Enhancing Object Detection in Autonomous Vehicles: Advanced Semi-Supervised Learning Techniques with YOLOv5 and Model Interpretability

by Shreeya Kumbhoje, MSc

Submitted to The University of Nottingham
in September 2024
in partial fulfilment of the conditions for the award of the degree of
Master of Data Science

I declare that this dissertation is all my own work, except as indicated in the text

Contents

Table of diagrams:

Abstract

This study presents a comprehensive exploration of advanced methodologies for enhancing object detection capabilities in autonomous vehicles. The study addresses the critical challenges associated with real-time object recognition, emphasizing the importance of accurate detection for ensuring safety and operational efficiency in self-driving systems. By leveraging semi-supervised learning (SSL) techniques, the research integrates both labelled and unlabelled data to improve model performance while minimizing the reliance on extensive manual annotations.

The proposed approach builds upon the YOLOv5 model, incorporating innovative strategies such as pseudo-labelling, consistency regularization, and contrastive learning. The research introduces a hybrid SSL framework that effectively utilizes the KITTI dataset, a benchmark in autonomous driving research, to train and validate the model. The methodology includes the development of three distinct models, each designed to enhance detection accuracy and robustness in dynamic environments.

Key contributions of this study include the integration of spatial attention mechanisms to improve model interpretability, allowing for a clearer understanding of the decision-making process in object detection. The empirical results demonstrate significant improvements in mean Average Precision (mAP) and F1 scores, particularly for vehicle classes, while highlighting the challenges faced in detecting less frequent categories.

Furthermore, the research identifies and addresses gaps in existing literature regarding the handling of noisy pseudo-labels and computational inefficiencies. By proposing a cohesive framework that combines various techniques, this study aims to advance the field of computer vision in autonomous driving, providing a scalable and efficient solution to the challenges of large-scale data labelling. The findings underscore the potential of hybrid methodologies in enhancing the reliability and accuracy of object detection systems, paving the way for future developments in autonomous vehicle technology.

Acknowledgements

I would like to extend my heartfelt gratitude to my supervisors for their invaluable guidance and unwavering support throughout my academic journey. Their mentorship has been instrumental in shaping my work. Additionally, I am deeply thankful to my family and friends for their constant encouragement and understanding, which has been a source of strength and motivation during my time at university. Your support has made all the difference, and I am truly appreciative.

Chapter 1: Introduction

“Self-driving [1] cars, also known as driverless or autonomous vehicles [2], represent a groundbreaking advancement in automotive technology and artificial intelligence. These vehicles are designed to navigate and operate without human intervention by leveraging a complex array of sensors, machine learning algorithms”, and decision-making systems. A critical component of the AI embedded in self-driving vehicles is the visual recognition system (VRS), which encompasses various tasks such as image classification, object detection, segmentation, and localization to enable the vehicle's basic ocular performance. This visual processing capability is essential for the vehicle to perceive its surroundings and make informed decisions in real time [3].

Among these tasks, object detection the process of scanning a scene and identifying one or multiple classes of interest is regarded as the heart of scene understanding for autonomous vehicles [4]. Object detection is a crucial technology that enables the identification and localization of objects within images or video streams. This capability plays a vital role across various fields such as autonomous driving, surveillance, robotics, and smart cities. In the context of autonomous driving, object detection is particularly critical because it is directly responsible for recognizing and distinguishing between different entities like vehicles, pedestrians, cyclists, and various obstacles. This recognition directly impacts navigation safety and decision-making processes in real time, ensuring that the vehicle can respond appropriately to dynamic and potentially hazardous environments. However, several challenges, such as variable lighting conditions, partial occlusions, and the inherent complexity of real-world environments, make accurate detection a challenging task [3]. These challenges highlight the need for more robust and adaptable models to ensure reliable performance under diverse operational conditions.

Semi-supervised learning (SSL) has recently emerged as a promising approach in machine learning, particularly for complex tasks like object detection, where the training process involves both labelled and unlabelled data. The significance of SSL in this domain lies in its ability to enhance model performance while significantly reducing the reliance on extensive manual annotation, which is both time-consuming and expensive. By effectively utilizing the vast amount of available unlabelled data, SSL methods often outperform traditional supervised learning approaches in various tasks such as image classification and object detection, thereby improving model generalization, adaptability, and robustness [5][3][4]. For instance, in the context of object detection, SSL techniques enable models to learn from the complexities of real-world environments without requiring exhaustive manual labelling, thereby enhancing the model's adaptability and overall detection accuracy in various scenarios [6][7][8].

Despite its advantages, SSL faces significant challenges, particularly when applied to object detection tasks. Techniques such as pseudo-labelling and consistency regularization, which are commonly used in SSL, are prone to issues such as error accumulation, confidence bias, and computational complexity. These limitations are even more pronounced in real-time applications, where factors like error propagation, sensitivity to data augmentations, and the need to scale across diverse and large datasets can severely impact model performance and reliability [9][6][10]. While pseudo-labelling can enhance model performance by treating predictions on unlabelled data as true labels, it is highly susceptible to confirmation bias, where incorrect predictions reinforce themselves, leading to deteriorated model performance. This technique heavily depends on the quality of the generated pseudo-labels, and poor quality can lead to error accumulation [9][6]. Similarly, consistency regularization, which aims to improve model robustness by ensuring stable predictions across various augmented versions of the same input, requires careful management of computational resources and augmentation strategies to avoid potential performance degradation [8][6]. The balance between computational efficiency and model accuracy remains a critical challenge.

To address these challenges, this research proposed a novel and integrative approach to multi-object detection that combined multiple techniques cohesively within the domain of autonomous vehicles. The methodology was built upon the YOLOv5 model, which served as a backbone for initial training on labelled data. For the purpose of semi-supervised implementation, three distinct models were developed. The first model utilized basic pseudo-labelling, where the model's predictions on unlabelled data were used as pseudo labels for further training. The second model employed a hybrid Mean Teacher consistency regularization framework, which combined pseudo-labelling with spatial explainability techniques, such as attention mechanisms, to enhance model interpretability and robustness. Finally, a third model was introduced, incorporating contrastive regularization with a mean teacher approach. This model aimed to improve feature representation and robustness by ensuring that the model distinguished between similar and dissimilar instances, thereby refining the learning process and reducing the risk of error propagation.

The KITTI dataset, a well-established benchmark in the field of autonomous driving research, will be utilized to train a YOLOv5 model on labelled data. The objective is to enhance the accuracy, robustness, and interpretability of object detection models in safety-critical environments, thereby pushing the boundaries of current SSL methodologies.

The contributions of this research lie in advancing computer vision methodologies, particularly within the scope of autonomous driving, by addressing the key issues in SSL, improving object detection accuracy, and enhancing model interpretability through a well-rounded, data-efficient approach.

Main Contributions:

- Hybrid SSL Approach for Multi-Object Detection: This research proposed a hybrid approach that combined Mean Teacher consistency regularization with pseudo-labelling for multi-object detection in autonomous driving. The proposed method aimed to leverage both labelled and unlabelled data more efficiently than traditional semi-supervised learning techniques. This approach addressed key challenges such as confirmation bias, error accumulation, and computational efficiency, which are critical for deploying robust models in real-world, safety-critical environments.

- Contrastive Regularization with Mean Teacher Framework: The study introduced a novel model that integrated contrastive regularization within the mean teacher framework. This model was designed to improve feature representation by distinguishing between similar and dissimilar instances, enhancing model robustness and generalization capabilities.

- Integration of Explainability Techniques**:** This research incorporated advanced model explainability techniques, such as spatial attention mechanisms, into the object detection framework. The integration of these techniques allowed for improved interpretability of model decisions, which is crucial in safety-critical environments like autonomous driving, where understanding the reasoning behind a model’s decision is vital for validation and regulatory compliance.

- Empirical Validation Using the KITTI Dataset: The proposed methods were empirically validated using the KITTI dataset, a widely recognized benchmark in autonomous research. The study utilized YOLOv5 for training, demonstrating the effectiveness of the proposed hybrid approach. The results suggested a robust and generalized solution that could adapt to different driving scenarios, thereby providing potential improvements in real-world applications.

- Addressing Gaps in Current Research**:** This research identified and addressed specific gaps in the current literature, such as the difficulty in handling noisy pseudo-labels, computational inefficiencies, and limited interpretability in complex object detection frameworks. By cohesively combining multiple techniques, including contrastive regularization, the proposed approach aimed to overcome these challenges and provide a more holistic solution, contributing to the broader field of autonomous vehicle research and development.

Chapter 2: Related Works:

In the rapidly evolving field of computer vision, semi-supervised learning (SSL) has emerged as a pivotal technique to address the challenge of limited labelled data. SSL methods, such as pseudo-labelling, leverage large amounts of unlabelled data to enhance model performance by generating additional training labels based on initial predictions [5]. Despite its promise, pseudo-labelling faces significant hurdles, including the risk of confirmation bias and the potential for incorporating noisy labels, which can degrade the quality of model training [9].

Model explainability has become increasingly critical in understanding and trusting complex deep learning models, particularly in high-stakes applications such as autonomous driving. Traditional explainability techniques like Local Interpretable Model-agnostic Explanations (LIME) [10] and SHapley Additive exPlanations (SHAP) [[11] offer insights into model predictions but often struggle with the complexity of multi-output scenarios inherent in object detection frameworks like YOLOv5 [10]. These methods can be computationally expensive and provide limited global interpretability, which is essential for effective debugging and ensuring fairness.

YOLOv5, an object detection model, combines efficiency and accuracy but also faces challenges in adapting to diverse domains and integrating advanced SSL strategies. Addressing these issues requires innovative approaches to both enhance model performance and provide clearer, more actionable insights into the model's decision-making process.

2.1 Semi-Supervised Learning:

Semi- Supervised learning (SSL) [[12] is a pivotal machine learning paradigm that integrates a small amount of labelled data with a larger pool of unlabelled data during the training process. The approach is particularly advantageous in scenarios where acquiring labelled data is costly and time-consuming, while unlabelled data is often abundant and readily available [13][14]. The primary objective of SSL is to leverage the information contained in unlabelled examples to enhance the model's performance beyond what could be achieved using only the labelled data [14]

In autonomous driving, the ability to accurately detect and classify objects is essential for making real-time driving decisions. A notable advancement in the exploration of various SSL approaches, including consistency regularization, generative models, and graph-based methods. These techniques aim to improve the robustness and accuracy of models in real-world driving scenarios.[13] Such as Consistency regularisation encourages models to produce similar outputs for perturbed versions of the same input, which is particularly useful in dynamic environments encountered by autonomous vehicles [13].

Additionally, the use of graph-based methods has gained traction, as they allow for the effective propagation of labelled information through the relationships between data points. This is particularly relevant in self-driving, where the spatial relationships between detected objects can inform the classification process [15]. However, the KITTI dataset consists of a large number or images and annotated objects. Which could cause graph-based methods to struggle with scalability as the number of nodes and edges increases, leading to computational inefficiencies. Hindering the performance of models like YOLOv5, which are designed for real-time object detection [16].

Research has also highlighted the importance of addressing potential performance degradation when incorporating unlabelled data. Studies have shown that while SSL can improve model performance, it can also lead to decreased accuracy if the assumptions underlying the SSL methods do not align with the data distribution, this has promoted the development of more robust SSL techniques that minimize the risk of performance loss, ensuring that the integration of unlabelled data contributes positively to model training [15].

2.2 YOLOv5:

YOLOv5, or “You Only Look Once version 5,” is a state-of-the-art object detection model that has gained significant popularity due to its efficiency and accuracy in real-time applications. It is part of the YOLO family of models, which is designed to perform object detection in a single pass, making them particularly suitable for scenarios requiring fast inference.

2.2.1 Architecture of YOLO:

- “Backbone: The backbone is responsible for feature extraction from the input images. YOLOv5 offers flexibility in backbone choices, including CSPDarknet, EfficientNet, and ResNet variants. This allows users to tailor the model based on their computational resources and performance objectives. The backbone extracts hierarchical features essential for accurate object detection.
- Neck: The neck network aggregates and refines the features extracted by the backbone. It enhances the discriminative power of feature representations, leading to improved detection performance. The neck plays a crucial role in integrating contextual information and reducing spatial redundancy, facilitating more accurate localization and classification of objects.
- Head: The head of YOLOv5 is responsible for predicting bounding boxes, class probabilities, and confidence scores for each object in the input image. Unlike traditional two-stage detectors, YOLO employs a single stage detection head, simplifying the inference process and enabling real-time performance without sacrificing accuracy.

Detailed examination of YOLO architecture and operational principles, discussing the backbone, neck and head networks in [17] and [18].”

2.2.2 Adapting pre-trained YOLOv5 Models for New Data:

Fine-tuning involves initializing a pre-trained model with its weights and then training it on a new dataset. This process typically requires fewer epochs and less data compared to training a model from scratch, as the model already has learned features that can be adapted to the new task. However, hyperparameters such as learning rate and batch size may need adjustment to optimize performance on the new dataset [18]. To further enhance the robustness of the model, various data augmentation techniques can be applied during training. Techniques such as random cropping, rotation, and scaling help the model generalize better to new data distributions by simulating variations in the input data [19].

Additionally, when adapting to a new dataset, it may be necessary to regroup or drop certain classes to achieve a more balanced class distribution, which helps mitigate false negatives and improves overall detection performance [20] learning techniques also play a crucial role in effectively adapting the model to new tasks. These techniques involve leveraging the features learned from the original dataset while focusing on the specific characteristics of the new dataset, thereby enabling the model to perform well in different contexts [[21]. After adapting the model, it is essential to evaluate its performance using metrics such as mean Average Precision (mAP) and recall. Based on the evaluation results, further adjustments can be made to the model or training process to enhance its performance on the new dataset [18].

2.2.3 Advances in the YOLOv5 for object detection:

Several studies have explored the application of YOLOv5 in various contexts, particularly focusing on its capabilities in multi-object detection and semi-supervised learning, including its performance on the KITTI dataset for 2D image data. In the realm of semi-supervised object detection, the study titled “Efficient Teacher: Semi-Supervised Object Detection for YOLOv5 [22] “. discusses a novel framework that enhances the performance of one-stage anchor-based detectors like YOLOv5 through a student-teacher mutual learning approach. This method aims to improve the quality of pseudo-labels generated from model predictions on unlabelled data, which is crucial for semi-supervised learning. The authors highlight the challenges faced in detecting objects of varying sizes and the limitations of traditional pseudo-labelling methods in the context of object detection. In [[21] another relevant study on cross-domain adaptation develops a framework that utilizes YOLOv5 for cross-domain object detection tasks. The focus is on adapting the model to handle domain shifts, such as variations in weather conditions and camera setups, which are common in datasets like KITTI. The study demonstrates that their approach significantly improves detection performance across different driving scenarios, showcasing YOLOv5's versatility in multi-object detection.

Further research on mixed pseudo labels, titled "[23] Mixed Pseudo Labels for Semi-Supervised Object Detection," emphasizes the importance of generating high-quality pseudo-labels for training object detectors in a semi-supervised manner. The authors propose a method that combines different scales of pseudo-labels to enhance the detection capabilities of YOLOv5. This approach is particularly relevant for datasets like KITTI, where the diversity of object sizes and categories can impact detection accuracy. A comprehensive review, "A Comprehensive Review of YOLOv5: Advances in Real-Time Object Detection [17]." provides insights into the performance metrics of YOLOv5 across various datasets, including KITTI. The study compares YOLOv5 with other models, highlighting its strengths in speed and accuracy, which are critical for real-time applications in autonomous driving and multi-object detection scenarios. Finally, the paper" [[18] Deep Learning Approach: YOLOv5-based Custom Object Detection" discusses the implementation of YOLOv5 for custom object detection tasks, emphasizing its adaptability for different datasets. The study showcases how YOLOv5 can be fine-tuned for improved performance in detecting multiple objects in complex environments

2.3 Pseudo- Labelling:

Pseudo-labelling [24] is a prominent technique in semi-supervised learning (SSL) [[15] that involves generating labels for unlabelled data based on the predictions of a trained model.

2.3.1: Methodology:

- Initial Training: The process begins with training a model on a limited set of labelled data. The model learns to make predictions based on the features of the input data [25].
- Generating Pseudo-Labels: Once the model is trained, it is used to predict labels for a larger set of unlabelled data. The model generates bounding boxes and class labels for the objects detected in these unlabelled images.
- Confidence Thresholding: Then the predicted labels are filtered based on a confidence threshold. Only those predictions that exceed a certain confidence level are retained as pseudo-labels. This step is crucial as it helps to ensure that the pseudo-labels [24] of high quality.
- Augmented Training set creation: The high confidence pseudo labels are combined with the original dataset to create an augmented training set. This new dataset is then used to retrain the model.
- Iterative refinement: The process can be repeated iteratively. After retraining, the model can again predict labels for the unlabelled data, and the cycle continues, allowing the model to improve progressively.

2.3.2 Advantages:

By incorporating pseudo labels into the training process, models can achieve higher accuracy. For instance, studies have shown that model using pseudo labels can outperform those trained solely on labelled data, as they can learn from a broader range of examples [24].

Pseudo-labelling can help models adapt to new environments or conditions by providing additional training data that reflects the target domain this is particularly useful in cross-domain object detection scenarios, where the model needs to generalize across different settings [21]. It can focus on hard to detect instances by refining labels based on model confidence and tracking information. Which can lead to better performance in challenging detection scenarios, such as occlusions or small object sizes [26].

Pseudo-labelling methods can be easily integrated into various object detection frameworks and adapted to different architecture, making them versatile for different applications [25].

2.3.3 Challenges:

Pseudo-labelling methods face several challenges that can impact their effectiveness in training models for tasks like object detection. One major issue is the introduction of false positives (incorrectly labelled objects) and false negatives (missed objects), which can accumulate, especially when the model is trained over multiple frames or iterations, leading to significant performance drops. Addressing these errors requires careful filtering and validation of pseudo-labels [25]. Another challenge is the dependence on model confidence scores to determine which labels to retain, which may result in the exclusion of potentially useful low-confidence predictions that could still be correct, particularly in challenging detection scenarios [26]. Additionally, some pseudo-labelling approaches, especially those involving consistency regularization or multiple model evaluations, can be computationally intensive, which can hinder their application in resource-constrained environments such as mobile or edge devices [25]. Domain shift issues also pose a significant challenge; pseudo-labelling often assumes that training and test data come from the same distribution. However, in real-world applications like autonomous driving, the target domain may differ significantly from the source domain, leading to a drop in detection accuracy, necessitating adaptations to handle domain shifts [27]. Furthermore, there is a risk of overfitting to noisy labels when pseudo-labels are used for training, particularly if the model is trained for too many iterations without sufficient validation, which can result in poor generalization to unseen data [24]. Lastly, pseudo-labelling methods may lack robustness to variations in data, such as changes in lighting, weather conditions, or object appearances, complicating the training process and affecting the model's performance in real-world scenarios [21].

2.3.3.1 Confirmation bias:

In pseudo-labelling, a model generates labels for unlabelled data based on its predictions. High-confidence predictions are typically retained for training, while low-confidence predictions are discarded. However, if the model's confidence is misplaced, it can lead to confirmation bias, where the model reinforces its own errors by training on incorrect pseudo labels [25] Confidence bias can result in decreased generalization, increased error rates, and imbalanced performance across different object categories. This is particularly problematic in object detection tasks, where accurate localization is critical [26].

Arazo et al. [24] discuss the implications of pseudo-labelling and confirmation bias in deep semi-supervised learning, highlighting how reliance on high-confidence predictions can lead to performance degradation and suggesting methods to mitigate this issue by refining the pseudo-labelling process. Similarly, Chun et al. (2024) [25]. proposes an uncertainty-based approach to enhance pseudo-label reliability by leveraging uncertainty obtained through Gaussian modelling to filter out low-reliability annotations, thereby addressing the challenges posed by confidence bias in pseudo-labelling Sajid et al. [26] introduce a bi-directional recovery method that utilizes tracking information to recover lost pseudo labels, ensuring that both high- and low-confidence predictions are considered, which improves the overall quality of the pseudo labels and reduces the impact of confidence bias. Wang et al. (2021) [27] present a domain adaptation framework that optimizes pseudo labels to correct errors, focusing on hard samples and employing adaptive sampling to enhance the reliability of pseudo labels, indirectly addressing issues related to confidence bias. Li et al. [21] the S-DAYOLO framework, which incorporates category-consistent regularization and adaptation modules to generate domain-invariant representations, thereby mitigating the effects of confidence bias by ensuring that the model learns robust features across different domains.

2.4 Consistency Regularisation:

Consistency regularization promotes stability in model predictions across different augmentations or transformations of the same unlabelled sample. This approach ensures that the model's predictions remain consistent, thereby enhancing robustness against variations in object appearance and position [26]. consistency regularization is highlighted as a method that can be seen as a form of label propagation. It operates under the assumption that training samples that resemble each other are likely to belong to the same class. The paper [28] proposes the Mean Teacher method, which averages model weights to form a target-generating teacher model, thereby improving the speed of learning and classification accuracy, The technique enhances the quality of teacher-generated targets in teacher-student frameworks, which in turn boosts the overall performance of the model.

2.4.1 Mean Teacher:

1. Framework Structure: The Mean Teacher method consists of two models: a student model and a teacher model, both of which share the same architecture. The student model is trained using standard backpropagation, while the teacher model's weights are updated using an exponentially weighted moving average (EMA) of the student model's weights. This allows the teacher model to provide stable predictions that guide the student model's learning process [28].
2. Consistency Regularisation: The core idea behind the Mean Teacher method is to enforce consistency between the predictions of the student and teacher models. During training, both models receive the same input, but with added noise or perturbations. The consistency loss is computed based on the difference between the outputs of the student and teacher models, encouraging them to produce similar predictions for the same input [29].
3. Application in autonomous driving**:** In the context of autonomous driving, the Mean Teacher framework can be utilized to improve the robustness of object detection systems. For instance, the method can be applied to detect various objects in driving scenes, such as pedestrians, vehicles, and traffic signs, even when the training data is partially labelled. The teacher model helps to refine the predictions of the student model, leading to better performance in real-world driving conditions [30].
4. Performance Improvement: Research has shown that the Mean Teacher method can significantly enhance the performance of object detection models in autonomous driving applications. For example, it has been demonstrated that combining the Mean Teacher approach with advanced architectures like Residual Networks can lead to state-of-the-art results on benchmark datasets, improving detection accuracy while using fewer labelled samples [28].
5. Challenges and Future Directions: While the Mean Teacher method has shown promise, challenges remain, particularly in dealing with noisy pseudo-labels and ensuring that the model generalizes well to diverse driving conditions. Future research may focus on integrating additional techniques, such as contrastive learning, to further enhance the feature adaptation process in object detection for autonomous driving [30].

2.4.2 Combining consistency Regularisation with Pseudo-Labelling:

The Contrastive Mean Teacher (CMT) [30] framework, as discussed in recent studies, has demonstrated notable success in domain adaptation tasks. This framework integrates pseudo-labelling with consistency regularization and object-level contrastive learning to enhance feature adaptation across different domains. Specifically, when applied to the KITTI to Cityscapes domain adaptation, the CMT framework, in combination with the Probabilistic Teacher (PT) method, achieved a mean Average Precision (mAP) of 64.3% for the "Car" category, marking a significant improvement over the baseline PT method which had a mAP of 60.2% [30]. The success of the CMT framework suggests that similar techniques, with appropriate modifications, could be effectively adapted for semi-supervised learning. Such adaptations have the potential to enhance feature learning, improve robustness to domain shifts, and achieve better performance with limited labelled data. Applying these principles to semi-supervised learning scenarios could lead to more accurate and reliable models by leveraging the strengths of both labelled and unlabelled data.

[30] The effectiveness of the Mean Teacher method in object detection can be evaluated using metrics such as mean Average Precision (mAP) on benchmark datasets like PASCAL VOC or COCO, where improvements in detection accuracy can be observed when using the Mean Teacher framework compared to traditional supervised learning methods [31]. The integration of pseudo-labelling and consistency regularization has been applied to various object detection frameworks, including YOLO and Faster R-CNN. These frameworks benefit from the enhanced training data generated through pseudo-labelling, which allows them to generalize better across different environments [21].

However, despite the successes, challenges remain, such as the potential for confirmation bias when using low-quality pseudo-labels [25].

2.5 Contrastive Regularisation:

Contrastive regularization is another powerful technique that enhances model robustness, particularly in semi-supervised and self-supervised learning scenarios. This approach works by contrasting positive and negative pairs of data points in the feature space. Positive pairs, such as different augmentations of the same image, are encouraged to have similar representations, while negative pairs, such as images from different classes, are pushed apart. A contrastive loss function, like InfoNCE loss [32].

, is employed to minimize the distance between positive pairs and maximize the distance between negative pairs, thereby enabling the model to learn to cluster similar instances together. In semi-supervised learning, contrastive regularization can be combined with pseudo-labeling, where the model generates pseudo-labels for unlabelled data and maintains consistency across augmented views of the same input. This technique has proven to enhance the efficiency and accuracy of semi-supervised methods by effectively leveraging unlabeled data, leading to improved generalization and task performance. Studies such as "Contrastive Regularization for Semi-Supervised Learning" by Doyup Lee et al. and "ConMatch: Semi-Supervised Learning with Confidence-Guided Consistency Regularization" by Jiwon Kim et al. demonstrate the utility of contrastive regularization in clustering features and improving model confidence in predictions [33]. Similarly, Enrico Fini et al. discuss the effectiveness of clustering-based self-supervised methods adapted for semi-supervised learning, while Khanh-Binh Nguyen explores contrastive regularization alongside ensemble methods for debiasing and improving model calibration [34].

2.6 Model Explainability:

Model explainability is crucial in machine learning (ML) and deep learning (DL) due to the complexity and opacity of these systems. It fosters trust and acceptance, especially in critical fields like healthcare and finance, autonomous driving by providing insights into decision-making processes [11]. Explainable AI (XAI) promotes transparency and accountability, which is vital for regulated industries [35]. Ethical considerations are addressed by ensuring fairness and mitigating biases [[36]. Explainability is also essential for regulatory compliance, such as GDPR requirements [35], and improves model performance by allowing developers to identify weaknesses and enhance design [16]. It empowers users to understand and question AI decisions, aiding fields like recommender systems [36]. In sectors like healthcare, autonomous driving it facilitates collaboration between human experts and AI systems by making AI-driven insights more interpretable [[11].

2.6.1 Common model explainability models:

1. LIME:

Local Interpretable Model-agnostic Explanations (LIME) is a widely used technique that explains individual predictions by approximating the model locally with a simpler, interpretable model. In the context of object detection, LIME generates perturbations of the input image and observes how these changes affect the model's predictions. By fitting a local surrogate model to these perturbed samples, LIME provides insights into which parts of the image are most influential in the model's decision-making process. This method is particularly useful for understanding the behaviour of complex models like deep neural networks in autonomous systems [37].

Limitations of LIME for Explaining YOLOv5 Object Detection in Semi-Supervised Learning:

Applying LIME to object detection models like YOLOv5, especially in a semi-supervised learning context involving pseudo-labelling, presents several challenges. These models produce complex multi-output predictions per image, such as bounding boxes and class labels, making it difficult for LIME to approximate each output with a simpler model and provide meaningful interpretations [37]. The approach also requires generating numerous perturbations of the input image to evaluate the model's responses, which becomes computationally expensive due to YOLOv5's high-dimensionality and complexity, rendering it impractical for real-time scenarios or applications demanding swift decisions [35]. Furthermore, LIME's effectiveness depends significantly on the quality and relevance of these perturbations. Small changes in input images can lead to large variations in predicted bounding boxes or class labels, resulting in unstable and inconsistent explanations that may not accurately reflect the model's actual decision-making process [37]. In a semi-supervised setting where unlabelled data is utilized through pseudo-labelling, incorrect or noisy pseudo-labels can further compromise the reliability of explanations provided by LIME, potentially leading to misleading interpretations of the model’s behaviour [16]. Moreover, LIME's focus on local explanations of individual predictions does not provide a holistic view of the model's behaviour, which is crucial for understanding, debugging, and ensuring fairness and transparency when deploying object detection systems trained on both labelled and unlabelled data [36]

2. SHAP:

SHapley Additive exPlanations (SHAP) is based on cooperative game theory and provides a unified measure of feature importance. It assigns each feature a Shapley value, quantifying its contribution to the prediction. In object detection, SHAP can be adapted to evaluate the importance of different regions in an image for the model's predictions. By calculating the Shapley values for each pixel or region, SHAP helps identify which parts of the image are critical for detecting specific objects, thereby enhancing transparency in autonomous systems [11].

Limitations of SHAP for Explaining YOLOv5 Object Detection in Semi-Supervised Learning:

SHAP is primarily suited for simpler models and tabular data and struggles with the complexity of multi-output scenarios like object detection, where multiple outputs (bounding boxes and class labels) need interpretation[16];[35], the computational cost of computing SHAP values for high-dimensional models like YOLOv5 is prohibitive, requiring evaluation of all feature combinations, making it impractical for large datasets such as KITTI [35]. In semi-supervised learning with mean teacher consistency regularization, noisy or incorrect pseudo-labels from a teacher model can lead to misleading SHAP explanations, compromising the interpretability of the results [[36]. Furthermore, SHAP assumes feature independence, which is often not the case in object detection where features (pixels) are interdependent, causing inaccurate SHAP values and failing to capture the complex relationships between features [38]. Finally, SHAP focuses on local interpretability, explaining individual predictions rather than providing a global understanding of the model's behaviour. In object detection, especially when dealing with unlabelled data, global interpretability is essential for ensuring fairness and transparency [36].

Attention mechanisms are integral to many modern deep learning architectures, particularly in natural language processing and computer vision. In the context of object detection, attention mechanisms allow models to focus on specific regions of an image when making predictions. By generating attention maps, these mechanisms highlight the areas of the input that are most relevant to the model's decision. This is particularly beneficial in autonomous scenarios, where understanding the model's focus can improve trust and safety in applications like self-driving cars [37].

Limitations of Attention Mechanism for Explaining YOLOv5 Object Detection in Semi-Supervised Learning:

Attention mechanisms in object detection models like YOLOv5 produce attention maps highlighting the regions the model focuses on when making predictions. However, interpreting these maps can be complex due to overlapping attention regions in multi-object scenes, which creates ambiguity in determining feature influence [37] Attention maps are also sensitive to input variations, making consistent explanations challenging [37]. Furthermore, incorporating attention mechanisms may necessitate architectural changes, impacting real-time performance and global interpretability, especially when using semi-supervised learning with pseudo-labelling see [16];[38]. Lastly, noisy pseudo-labels in semi-supervised learning can undermine attention-based explanations, potentially leading to misleading interpretations [[36].

Advantages:

Attention mechanisms play a crucial role in enhancing model performance and interpretability across various applications. They allow models to focus on specific parts of the input data that are most relevant for making predictions, such as highlighting regions in an image where objects are likely to be located. This selective focus not only aids in better localization and classification but also provides a clear visual representation of which features are influential in the model's decisions, thereby improving interpretability [37]. Additionally, visualizing attention weights offers significant transparency into the model's decision-making process, which is especially valuable in critical applications like autonomous driving and healthcare, where understanding the rationale behind predictions is essential for trust and safety [16].

Attention mechanisms are particularly effective in handling complex scenes with multiple overlapping objects by enabling the model to disentangle these complexities and focus on different objects selectively. This capability helps in ignoring irrelevant background information and improving object localization and classification [37]. Furthermore, in semi-supervised learning scenarios where labelled data is limited, attention mechanisms can optimize the use of abundant unlabelled data by focusing on the most informative features from both labelled and pseudo-labelled data, thus enhancing performance and generalization [38].

Moreover, attention maps are valuable for error analysis, as they reveal the model's focus during incorrect predictions. This insight can help identify potential weaknesses or biases in the model, facilitating targeted improvements in the training process [36] Overall, attention mechanisms contribute significantly to model transparency, interpretability, and performance across various challenging scenarios

Spatial attention Mechanisms Explainability model:

Spatial attention mechanisms in model explainability focus on enhancing interpretability by allowing models to concentrate on specific regions of an input image that significantly influence their predictions [39]. In semi-supervised learning, where labelled data is limited, spatial attention helps the model generalize better from the labelled to the unlabelled data by emphasizing the most informative features, thus facilitating robust feature learning across diverse contexts [40]. Moreover, spatial attention aids consistency regularization by ensuring the model focuses on consistent regions across different augmentations, thereby stabilizing feature learning [1]. It also mitigates the impact of label noise by concentrating on relevant features and ignoring misleading labels, which is crucial in multi-object scenarios where some objects may be mislabelled or occluded [41]. Finally, integrating spatial attention with YOLOv5 enhances both speed and accuracy in real-time detection tasks, making it well-suited for applications like autonomous driving [39].

2.7 KITTI Dataset:

The KITTI dataset has been extensively utilized for developing and benchmarking algorithms for multi-object detection. It provides a rich set of labelled data, including 3D bounding box annotations for various object classes such as cars and pedestrians. Researchers have leveraged this dataset to train and evaluate models that can detect and track multiple objects in complex urban environments. The dataset's benchmarks for 3D object detection and tracking have facilitated the comparison of different algorithms, pushing the state-of-the-art in computer vision methods [42].

In the context of SSL, the KITTI dataset has been used to explore methods that can effectively utilize both labelled and unlabelled data. Researchers have developed techniques that propagate labels from a sparse set of annotated keyframes to all unlabelled frames based on temporal coherence and spatial information. This approach has been shown to improve the performance of models in detecting and segmenting objects in video sequences, as it allows for leveraging the temporal continuity of video data to enhance learning [43].

2.8 Conclusion:

These approaches including contrastive regularization, pseudo labelling, and consistency regularization demonstrate significant high potential in using unlabelled data to improve model performance. Pseudo-labeling and its extensions such as mean teacher method showcase the effectiveness of generating high quality pseudo labels to guide the learning process. Meanwhile, contrastive regularization offers complimentary benefits by refining feature representations through contrastive learning, leading to more discriminative models. The integration of these methods with advanced architectures, like YOLOv5 and attention mechanisms, further enhances interpretability and model performance, providing deeper insights into decision-making processes and addressing key challenges. The exploration of model explainability methods, including LIME, SHAP, and attention mechanisms, reveals both their strengths and limitations in explaining complex object detection models. However, challenges remain, particularly in handling noisy labels, ensuring model generalization, and achieving a balance between interpretability and performance. This research with the focus of refining such techniques conducts a study with different variations and hybrid models that could potentially me employed for deployment in self-critical applications like smart driving.

Chapter 3: Methodology:

3.1 Data Preparation:

The KITTI dataset ideal for object detection within the scope of autonomous driving is used. The dataset features a collection of 2D annotated images categorized into nine classes: "Car," "Truck," "Van," "Pedestrian," "Person sitting," "Cyclist," "Tram," "Miscellaneous," and "Don't Care”. The data is pre-partitioned into a test set containing 7518 images without annotations and a training dataset of 7481 images. To integrate semi-supervised the training set is further split into three categories a labelled data set containing 2000 annotated images, an unlabelled set with 4999 images without labels and a validation set of 482 images and their corresponding labels.

The dataset was refined to focus on specific vehicle classes ("Car," "Truck," "Van," and "Miscellaneous"), excluding others such as "Don't Care," "Pedestrian," and "Cyclist" to simplify the model's learning task. All images are standardized to 416 x 416 pixels using bilinear interpolation, normalised and converted to tensor format for input consistency. Labelled data is formatted for YOLOv5 with class IDs and bounding box coordinates,

$$x\_center = \frac{X_{\min} + X_{\max}}{2}$$

$$y\_center = \frac{Y_{min} + Y_{\max}}{2}$$

$$width = X_{max} - X_{\min}$$

$$height = Y_{max} - Y_{\min}$$

followed by data augmentations like random rotations and colour adjustments to improve generalization. Unlabelled data undergoes simpler transformations, including resizing, tensor conversion, and normalization.

3.2 YOLOv5s:

The YOLOv5 model known for it real time implementation was leveraged for training the labelled dataset, with key hyperparameters such as batch size, number of epochs and image pixel dimensions. For YOLOv5, the input images were sized to a resolution of 416 x 416 pixels, a standard size that balances the trade-off between computational efficiency and detection accuracy and the batch size which determines the number of images processed simultaneously during each training iteration was set to **16**. The training involved 50 epochs where each epoch represented a complete pass through the entire dataset ensuring that the model learns from the data iteratively improving its capabilities with each pass. A configuration file containing paths to training labelled data as well as validation data was used to drive the implementation ensuring the correct loading of data. After which the model underwent fine tuning for an additional 10 epochs for refinement, followed by performance evaluation conducted on validation set.

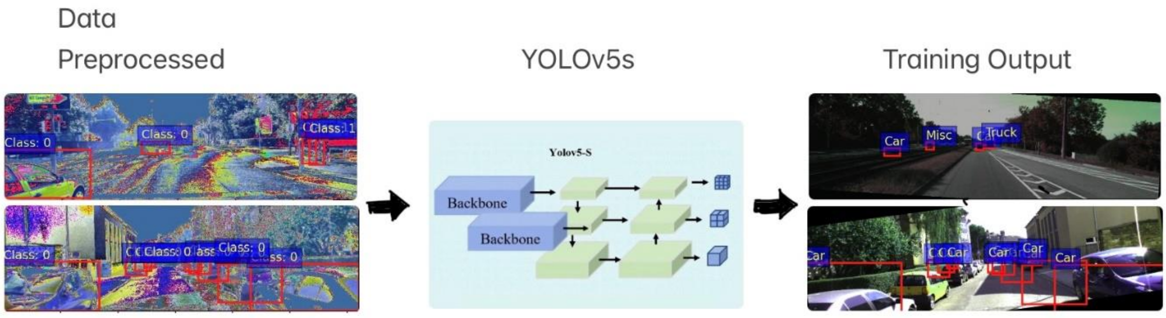

Figure 1. Yolov5s

3.3 Pseudo-Labelling implementation:

For the Implementation of Semi-supervised learning the basic Pseudo-labelling was the first chosen approach initially, the YOLOv5 model was loaded and set to evaluation mode to ensure the proper configuration of dropout and batch normalization layers for inference. The model was then employed to generate predictions on a set of unlabelled data, which had been resized to 416 x 416 pixels and normalized according to the model's input specifications. During inference, the model's output bounding boxes and class probabilities were filtered based on a predefined confidence threshold. Experimentation was conducted using threshold values of 0.3 and 0.5. Subsequently, these filtered predictions were converted to the YOLO format by normalizing the bounding box coordinates and saved in a text file. The resulting pseudo labels were then integrated with an existing labelled dataset to enhance the training data.

To address potential challenges such as error accumulation and confidence bias, multiple strategies were implemented. One approach involved using a confidence threshold to filter predictions based on their confidence scores, ensuring only high-confidence predictions were included in the pseudo-labels. This threshold was crucial in minimizing the risk of incorporating incorrect labels, which could adversely affect the training process. Additionally, a try-except block was incorporated to handle errors during file operations and visualizations, such as file not found errors or issues writing to directories. This approach prevented script crashes and enabled logging for further debugging. Furthermore, the predictions were validated before converting them into YOLO format by excluding those with invalid class names or belonging to ignored categories, thus maintaining the accuracy and relevance of the pseudo labels. The visualization of attention maps was also managed effectively to ensure clarity and correctness.

The effectiveness of pseudo-labelling was evaluated using various key criteria. The primary metric for assessment was the improvement in detection accuracy, determined by comparing model performance indicators, such as Mean Average Precision (mAP), Precision, Recall, and F1 Score, before and after incorporating pseudo-labelled data. This comparison helped to establish whether the pseudo labels contributed to more precise object detection. Another critical criterion involved assessing the model's robustness and generalization capabilities by testing it on a separate dataset that was not part of the training process. The model's performance across diverse scenarios, including variations in lighting and object occlusions, was analysed to evaluate its robustness. Additionally, the consistency and quality of the pseudo-labels were scrutinized to ensure they accurately represented object locations and classifications. Finally, the stability and convergence of the model's training process were monitored by examining metrics such as loss and accuracy over epochs. Stable and consistent training patterns indicated that the pseudo-labelled data positively influenced the training process without introducing significant noise. Collectively, these evaluation criteria provided a comprehensive framework for assessing the impact of pseudo-labelling on enhancing model performance and its ability to generalize effectively.

3.4 Hybrid Consistency Regularisation Implementation:

To enhance the performance of the YOLOv5 object detection model strategy such as consistency regularisation can be applied. Consistency regularisation helps in stabilizing the training process [33]. In the proposed model, a semi-supervised learning approach is utilized for object detection by leveraging the KITTI dataset. The model architecture is further enhanced with a spatial mechanism and a mean teacher framework. This innovative setup is designed to make effective use of both labelled and unlabelled data during the training process. By doing so, the model can learn more robust features and achieve higher accuracy. The combination of these elements 1. Consistency regularization; 2. Semi-supervised learning; 3. Mean teacher framework; and 4. Model explainability technique study how a more thorough training procedure might enhance the model's robustness and performance in object detection tasks.

3.4.1 Custom Function and Modification:

In this section a few custom modifications and changes are discussed:

1. CIoU loss function: The Complete Intersection over Union (CIoU) loss function [44] (ciou_loss) is a regression loss function used in object detection tasks to improve the accuracy of bounding box predictions. It builds upon the traditional Intersection over Union (IoU) metric by incorporating additional geometric factors that enhance the optimization process during training. CIoU was implemented to enhance bounding box regression in object detection tasks by improving upon the standard Intersection over Union (IoU) metric. CIoU integrated additional geometric factors, such as the Euclidean distance between the centre points of predicted and ground truth bounding boxes, as well as their aspect ratios, providing a more precise alignment and optimization for both non-overlapping and partially overlapping boxes. The CIoU loss function was defined mathematically as:

$$CIoU = IoU - \frac{\rho^2(b,\ b_g)}{c^2} + \alpha v$$

Where $\rho^2(b,\ b_g)$ represented the Euclidean distance between the centres of the predicted box b and the ground truth box $b_{g\,,}$ c was the diagonal length of the smallest enclosing box covering the aspect ratio both b and $b_{g,}$ v measured the similarity of aspect ratios, and α was a weighting factor to balance the aspect ratio term. The IoU metric, which was the ratio of the intersection area to the union area of the predicted and ground truth boxes, served as the base of this formula but lacked consideration of centre distance and aspect ratios:

$$IoU = \frac{Intersection\ Area}{Union\ Area}$$

To compute CIoU, the intersection area was derived by finding the maximum and minimum coordinates of the intersecting rectangle, and the union area was calculated by summing the areas of the predicted and target boxes and subtracting the intersection area. Furthermore, the Euclidean distance between the centres of the predicted and target boxes was calculated as:

$$\rho^2(b, b_g) = \sqrt{\left(c_x - c_{x_g}\right)^2 + \left(c_y - c_{y_g}\right)^2}$$

The CIoU formula penalized bounding boxes not only based on overlap but also their centre distance and aspect ratio differences, resulting in more stable and precise convergence during training. The final CIoU loss was then formulated as:

$$CIoU\ Loss = 1 - CIoU$$

This comprehensive formulation ensured better overlap, alignment, and aspect ratio consistency between the predicted and target boxes, ultimately leading to enhanced model performance in object detection tasks.

2. Spatial Attention: A Spatial Attention class was introduced, defining a straightforward convolutional layer designed to compute spatial attention maps. This mechanism was implemented to enhance the model's ability to focus on the most pertinent regions within an image, which potentially improved object detection performance by refining features essential for object localization. The forward method was executed to multiply the input features by the computed attention map, resulting in attention-weighted features that were then fed into the base YOLOv5 model.

3. Mean Teacher:
In conjunction with this, a Mean Teacher framework was adopted, comprising a student model, which was trainable, and a teacher model, which was non-trainable. The teacher model's weights were updated through an exponential moving average of the student model's weights. This updating process was managed by blending the student and teacher model weights based on a parameter alpha. The update mechanism was expressed as:

$$\theta_T \leftarrow \alpha\theta_T + (1 - \alpha)\theta_S$$

Where θ_T represented the weights of the teacher model, θ_S denoted the weights of the student model, and α was the smoothing coefficient, typically close to 1, regulating the influence of the previous teacher weights versus the new student weights.

After which the base YOLOv5 model was integrated with a spatial attention module. This enhanced model first extracted features using the YOLOv5 backbone, applied spatial attention to these features, and then predicted bounding boxes based on the attention-weighted features. This configuration was designed to improve the model's ability to capture critical details within an image, thereby enhancing object detection accuracy.
A number of techniques were used to help with the efficient processing and alignment of the model outputs. The management of outputs from the teacher and student models was the responsibility of these methods. To be more precise, the student model's outputs were rearranged to fit the YOLO format after bounding box predictions were taken from the YOLOv5 outputs at different scales. Likewise, processing was done on the instructor model's outputs to guarantee that they were compatible with the YOLO format. For correct loss computation, the outputs from the teacher and student models were also aligned to preserve uniformity in shape and format.

4. Tensor Size Alignment for Bounding Boxes:
The process for managing tensor sizes and ensuring alignment across bounding boxes was addressed through a method that was designed to handle cases where the number of bounding boxes in a tensor either fell short of or exceeded the required quantity. Padding was applied when the number of bounding boxes was less than the desired count, and truncation was used when the count exceeded the required number. This procedure was crucial for avoiding dimension mismatch errors during loss calculation and training, particularly when integrating outputs from different models or datasets.

Mathematically, this process was represented as follows:

Padding: When the number of bounding boxes N in the tensor was less than the required number MMM, padding was applied. The resulting padded tensor T_{pad} was expressed as:

$$T_{pad} = \begin{cases} T \oplus \ \ zeros\ (M - N)\ if\ N < M \\ T \qquad\qquad\qquad\quad if\ N = M \end{cases}$$

Where ⊕ denoted concatenations with zero to meet the desired count M.

Truncation: when the number of bounding boxes N exceeded the required number M, truncation was applied. The truncated tensor $T_{truncate}$ was expressed as:

$$T_{truncate} = \begin{cases} T[:M] & if\ N > M \\ T & if\ N = M \end{cases}$$

Where T[:M] represented slicing the tensor to retain only the first MMM bounding boxes.
These operations ensured that tensor dimensions remained consistent and compatible for subsequent processing stages, facilitating accurate loss computation and effective model training.

5. Training Loop:
The core training loop was executed within the main function, where both labelled and unlabelled data were utilized. During each iteration, labelled images and their corresponding labels were fed into the student model, while unlabelled images were processed by the teacher model to generate pseudo-labels. The outputs from both models were then processed and aligned. Two loss components were computed: the supervised loss from labelled data and the unsupervised loss from the pseudo-labels generated by the teacher model. The total loss was determined as a weighted sum of these two components, allowing the model to learn effectively from both labelled and unlabelled data. Error handling and debugging were incorporated through various checks and assertions to ensure compatibility of tensor sizes and shapes during operations such as reshaping and loss computation. For instance, errors were raised if the reshaped outputs did not match the expected number of elements, aiding in debugging and maintaining the stability and reliability of the training process.
Following the computation of the total loss, backpropagation was performed, and the optimizer updated the student model's weights. Subsequently, the teacher model was updated to adhere to the Mean Teacher framework’s semi-supervised learning strategy, ensuring smooth evolution of the model's parameters.

3.5 Hybrid Contrastive Regularisation:

Another method of contrastive regularization was tried. Techniques including mean teacher, contrastive learning, and semi-supervised learning were used in the second proposed model. Learning rich feature representations that capture the underlying structure of the data is made easier with the use of contrastive learning. This is especially useful in cases where there is a limited labelled data since it enables the model to efficiently utilize the large amount of unlabelled data. (Chen and colleagues, 2020; He and colleagues, 2020). By training the model to distinguish between similar and dissimilar instances, contrastive learning promotes generalization. This is crucial in semi-supervised learning settings, where the model must perform well on unseen data. The ability to learn from both labelled and unlabelled data through contrastive methods has been shown to improve performance significantly [45].

3.5.1 Custom Function and Modifications:

The functions for this model have been altered and modified as follows:

1. Preprocessing and Resizing Functions for Model Outputs:
In order to suit the necessary dimensions, the student model's outputs first went through standardization procedures. 2-dimensional tensors were unsqueezed to a 4-dimensional format to satisfy the model's input requirements, while 5-dimensional tensors were averaged along one dimension to lessen their complexity. Next, a common reference size was established by identifying the smallest spatial dimensions among a group of feature maps. To properly aggregate the feature maps, it was necessary to resize each one to a uniform dimension using this reference size. Bilinear interpolation was used to resize feature maps in order to get the necessary dimensions. This technique guarantees that features will scale and align smoothly at various scales. By preserving consistency and compatibility across feature maps, this method allowed the model to handle multi-scale object detection.

2. Feature Aggregation Module:

A feature aggregation module was implemented to effectively manage and integrate features extracted from various layers or scales of the model, addressing the challenge of varying output dimensions encountered with YOLOv5. In the context of YOLOv5, the output dimensions of feature maps frequently varied due to the use of images with different resolutions during training. This variability often led to issues with training consistency, as the model required a fixed feature map size for optimal performance, causing frequent interruptions and necessitating a robust solution to handle such inconsistencies. To address this challenge, the feature aggregation module was designed to concatenate features from different layers or scales of the model, ensuring that tensor dimensions were aligned for proper integration. The concatenation process involved merging features from multiple layers, with careful attention to dimension matching to facilitate seamless integration. This approach allowed the model to leverage multi-scale features effectively, enhancing its ability to detect objects with varying sizes and scales.

3. Custom Contrastive Loss Function:

A custom contrastive loss function was introduced to refine feature representations. This process began by calculating a similarity matrix S between normalized feature vectors z_i , which represented the embeddings of different objects or samples. Specifically, the similarity between each pair of feature vectors was measured using the dot product:

$$S_{ij} = z_i + z_j$$

This matrix S provided a measure of how closely related feature vector were to one another.

To derive the contrastive loss, a log-softmax operation was applied to this similarity matrix. The contrastive loss function $\mathcal{L}_{contrastive}$ was computed as:

$$\mathcal{L}_{contrastive} = -\frac{1}{N}\sum_{i=1}^{N} log \frac{\exp{(S_{ij})}}{\sum_{j=1}^{N}\exp{(S_{ij})}}$$

Where N is the number of samples in the batch. This log-softmax operation transformed the similarity scores into a probability distribution, which emphasized the separation between feature vectors of different classes while minimizing the distance between vectors of the same class. By focusing on this loss function, the model aimed to enforce a structure in the feature space where similar objects had closely aligned embeddings, and dissimilar objects were positioned farther apart

4. Pseudo-Labelling Pipeline for Data Augmentation

To enhance the labelled dataset, a pseudo-labelling pipeline was put into place, which used a teacher model's predictions to generate pseudo-labels. Using the teacher model on an unlabelled data set was the first step in this approach. For these unlabelled data, the teacher model produced predictions that were updated in real time via the Mean Teacher framework. A confidence criterion was set to verify that the pseudo-labels were useful in training the model. To maintain the accuracy and potential of the pseudo-labels, this threshold acted as a filter, keeping only those predictions that satisfied a predetermined degree of confidence. Robust error handling procedures were incorporated throughout the pipeline to handle potential problems arising from missing or corrupt photos in the collection default placeholders were used to maintain the continuity and integrity of the dataset.

5.Training Loop:

The main training loop was structured to integrate and coordinate all components involved in model training and updates. Initially, the YOLOv5 student and teacher models were loaded, and the training environment was prepared by configuring the appropriate device settings. This setup ensured that both models were ready for the training process, In the semi-supervised training phase, the process involved leveraging both pseudo-labelled and original datasets. The teacher model, equipped with the Mean

Teacher framework, generated pseudo-labels for the unlabelled data. These pseudo-labels were used in conjunction with the labelled data to train the student model. During the training process, the student model was optimized using both CIoU (Complete Intersection over Union) loss and contrastive loss functions. The CIoU loss function, a refined version of the standard IoU loss, was computed to enhance bounding box regression by considering not just the overlap but also the centre distance and aspect ratio differences between the predicted and ground truth boxes similar to that implemented in the previous model. Mathematically, the CIoU loss was represented as:

$$CIoU = IoU - \frac{center\ distance}{enclosure\ diagonal + \varepsilon}$$

where IoU denotes the Intersection over Union score, centre distance represents the normalized distance between the centres of the predicted and target bounding boxes, enclosure diagonal refers to the diagonal length of the smallest enclosing box and is a small constant to prevent division by zero. The loss function aimed to minimize thereby improving the accuracy of bounding box predictions. The contrastive loss function was applied to ensure that feature representations for similar objects were close together while those for different objects were far apart. This was achieved by computing the similarity matrix between normalized feature vectors and applying a log-softmax operation to derive the loss, which enhanced the discriminative power of the feature embeddings. This continuous feedback loop was crucial for achieving convergence and robustness in the semi-supervised learning framework.

3.6 Visualisations for attention maps:

Some additional explainability settings were incorporated and experimented along with these models. the focus was on visualizing attention maps generated by a neural network model. The visualisation_attention function was developed to display the original image alongside its corresponding attention map, facilitating an understanding of the areas of the image on which the model concentrates. To enable visualization, the function first converted the input image tensor from the format [C, H, W] to [H, W, C] using image.permute (1, 2, 0) and then transformed it into a NumPy array with .cpu() .numpy(). The attention map, which was squeezed to remove singleton dimensions and converted to a NumPy array, was normalized to the range [0, 1] using the formula:

$$normalisedmap = \frac{attentionmap - attentionmap.\min()}{attentionmap.() - attentionmap.\min()}$$

This normalization process ensured that the attention map was suitable for visualization. Two subplots were subsequently created using Matplotlib: the first subplot displayed the original image, while the second subplot overlaid the attention map on the image using the jet colormap with an alpha transparency of 0.5. The plt.axis('off') function was utilized to remove axis labels and ticks, resulting in a cleaner presentation.

The visualize_attention_from_model function was designated to extract and visualize attention maps from the model. The model was set to evaluation mode using model.eval() to disable dropout and batch normalization layers, ensuring consistent results during inference. The function processed batches from the dataloader, transferring images to the appropriate device and generating attention maps through the model. Each image and its corresponding attention map were then passed to the visualize_attention function for visualization. The iteration was halted after processing one image and attention map, as indicated by the break statement.

The main execution block was responsible for initializing the dataset, dataloader, and model, and for invoking the visualize_attention_from_model function to carry out the visualization. The validation dataset was loaded using the KittiDataset class, with transformations applied including resizing, normalization, and tensor conversion. The DataLoader was configured to load data in batches. The model was loaded from a specified path using torch.load() and transferred to the appropriate device (GPU if available). Finally, the visualize_attention_from_model function was executed to process the data and visualize the attention maps, thereby providing insights into the model's focus during inference.

3.7 Conclusion:

In summary this research is designed to address the complexities of multi-object detection in autonomous vehicles, aligning closely with the overarching research objectives. The experimental design incorporates a comprehensive evaluation framework utilizing metrics such as precision, recall, F1 score, and Mean Average Precision (mAP) to assess model performance, alongside specific metrics related to explainability, such as attention map visualizations. Baseline comparisons are established against existing techniques, including traditional supervised learning models and other semi-supervised approaches, to highlight the advancements offered by the proposed hybrid methodology.

The experimental setup is robust, leveraging high-performance hardware specifications and a structured training procedure that includes hyperparameter tuning to optimize model performance. Performance analysis is conducted through statistical tests and visualizations, ensuring a thorough understanding of the model's capabilities. Robustness testing is integral to the methodology, with the model evaluated across various scenarios, including noise levels and data distribution shifts, to ensure its reliability in real-world applications.

The explainability assessment is a critical component, employing user studies and expert evaluations to gauge the quality and utility of the model's explanations, thereby enhancing trust in its predictions. Anticipated challenges, such as computational limitations and the integration of diverse techniques, are proactively addressed through strategic mitigation measures, ensuring the research outcomes are both robust and reliable.

Chapter 4: Results

This section presents the outcomes of the experiments conducted to evaluate the effectiveness of the proposed methodologies for multi-object detection in autonomous vehicles

4.1 YOLO model Performance:

The YOLOv5 model was applied to the KITTI dataset, demonstrating its effectiveness in multi-object detection tasks. The model was evaluated based on key performance metrics: precision, recall, and F1 score. These metrics were calculated to gauge the accuracy and reliability of the model's predictions: The YOLOv5 model demonstrated strong performance in object detection across the KITTI dataset. The model achieved a mean Average Precision (mAP@0.5) of 0.763, reflecting a robust capability to detect and localize objects with a moderate level of precision and recall. The F1 score, peaking at 0.73 at a confidence threshold of 0.099, underscores a balanced trade-off between precision and recall. Specifically, the model showed high average precision (AP) for "Car" (0.882) and "Truck" (0.892), indicating excellent detection accuracy for these categories. Conversely, the model struggled with "Van" (AP = 0.728) and "Misc" (AP = 0.549), showing lower detection performance for these less frequent classes. Precision-recall curves indicated that while precision for cars and trucks remained high, precision dropped more rapidly for vans and miscellaneous objects as recall increased.

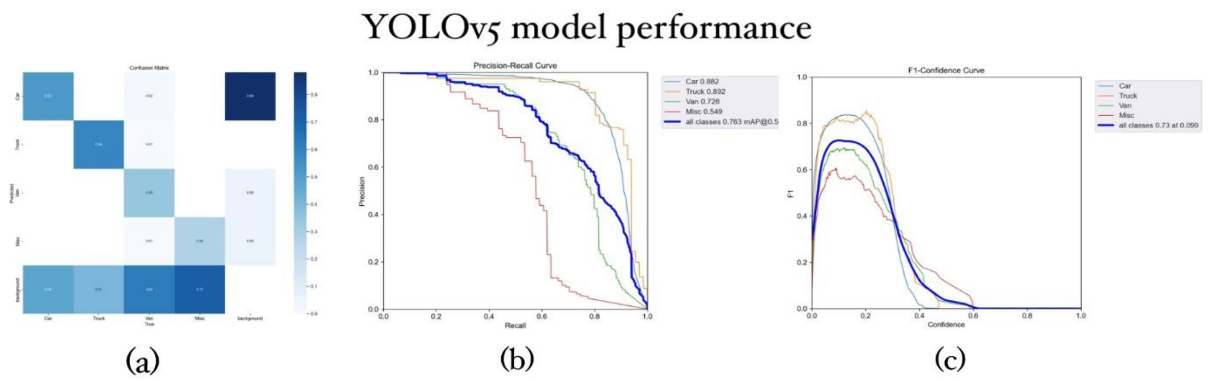

Figure 2. yolov5s results

4.1.1 Post Fine-Tunning Results:

After fine-tuning, notable improvements were observed. The mAP@0.5 increased slightly to 0.755, and the F1 score at the optimal confidence threshold of 0.143 indicated better balance in precision and recall. Enhanced performance was evident in the detection of "Car" (TP = 0.59) and "Truck" (TP = 0.68), with reduced false positives and false negatives. However, "Van" and "Misc" categories still exhibited challenges, with moderate improvements but persistent issues in detection accuracy. The precision-recall and F1-confidence curves show that the model achieved high performance for "Car" and "Truck," with improvements in the precision-recall trade-off but still requires further refinement for "Van" and "Misc."

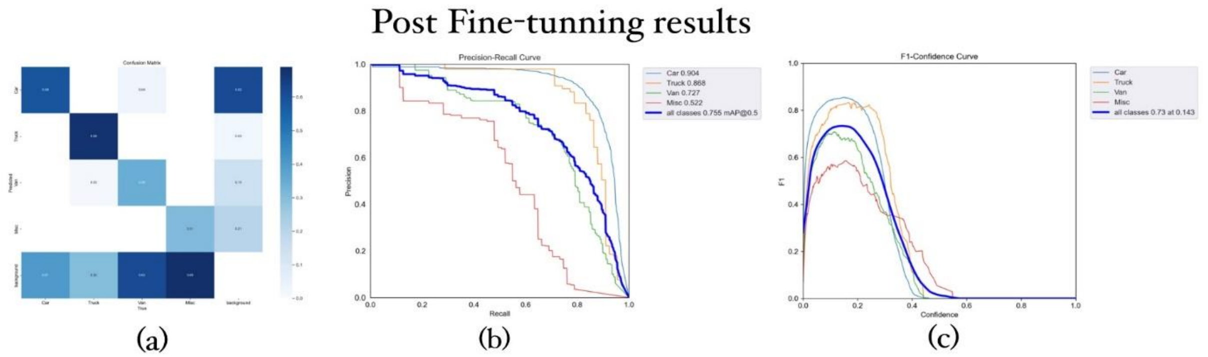

Figure 3 post fine-tunning of Yolov5 training

4.1.2 Validation Set Performance:

On the validation set, the YOLOv5 model exhibited exceptional performance with a near-perfect mAP@0.5 of 0.993, reflecting the model's ability to generalize well across unseen data. The precision-recall curves for all classes demonstrated high performance, with curves resembling a near-perfect square, indicating that the model maintained high precision and recall across various thresholds. The F1-confidence curve further confirmed that the model's performance was outstanding across all classes, achieving an F1-score of 1.00 at a confidence threshold of 1.000.

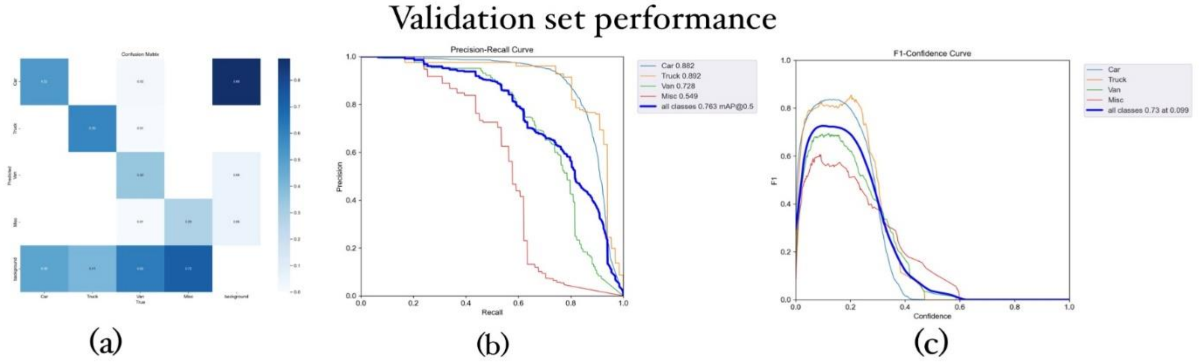

Figure 4 validation set performance using yolo weights

4.2 Pseudo-Labelling Outcomes:

Pseudo-labeling was employed to augment the training dataset by generating additional labels for previously unlabelled data, effectively increasing the volume and diversity of training samples. This augmentation helped improve the model's generalization capabilities and overall performance metrics, as seen in enhanced precision, recall, and F1 scores. The spatial attention mechanism provided further insights into the model's focus areas during object detection. Attention maps generated during this process revealed that the model primarily concentrated on central regions of the images, especially around vehicles, correctly identifying objects of interest while effectively suppressing irrelevant background information. This focused attention distribution indicates that the model is not only distinguishing between relevant and irrelevant elements but is also considering multiple aspects of the detected objects, leading to more robust predictions.

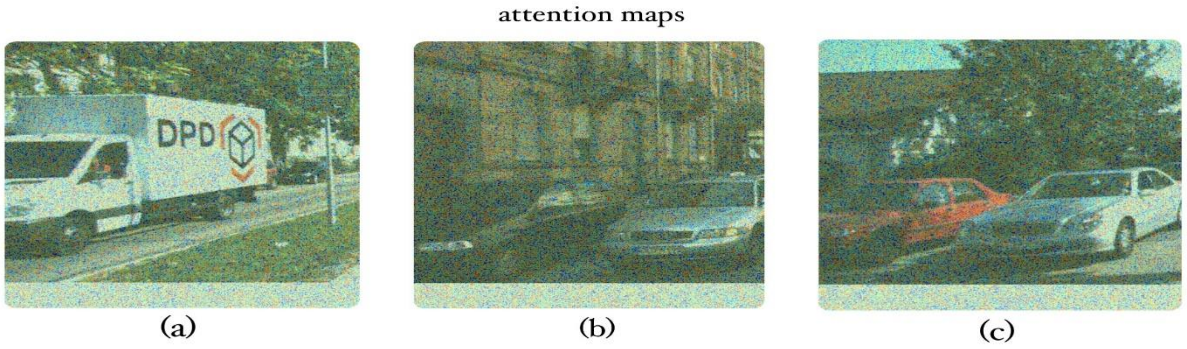

Figure 5 attention maps created using pseudo-labels

Moreover, the alignment between the attention maps and the pseudo-labelled data indicates that the model's attention is effectively guiding its predictions where it is most confident, validating the reliability of the pseudo-labels. Discrepancies, if any, between the attention maps and pseudo labels could signal potential areas where the model might be overconfident or misled by noisy data. To further enhance the model's performance and interpretability, additional analyses could include overlaying attention maps on original images to better visualize the regions of interest, correlating prediction confidence scores with attention regions to assess consistency, and experimenting with different attention mechanisms to optimize the model's focus on critical features. Such comprehensive analysis not only reinforces the effectiveness of the current attention mechanism but also provides a foundation for exploring further refinements in model training and prediction strategies.

4.3 Consistency Regularisation and Contrastive Regularisation outcomes:

The implementation of consistency regularization and contrastive regularization techniques was initially intended to further improve the robustness and generalization capabilities of the YOLOv5 model. However, significant challenges were encountered during the integration process, particularly involving tensor size mismatches and other complex technical issues. Despite extensive efforts to troubleshoot and resolve these issues, including multiple iterations of code adjustments and optimization attempts, the effective application of these advanced regularization techniques could not be achieved within the current scope of the study.

Given these difficulties, the present results section focuses on the successful outcomes obtained through the YOLOv5 model and pseudo-labelling methods. The challenges faced during the implementation of consistency and contrastive regularization have highlighted the need for more refined approaches and deeper technical exploration.

Chapter 5: Discussions:

This study addresses critical challenges in the field of autonomous driving by enhancing object detection capabilities through innovative methodologies. It emphasizes the significance of semi-supervised learning (SSL) techniques, which leverage both labelled and unlabelled data to enhance model performance while reducing the reliance on extensive manual annotation. This approach is particularly crucial for autonomous vehicles, where precise and real-time object detection is essential for ensuring safety and operational efficiency.

The key contributions of this research include the development and evaluation of three SSL-based models designed to enhance object detection in autonomous driving scenarios. The first model involved a hybrid SSL approach that integrates pseudo-labelling with consistency regularization using the YOLOv5 model. This combination enabled the model to leverage both labelled and pseudo-labelled data, enhancing its capacity to generalize across various environments. Consistency regularization was employed to enforce stability in the model's predictions across different augmented versions of the same input, which helped in reducing noise and error accumulation that typically arises from incorrect pseudo-labels. Furthermore, this model incorporated spatial explainability techniques enhancing the model's interpretability by providing visual insights into the decision-making process.

The **second model** introduced an innovative approach by incorporating contrastive regularization within the Mean Teacher framework. This method further strengthened the model's feature representation capabilities by ensuring the model distinguishes between similar and dissimilar instances. Moreover, the study placed significant emphasis on the importance of model explainability, particularly in safety-critical applications like autonomous driving. Model interpretability is crucial for regulatory compliance, user trust, and the broader adoption of AI systems in autonomous vehicles. The integration of spatial attention mechanisms provided a valuable tool for enhancing the transparency of model decisions, allowing developers and stakeholders to visualize where the model focuses when making decisions. Techniques such as Local Interpretable Model-Agnostic Explanations (LIME) and SHapley Additive exPlanations (SHAP) have been effective in various machine learning tasks. Still, they face limitations in complex multi-output scenarios, such as those typical of object detection tasks. The use of attention maps, as advocated by this research, provided deeper insights into the decision-making process, revealing the model's focus areas and contributing to better understanding and debugging of model behaviour

However, the research also identified several technical challenges that warrant further investigation. The integration of advanced regularization techniques, such as consistency and contrastive regularization, encountered practical difficulties, particularly related to tensor size mismatches during the model training phase. These challenges underscore the complexity involved in balancing model accuracy, computational efficiency, and the adaptability required for real-time object detection systems. Addressing these technical issues will be essential for future studies aiming to refine the proposed methodologies and enhance their applicability in real-world autonomous driving scenarios

In conclusion, this research contributes to advancing computer vision methodologies within the realm of autonomous driving by addressing key issues in SSL, improving object detection accuracy, and enhancing model interpretability. The findings suggest that the proposed hybrid approach offers a scalable and efficient solution to the challenges of large-scale data labelling in machine learning and deep learning applications. This lays a foundation for more effective AI training processes in the field of autonomous vehicles.

Chapter 6: Conclusion:

This research presents significant advancements through the introduction of innovative methodologies that utilize semi-supervised learning techniques. The development includes two hybrid models: the first integrates pseudo-labelling, consistency regularization, and model explainability methods, while the second combines the Mean Teacher framework with contrastive regularization for semi-supervised learning. These contributions are valuable for future research. Using the KITTI dataset, the study demonstrates the effectiveness of the YOLOv5 model in real-time applications and addresses critical issues such as error accumulation and computational inefficiencies. The varying outputs of YOLOv5 highlight the need for more stable models, which could improve overall performance. Additionally, the integration of spatial attention mechanisms enhances model interpretability, offering crucial insights into decision-making processes essential for safety-critical environments. The findings emphasize the potential of these advanced techniques to support the development of reliable and efficient object detection systems, setting the stage for future advancements in autonomous driving technology. This research establishes a robust foundation for further exploration and refinement of semi-supervised methodologies, contributing to the overarching goal of improving the safety and operational efficiency of autonomous vehicles.

Chapter 7: References

[1] J. Kim, A. Rohrbach, T. Darrell, J. Canny, and Z. Akata, "Textual Explanations for Self-Driving Vehicles," Jul. 2018, [Online]. Available: http://arxiv.org/abs/1807.11546

[2] A. Majumdar, S. Sarma, B. Tiple, A. Mankar, and S. Satone, "2D Object Detection for Autonomous Vehicles using Transfer Learning." [Online]. Available: https://ssrn.com/abstract=4145291

[3] A. Gupta, A. Anpalagan, L. Guan, and A. S. Khwaja, "Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues," *Array*, vol. 10, p. 100057, Jul. 2021, doi: 10.1016/j.array.2021.100057.

[4] S. K. Pal, A. Pramanik, J. Maiti, and P. Mitra, "Deep learning in multi-object detection and tracking: state of the art," *Applied Intelligence*, vol. 51, no. 9, pp. 6400–6429, Sep. 2021, doi: 10.1007/s10489-021-02293-7.

[5] D.-H. Lee, "Pseudo-Label : The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks."

[6] S. Sajid, Z. Aziz, O. Urmonov, and H. W. Kim, "Improving Object Detection Accuracy with Self-Training Based on Bi-Directional Pseudo Label Recovery," *Electronics (Switzerland)*, vol. 13, no. 12, Jun. 2024, doi: 10.3390/electronics13122230.

[7] J. Han *et al.*, "SODA10M: A Large-Scale 2D Self/Semi-Supervised Object Detection Dataset for Autonomous Driving," Jun. 2021, [Online]. Available: http://arxiv.org/abs/2106.11118

[8] J. Zhang, H. Liu, and J. Lu, "A semi-supervised 3D object detection method for autonomous driving," *Displays*, vol. 71, Jan. 2022, doi: 10.1016/j.displa.2021.102117.

[9] *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020.

[10] M. T. Ribeiro, S. Singh, and C. Guestrin, "Model-Agnostic Interpretability of Machine Learning," Jun. 2016, [Online]. Available: http://arxiv.org/abs/1606.05386

[11] S. Sharma, K. Kaushik, R. Sharma, and N. Chaturvedi, "IJFANS INTERNATIONAL JOURNAL OF FOOD AND NUTRITIONAL SCIENCES Explainable Artificial Intelligence (XAI)," 2012.

[12] A. B. Goldberg, X. Zhu, A. Singh, Z. Xu, and R. Nowak, "Multi-Manifold Semi-Supervised Learning," 2009.

[13] Y. Ouali, C. Hudelot, and M. Tami, "An Overview of Deep Semi-Supervised Learning," Jun. 2020, [Online]. Available: http://arxiv.org/abs/2006.05278

[14] I. Nurcahyani and J. W. Lee, "Role of machine learning in resource allocation strategy over vehicular networks: A survey," Oct. 01, 2021, *MDPI*. doi: 10.3390/s21196542.

[15] J. E. van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Mach Learn*, vol. 109, no. 2, pp. 373–440, Feb. 2020, doi: 10.1007/s10994-019-05855-6.

[16] L. Fernandes, J. N. D. Fernandes, M. Calado, J. R. Pinto, R. Cerqueira, and J. S. Cardoso, "Intrinsic Explainability for End-to-End Object Detection," *IEEE Access*, vol. 12, pp. 2623–2634, 2024, doi: 10.1109/ACCESS.2023.3347038.

[17] S. K. Jaiswal and R. Agrawal, "A Comprehensive Review of YOLOv5: Advances in Real-Time Object Detection," *International Journal of Innovative Research in Computer Science and Technology*, vol. 12, no. 3, pp. 75–80, May 2024, doi: 10.55524/ijircst.2024.12.3.12.

[18] T. Saidani, "Deep Learning Approach: YOLOv5-based Custom Object Detection," *Engineering, Technology and Applied Science Research*, vol. 13, no. 6, pp. 12158–12163, Dec. 2023, doi: 10.48084/etasr.6397.

[19] A. A. Alsuwaylimi, R. Alanazi, S. M. Alanazi, S. M. Alenezi, T. Saidani, and R. Ghodhbani, "Improved and Efficient Object Detection Algorithm based on YOLOv5," *Engineering, Technology and Applied Science Research*, vol. 14, no. 3, pp. 14380–14386, Jun. 2024, doi: 10.48084/etasr.7386.

[20] P. Azevedo and V. Santos, "YOLO-Based Object Detection and Tracking for Autonomous Vehicles Using Edge Devices," in *Lecture Notes in Networks and Systems*, Springer Science and Business Media Deutschland GmbH, 2023, pp. 297–308. doi: 10.1007/978-3-031-21065-5_25.

[21] G. Li, Z. Ji, X. Qu, R. Zhou, and D. Cao, "Cross-Domain Object Detection for Autonomous Driving: A Stepwise Domain Adaptative YOLO Approach," *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 3, pp. 603–615, Sep. 2022, doi: 10.1109/TIV.2022.3165353.

[22] B. Xu, M. Chen, W. Guan, and L. Hu, "Efficient Teacher: Semi-Supervised Object Detection for YOLOv5," Feb. 2023, [Online]. Available: http://arxiv.org/abs/2302.07577

[23] Z. Chen, W. Zhang, X. Wang, K. Chen, and Z. Wang, "Mixed Pseudo Labels for Semi-Supervised Object Detection," Dec. 2023, [Online]. Available: http://arxiv.org/abs/2312.07006

[24] S. Hu, C.-H. Liu, J. Dutta, M.-C. Chang, S. Lyu, and N. Ramakrishnan, "PseudoProp: Robust Pseudo-Label Generation for Semi-Supervised Object Detection in Autonomous Driving Systems."

[25] D. Chun, S. Lee, and H. Kim, "USD: Uncertainty-Based One-Phase Learning to Enhance Pseudo-Label Reliability for Semi-Supervised Object Detection," *IEEE Trans Multimedia*, vol. 26, pp. 6336–6347, 2024, doi: 10.1109/TMM.2023.3348662.

[26] S. Sajid, Z. Aziz, O. Urmonov, and H. W. Kim, "Improving Object Detection Accuracy with Self-Training Based on Bi-Directional Pseudo Label Recovery," *Electronics (Switzerland)*, vol. 13, no. 12, Jun. 2024, doi: 10.3390/electronics13122230.

[27] K. Wang, L. Zhang, Q. Xia, L. Pu, and J. Chen, "Computing and Applications on October 26th, 2021. See the published version," 2021, doi: 10.21203/rs.3.rs-277735/v1.

[28] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results."

[29] Z. Diffallah, H. Ykhlef, and H. Bouarfa, "Mean Teacher for Weakly Supervised Polyphonic Sound Event Detection: An Empirical Study," in *2022 7th International Conference on Image and Signal Processing and their Applications, ISPA 2022 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/ISPA54004.2022.9786322.

[30] S. Cao, D. Joshi, L.-Y. Gui, and Y.-X. Wang, "Contrastive Mean Teacher for Domain Adaptive Object Detectors." [Online]. Available: https://github.com/Shengcao-Cao/CMT

[31] J. Jeong, S. Lee, J. Kim, and N. Kwak, "Consistency-based Semi-supervised Learning for Object Detection." [Online]. Available: https://github.com/soo89/CSD-SSD

[32] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations," 2020. [Online]. Available: https://github.com/google-research/simclr.

[33] D. Lee, K. Brain, S. Kim, I. Kim, M. Cho, and W.-S. Han, "Contrastive Regularization for Semi-Supervised Learning."

[34] K.-B. Nguyen and S. Korea, "Debiasing, calibrating, and improving Semi-supervised Learning performance via simple Ensemble Projector." [Online]. Available: https://github.com/beandkay/EPASS.

[35] A. Holzinger, A. Saranti, C. Molnar, P. Biecek, and W. Samek, "Explainable AI Methods - A Brief Overview," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Science and Business Media Deutschland GmbH, 2022, pp. 13–38. doi: 10.1007/978-3-031-04083-2_2.

[36] B. Abdollahi and O. Nasraoui, "Transparency in Fair Machine Learning: the Case of Explainable Recommender Systems."

[37] V. B. Truong, T. T. H. Nguyen, V. T. K. Nguyen, Q. K. Nguyen, and Q. H. Cao, "Towards Better Explanations for Object Detection," Jun. 2023, [Online]. Available: http://arxiv.org/abs/2306.02744

[38] M. Merry, P. Riddle, and J. Warren, "A mental models approach for defining explainable artificial intelligence," *BMC Med Inform Decis Mak*, vol. 21, no. 1, Dec. 2021, doi: 10.1186/s12911-021-01703-7.

[39] J. Dong, S. Chen, M. Miralinaghi, T. Chen, and S. Labi, "Development and testing of an image transformer for explainable autonomous driving systems," *Journal of Intelligent and Connected Vehicles*, vol. 5, no. 3, pp. 235–249, Oct. 2022, doi: 10.1108/JICV-06-2022-0021.

[40] J. Wang *et al.*, "When, Where and How Does it Fail? A Spatial-Temporal Visual Analytics Approach for Interpretable Object Detection in Autonomous Driving," *IEEE Trans Vis Comput Graph*, vol. 29, no. 12, pp. 5033–5049, Dec. 2023, doi: 10.1109/TVCG.2022.3201101.

[41] C. Nogueira, L. Fernandes, J. N. D. Fernandes, and J. S. Cardoso, "Explaining Bounding Boxes in Deep Object Detectors Using Post Hoc Methods for Autonomous Driving Systems," *Sensors*, vol. 24, no. 2, Jan. 2024, doi: 10.3390/s24020516.

[42] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, Sep. 2013, doi: 10.1177/0278364913491297.

[43] Y. Liao, J. Xie, and A. Geiger, "KITTI-360: A Novel Dataset and Benchmarks for Urban Scene Understanding in 2D and 3D," *IEEE Trans Pattern Anal Mach Intell*, vol. 45, no. 3, pp. 3292–3310, Mar. 2023, doi: 10.1109/TPAMI.2022.3179507.

[44] S. Du, B. Zhang, and P. Zhang, "Scale-Sensitive IOU Loss: An Improved Regression Loss Function in Remote Sensing Object Detection," *IEEE Access*, vol. 9, pp. 141258–141272, 2021, doi: 10.1109/ACCESS.2021.3119562.

[45] D. Lee, K. Brain, S. Kim, I. Kim, M. Cho, and W.-S. Han, "Contrastive Regularization for Semi-Supervised Learning."