability loss.
Anderson (2018)	Faster R-CNN+LSTM	BLEU-4: 36.5, CIDEr: 117.9	VQA 2017 winner.
Yao (2019)	GCN+LSTM	BLEU-4: 30.3, CIDEr: 122.1	Graph-based connections.
Nezami (2019)	ATTEND-GAN	BLEU-4: 10.5, CIDEr: 62.85	Semantic attention.
Ding (2020)	VGG19+LSTM	BLEU-4: 35.5, CIDEr: 992.4	Psychological attention.
Cornia (2020)	Faster R-CNN+LSTM	BLEU-4: 35.5, CIDEr: 992.4	Memory-augmented layers.
Pan (2020)	SENE+154	BLEU-4: 35.4, CIDEr: 902.5	X-Linear attention.

Recently, there were also attempts to use pre-trained models with custom pipelines. Custom pipelines involve a variety of architectures. From each architecture, a particular strength is taken advantage of. For example,

YOLO Models provide high speed along with real-time adaptability Faster R-CNN offers accuracy in cases of occlusion and cluttered layout of objects.

CNN-based layers include Conv2D and Max-Pooling2D, that strongly draws meaningful features from the images, therefore enhancing the detection accuracy of so many objects.

Other key areas of research are the pre-processings and visualizations of data. Input preparation is also performed much more efficiently through the rescaling of images, normalizing pixel values, and padding bounding boxes. The more complex visualization schemes include bounding box visualization, heatmaps, confidence plots, and animations. This way, it offers intuitive insights into the performance of the detection as well as the areas in which improvement is necessary in the model.

Real-world challenges still exist: • High detection accuracy with varying conditions, for example, low illumination, changes in the weather, and occlusions due to objects.

• Low latency to be operative in real-time for dynamic adaptation in situations like autonomous driving.

• Improvement of detection of small and overlapping objects, which are typically missed by traditional models.

Future research directions in object detection include incorporating transformer-based architectures and learning semi-supervised. These techniques will improve generalization capabilities of detection models by reducing their dependency upon labeled data. The multimodal inputs can be images and contextual metadata about improving the accuracy of detection and adaptation in complex environments.

III. Problem Statement

Object detection has posed many critical challenges in the context of automotive autonomous driving and surveillance:

- **High Detection Accuracy**: The Models should detect a host of objects of wide illumination, occlusion, and weather conditions.

- **Real-time Adaptability**: The application should ensure minimal latency to satisfy applications such as real-time analytics, involving a prompt response and ensuing decision.
- **Handling Occlusions and Overlap**: Handling Occlusions and Overlap Sometimes, objects overlap and create occlusions that reduce accuracy. Non-overlapping models have difficulty distinguishing the objects, especially when densely populated environments exist.
- **Detection of Small Objects**: In road scenarios, small objects are very important to be detected, whereas traditional models fail to detect them due to low resolution or scaling.

IV. Methodology

Object detection plays a central role in many applications, especially autonomous driving, in identifying and localizing objects within images and visual scenes. Using advanced deep learning methods, object detection methodologies search for a balance among accuracy, speed, and adaptability. This chapter describes the parts and stages of a strong object detection pipeline.

A. Image Preprocessing

One of the most important methodologies in developing this pipeline is image preprocessing, where images become unified and optimized for training purposes, with pixel values normalized to be in the interval 0 to 1. The uniform dimension here is 128x128 pixels. Normalization really helps the model to really run more efficiently because it brings down variability in the datasets. Padding techniques are also used to ensure uniform bounding box dimensions. In such a case, missing boxes are filled with a default value for example -1. It then takes care of compatibility during batch processing and training.

B. Feature Extraction Using Convolutional Neural Networks (CNNs)

Convolutional Neural Networks are the base components of any pipeline for object detection. For this purpose, some popular architectures include VGG and ResNet, which have been pre-trained on huge datasets, like that of ImageNet. Hierarchical visual features, such as edges, textures, and patterns, extracted by models, are critical for the identification of objects.

- **YOLO Models**:YOLO models, like YOLOv3, YOLOv5, and YOLOv8, are really great real-time detectors, utilizing grid-based prediction techniques. Its latest version, YOLOv8, which is a new step forward, incorporating advanced backbone networks, deals with high accuracy by focusing specially on the detection of small and occluded objects.
- **Faster R-CNN**: Unlike YOLO, Faster R-CNN uses a two-stage detection model. It uses the first stage to propose regions of interest and then classifies the regions with a high precision in the second stage. Faster R-CNN provides detection accuracy in cluttered scenes, especially with small or overlapping objects.

C. Sequence Modeling for Object Localization

It is necessary to use sequence modeling when data need to be processed in the form of a sequence. An example is tracking the movement of an object from one video frame to another. Long Short-Term Memory (LSTM) networks are an extension of Recurrent Neural Networks and are robust in handling long term dependencies in temporal data. Because of the internal memory in LSTMs, sequences of images or feature images can be analyzed while remembering some contextual information.

D. High-Level Visualization Techniques

Visualization is the other key role in understanding the performance of a model. The following techniques are used to analyze and interpret detection results:

- **Bounding Box Visualization**:Objects are detected with color-coded bounding boxes that represent the model's confidence levels.
- **Heatmaps**: Heatmaps reveal areas with the presence of high density object detection, thus providing insights into regions of focus during inference.
- **Confidence Plots**: Confidence plots show trends of confidence scores, giving a statistical view about detection reliability.

- **Animation**: For video feeds, animated visualizations track object detection consistency across frames, allowing it to be evaluated.

E. Model Training and Optimization

During training, the objective is minimizing the loss between the predicted and ground truth bounding boxes and object classes. Usual techniques employed include the following:

- **IoU (Intersection over Union)**: IoU defines overlap between the predicted bounding box and the actual ground-truth bounding box as an important metric for localization accuracy.
- **Data Augmentation**: The techniques involve flipping, rotation, and scaling, aimed at increasing the diversity and robustness of the dataset.
- **Transfer Learning**: ImageNet pre-trained weight is fine-tuned on domain-specific data and the training time is drastically reduced and the model generalization improved.

F. Inference and Post-Processing

Among these are efficient post-processing strategies to support real-time adaptability during inference:

- **Non-Maximum Suppression (NMS)**: It actually removes duplicate bounding boxes and leaves only the boxes with the highest confidence scores.
- **Beam Search**: For sequential predictions, beam search will examine numerous candidate outputs and pick the one having the highest probability.

G. Custom Model Pipeline

The custom pipeline combines the speed and real-time adaptability of YOLO models with the precision of Faster R-CNN. This hybrid architecture strikes the balance between accuracy and latency, and the model is thoroughly suitable for deployment on hardware in real-world applications, such as autonomous driving. Also layered with CNN-based layers, such as Conv2D and MaxPooling2D, this amplifies feature extraction. Advanced visualization tools complement the pipeline with insights which can be acted upon.

H. Challenges and Improvements

Much has been achieved, but challenges are even in illumination variability, occlusions, and difficulty in the detection of small objects. A possible direction for future research could be in the incorporation of transformer-based architectures and semi-supervised learning methods to approach these limitations of working.

V. Model Architecture

The proposed architecture of object detection builds up from pre-trained Convolutional Neural Networks strengths and then integrates additional techniques for a balance between the two objectives, namely accuracy and efficiency. In designing this architecture, attempts would focus on exploring solutions to several challenges in object detection that include dealing with small and occluded objects, as well as achieving real-time performance.

A. Feature Extraction with Pre-Trained Models

Architectures used for pre-trained CNNs include YOLOv3, YOLOv5, and Faster R-CNN to extract high-level features fed with input: In the architecture, Networks proved themselves to be highly effective in capturing small and subtle patterns and spatial relationships in an image:

- **YOLOv3, YOLOv5, and YOLOv8**: These models operate on grids and make use of prediction mechanisms that can immediately detect fast objects. The more advanced model in the family is YOLOv8, which brings together high-backbone networks along with the capacity for producing results even when small overlapping objects are concerned.
- **Faster R-CNN**: It is region-based and captures two stages of the pipeline in detection. First of all, it generates the region of interest by making use of Region Proposal Network called RPN and then develops classification and localization. This makes it very efficient for scenes which are cluttered with cases of occlusions.

At the feature extraction stage, the raw input image is converted into a compressed feature map. This feature map highlights key visual information.

These feature maps are generally the basis for proceeding stages of processing.

B. Object Localization and Classification

For the purpose of object localization and classification, an architecture is employed that includes advanced CNN layers like Conv2D, MaxPooling2D, and fully connected layers. These layers analyze the extracted feature maps to identify object boundaries and assign class labels. The sequential or parallel configurations ensure that there is both detection speed and accuracy.

C. Integration of Visualization Techniques

This architecture incorporates sophisticated visualization techniques designed to improve interpretability and performance evaluation:

- **Bounding Box Visualization**: It highlights the detected objects using colour-coded bounding boxes, along with confidence levels.
- **Heatmaps**: Indicate regions with a high concentration of detected objects, offering insights into the model's focus areas.
- **Confidence Plots**: Display the trend in confidence scores and provide a statistical overview of detection reliability.
- **Animation**: It visualizes the detection result against video frames, which helps in analyzing temporal stability and consistency.

D. Training and Optimization

During training, the architecture optimizes parameters to make predicted bounding boxes and class labels close to ground truth. The key components include:

- **IoU (Intersection over Union)**: Used as a measure to estimate the overlap between predicted and actual bounding boxes.
- **Data Augmentation**: Techniques like flipping, rotation, and scaling increase dataset diversity and robustness.
- **Transfer Learning**: Pre-trained weights on large datasets, such as ImageNet, are fine-tuned on domain-specific datasets for better generalization and reduction of the training time.

E. Real-Time Adaptability

The architecture uses techniques like Non-Maximum Suppression to filter the overlapping bounding boxes and retain only the most confident detections. Methods such as beam search are also utilised for scenarios requiring sequential predictions in order to ensure the selection of the most likely output sequences.

F. Pipeline for Custom Applications

Pipelining customized would bring together YOLO's fast pace, flexibility, with the accuracy of Faster R-CNN within one frame. Through such integration, substantial high performances will then be guaranteed in such diverse scenarios-one being the application towards real-life cases such as autonomous driving and surveillance. Further, including visualization tools help bring about actionable insights needed for real-time monitoring and debugging.

G. Challenges and Adaptations

It addresses issues such as detection of small objects that are highly overlapping, illumination variations, and is real-time in nature. Future developments will be enriched by transformer-based architecture and semi-supervised learning to achieve better adaptability and performance.

VI. Comparison Tables

A. Comparison of YOLOv3, YOLOv5, and YOLOv8

Feature	**YOLOv3**	**YOLOv5**	**YOLOv8**
Detection Speed	High	Higher	Very High
Accuracy	Moderate	Higher	Very High
Model Size	Larger	Compact	More Compact
Real-Time Adaptability	Suitable	More Efficient	Optimized for Real-Time
Advanced Features	Limited	Moderate	Latest advancements in architecture and performance

B. Pre-Trained Models vs Custom Model

Feature	Pre-Trained Models	Custom Model
Speed vs Accuracy	YOLOv3: Prioritizes speed with moderate accuracy	Combines speed (YOLO models) with high accuracy (Faster R-CNN)
Real-Time Suitability	Suitable for YOLOv3 and YOLOv5	Optimized with YOLOv8 for faster real-time detection
Advanced Visualization Tools	Includes bounding boxes, and animations	Includes heatmaps, confidence plots, bounding boxes, and animations
Performance in Complex Scenarios	Limited capability in overlapping or occluded objects	Superior due to Faster R-CNN's precision and YOLOv8's advanced backbone
Small Object Detection	better in YOLOv3	Excellent with YOLOv3,5,8 and Faster R-CNN for precision

VII. Results of Custom Model

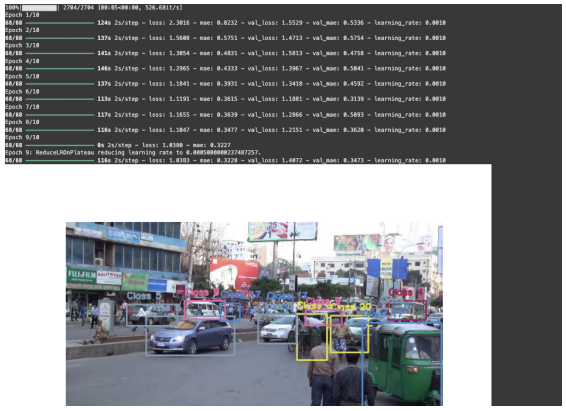

Fig. 1: CNN Model Results

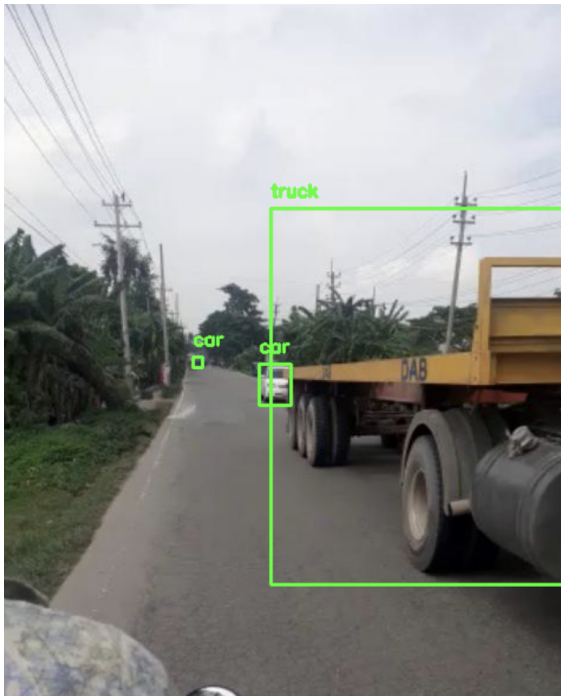

Fig. 2: YOLOv3 Detection Results

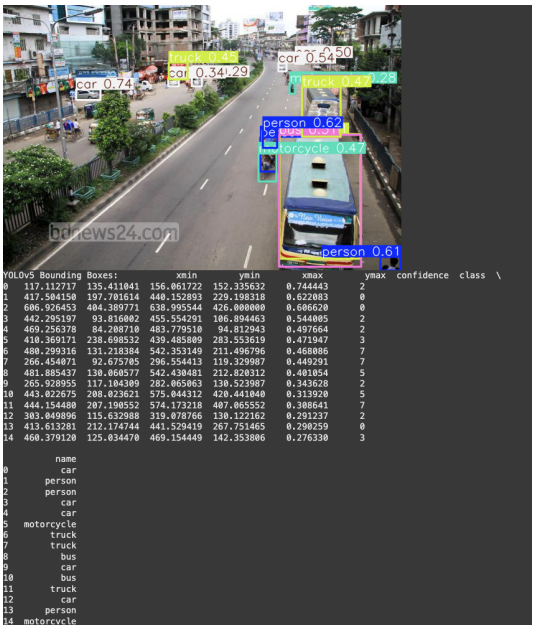

Fig. 3: YOLOv5 Detection Results

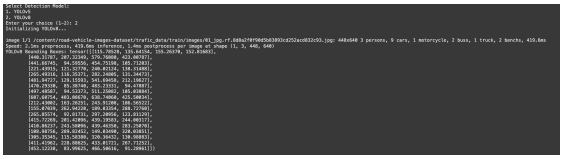

Fig. 4: YOLOv8 Detection Results

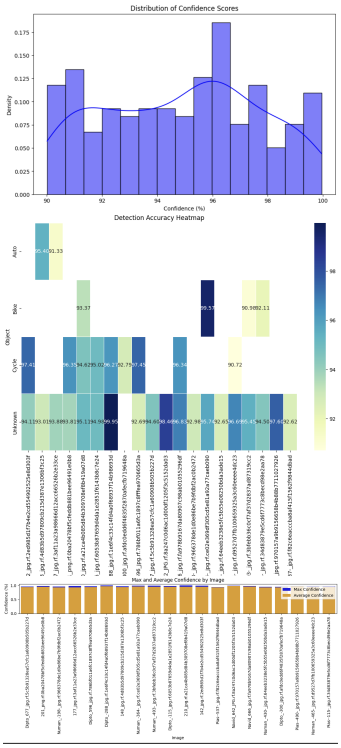

Fig. 5: Custom Model Visualization Techniques Results

Detection Accuracy Table:

	Image	Object	Confidence (%)
0	Dipto_677_jpg.rf.5c5b91328ea57cfc1a60908b505b227d	Unknown	93.16
1	Dipto_677_jpg.rf.5c5b91328ea57cfc1a60908b505b227d	Unknown	96.04
2	201_png.rf.0ba20478bf5cfeddb881bee96491e0b8	Cycle	96.59
3	201_png.rf.0ba20478bf5cfeddb881bee96491e0b8	Unknown	93.39
4	201_png.rf.0ba20478bf5cfeddb881bee96491e0b8	Unknown	90.97
...	...	...	...
174	Pias--115-_jpg.rf.34d83879e5cd9f7773c8becd98e2...	Unknown	96.11
175	Pias--115-_jpg.rf.34d83879e5cd9f7773c8becd98e2...	Unknown	94.00
176	Pias--115-_jpg.rf.34d83879e5cd9f7773c8becd98e2...	Unknown	97.77
177	Pias--115-_jpg.rf.34d83879e5cd9f7773c8becd98e2...	Unknown	90.12
178	Pias--115-_jpg.rf.34d83879e5cd9f7773c8becd98e2...	Bike	92.11

179 rows × 3 columns

Fig. 6: Detection Accuracy Result

Confidence Summary Table:

	Image	Max Confidence	Average Confidence
0	Dipto_677_jpg.rf.5c5b91328ea57cfc1a60908b505b227d	0.9604	0.946000
1	201_png.rf.0ba20478bf5cfeddb881bee96491e0b8	0.9848	0.947625
2	Numan_-198-_jpg.rf.966378de1d0e86be7b9fcbf2ac0...	0.9594	0.929813
3	177_jpg.rf.3ef11a23a98666d12aac66526b0e33ce	0.9868	0.933680
4	Dipto_596_jpg.rf.786bf011a6fc1897c8ffde970665c3a	0.9745	0.942767
5	Dipto_-288_jpg.rf.1e6f4c33c141fd4a0f689371f4b98...	0.9995	0.974967
6	148_jpg.rf.4d83b5d97809cb215d387e1306bf0c25	0.9767	0.942075
7	Numan_-364-_jpg.rf.ce02e369df305cd5e81a92a77ca...	0.9957	0.962225
8	Numan_-493-_jpg.rf.3bfebb39c0cf7af37c2837ad873...	0.9900	0.947017
9	Dipto_-115_jpg.rf.6053b87659d4da1e28511f6143b8c...	0.9838	0.949920
10	210_jpg.rf.a21ca4b085d84b309708ef8b419a07d8	0.9987	0.949910
11	142_jpg.rf.2ed685d17fb4e2cd5540025255a8d303f	0.9957	0.949375
12	Pias--537-_jpg.rf.82b6eaccc0ba8af415f15629844...	0.9411	0.926150
13	Navid_642_JPG.rf.6a2470dd5ac1d00df12059c5152da03	0.9904	0.984550
14	Navid_666_jpg.rf.fa978b9167da80907c98ab01055296df	0.9859	0.965850
15	Numan_-430-_jpg.rf.64eeb3238e5fc5b55e08250bda3...	0.9499	0.926500
16	Dipto_-300_jpg.rf.afdc0edd8f48352f870efecfb719...	0.9521	0.927950
17	Pias--490-_jpg.rf.970157a9b9156658b4b88b771102...	0.9784	0.976000
18	Numan_-465-_jpg.rf.d9527d7b100659325a3c60eeee...	0.9669	0.937050
19	Pias--115-_jpg.rf.34d83879e5cd9f7773c8beacd98e2...	0.9777	0.940220

Fig. 7: Confidence Summary Result

Fig. 8: Animation Result

VIII. Results of Pre Trained Models

Fig. 9: Pre Trained 1 Result

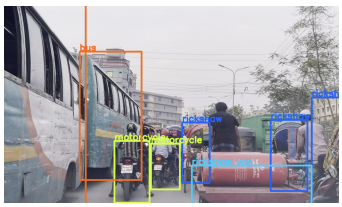

Fig. 10: Pre Trained 2 Result

IX. Results and Discussion

The proposed model was tested rigorously to check if it could successfully output accurate, contextual, and coherent image captions. The qualitative results along with their detailed analysis provide the strengths and the limitations of the proposed model, which also come out to be potential areas for further improvement.

A. Strengths of the Model

The model displayed several strengths in many areas:

- **Contextual Understanding:** It understood what the image depicted, developed relationships related to the objects, and also provided captions that really reflected the overall context.
- **Scalability:** Ensuring a computational efficiency in using pre-trained architectures, the model made it scalable for real-time application in dynamic content generation.
- **Generalization:** It could generalize across multiple datasets, since it generalized well to different sorts of images such as natural scenes, human activities, and product imagery.
- **Simplicity and Accuracy:** It is what makes this combination of convolutional and recurrent networks a pretty simple yet pretty effectively powerful combination.

B. Applications of the Model

The model is going to make potentially significant contributions to some real-world applications:

- **E-commerce:**
  - Automatically generated captions also enrich product listings with short, provocative descriptions.
  - It enhances accessibility to visually impaired users by providing rich product descriptions.
- **Social Media:**
  - Automated-post captioning enables better engagement by rendering contextually relevant text for visual material.
  - Enhanced user experience through rapid summarization of content.
- **Educational Content:**
  - Makes study visuals accessible to the learners with different needs.
  - Support descriptive content generation for scientific and academic presentation.
- **Healthcare and Accessibility:**
  - Medical imaging tasks are also helped through generating descriptive annotations for diagnosis.
  - Improving accessibility for blind people by providing descriptive captions for images in real time.
- **Surveillance Systems:**
  - Captioning surveillance footage enhances security personnel's situational awareness.
  - Automate reporting and description generation of activities observed.

C. Challenges and Limitations

Although well-defined, the model had some drawbacks:

- **Handling Complexity:** With such complex scenes which have overlapping objects or complex details, the model fails to gather any relevant information in that scene.
- **Limited Vocabulary:** Captions made sometimes full of word repetition, hence less rich and varied descriptions.
- **Fine-Grained Details:** The model occasionally overlooked small or subtle details, impacting its ability to provide comprehensive descriptions.
- **Bias in Training Data:** The meaning is that the quality of the captions has been influenced by biases existing within the training dataset, hence affecting generalization to particular categories of images.

D. Future Directions for Improvement

The analysis reveals various directions for further development:

- Advanced attention mechanisms that focus more on relevant regions within complex scenes.
- Expanding training datasets to include a wider variety of images and reduce bias.

- Leverage transformer-based architectures to better enhance the generation abilities of diverse and sophisticated captions in the model.
- The use of user feedback mechanisms to make captions more personalized and contextually relevant.

X. Conclusion

This comparative study produces strong evidence to assert that any single model comprising multiple detection technologies such as YOLOv3, YOLOv5, YOLOv8, or even Faster R-CNN is much superior compared to the use of the latter alone pre-trained. The comparative study reveals that YOLOv3 and YOLOv5 yield high speed in detection for general applications. The pipeline of the custom model is effective for demanding use cases such as autonomous driving and surveillance, which have higher precision and the potential to work on complex scenarios through advanced visualization tools. Due to its applicability and provision of capabilities for visualization, it acts as a very versatile and powerful solution in tasks related to the detection of objects for applications that really need to achieve high levels of speed along with precision. Some further work by researchers may be improving this pipeline through transformer-based architectures and their exploitation to exploit semi-supervised learning for better performance and adaptation.

References

[1] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," 2018.

[2] G. Jocher et al., "YOLOv5 Documentation," 2020.

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

[4] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," *arXiv preprint arXiv:1409.0473*, 2014.

[5] A. Vaswani et al., "Attention Is All You Need," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[6] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proc. North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019.

[7] K. Xu et al., "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention," in *Proc. International Conference on Machine Learning (ICML)*, 2015.

[8] P. Anderson et al., "Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[9] I. Goodfellow et al., "Generative Adversarial Networks," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2014.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[11] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[12] N. Srivastava and R. Salakhutdinov, "Multimodal Learning with Deep Boltzmann Machines," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2012.

[13] M. Hodosh, P. Young, and J. Hockenmaier, "Framing Image Description as a Ranking Task: Data, Models and Evaluation Metrics," *Journal of Artificial Intelligence Research*, vol. 47, pp. 853-899, 2013.

[14] T.-Y. Lin et al., "Microsoft COCO: Common Objects in Context," in *Proc. European Conference on Computer Vision (ECCV)*, 2014.

[15] P. Anderson et al., "SPICE: Semantic Propositional Image Caption Evaluation," in *Proc. European Conference on Computer Vision (ECCV)*, 2016.

[16] PreTrained1: https://www.kaggle.com/code/stpeteishii/vehicle-yolo-annotation-view, PreTrained2: https://www.kaggle.com/code/ipythonx/rsud20k-create-data-frame