Evaluating modern computer vision techniques for Shape Language classification in meetings

Automatic understanding of meetings and negotiations

Sorana Stan

Supervisor(s): Stephanie Tan[1], Edgar Salas Gironés[1]

[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
January 26, 2025

Name of the student: Sorana Stan
Final project course: CSE3000 Research Project
Thesis committee: Stephanie Tan, Edgar Salas Gironés, Sole Pera

Abstract

Meetings represent a key component of collaboration in the workplace, serving purposes like brainstorming, discussion, and negotiation. Despite their importance, reaching a consensus among participants can frequently be difficult because different people can leave the debate with different perspectives. In order to promote efficient communication and decision-making in organisational contexts, the use of the Shape Language is proposed. The Shape Language consists of shapes that people can use in meetings in order to represent abstract ideas, that would be difficult to represent by only words. In order to track how people interact with these objects, computer vision tools can be used. This study aims to explore the current existing computer vision tools for segmenting and classifying objects in meetings, aiming to find limitations in how well these models are able to recognize objects in the context of meetings and negotiations. Results of this study show that after fine-tuning four models on the custom dataset, they can recognize the three shapes provided as classes in most of the cases, but still make mistakes when assigning classes, or miss objects that they should classify all together, which show limitations of these modern tools.

Keywords: Meetings and negotiations, Shape Language, computer vision tools, object segmentation, object classification, fine-tuning models, custom dataset, limitations, human-object interaction

1 Introduction

Advancements in computer vision have provided opportunities for analysing and monitoring human behaviour in a variety of settings. Workplace meetings and negotiations are among the most essential settings for effective communication and decision-making. However, these encounters frequently meet obstacles, such as different interpretations of discussion points, making consensus-building difficult. The introduction of Shape Language [8], a system of geometric shapes such as spheres, cubes, and pyramids is intended to address this issue. The Shape Language promotes collaboration and comprehension by providing participants with concrete tools for representing abstract ideas. However, incorporating Shape Language into meeting contexts presents a substantial challenge: effectively identifying and categorising these shapes in dynamic surroundings, using computer vision tools. An example of how a frame containing the Shape Language objects look, can be seen in figure 1.

Recent improvements in computer vision, particularly in object detection and segmentation, offer intriguing methods for tackling this problem. Tools such as the Segment Anything Model (SAM) [3] and its successor [5] have revolutionized object segmentation and tracking in images and videos. Furthermore, object detection, recognized as one of the most fundamental yet challenging problems in computer vision,

Figure 1: Example of a frame from the dataset containing the Shape Language objects

has gained significant attention in recent years [10]. The current landscape offers not only the capability to perform these tasks but also a variety of models to choose from, each presenting unique trade-offs in terms of performance, accuracy, and application.

Despite their innovative potential, these tools exhibit notable limitations, primarily due to the novel state of the technology. Their effectiveness has not yet been thoroughly examined in specialized contexts, such as negotiations that employ human-object interaction. Additionally, there is a lack of detailed evaluations regarding the accuracy, limitations, and trade-offs of these models in specific application scenarios (since most comparisons were done on generic object datasets [9]), leaving researchers with limited guidance on the suitability of specific models for particular tasks.

The following work addresses this gap by evaluating the performance of novel object detection models in recognising and classifying Shape Language shapes during meetings. By fine-tuning these models on a custom dataset, we are determining their strengths, limitations, and applicability for this specific scenario. The findings contribute to expanding our understanding of how modern computer vision tools can improve collaborative processes, providing a foundation for improved human-object interaction technologies in organisational contexts.

The remainder of this paper is organized as follows. Section 2 provides information about the literature used in order to build this paper. Section 3 provides a detailed overview of the methodology, including the frame annotation process, the selection and fine-tuning of object detection models, and the metrics used for evaluation. Section 4 presents the experimental setup, results, and a comprehensive discussion of the performance of the models. Section 5 provides a discussion that situates the results within the context of prior work, offering insights into the capabilities and limitations of the models. Section 6 outlines the limitations of the study, in-

cluding constraints related to the dataset and computational resources. Section 7 addresses responsible research, focusing on ethical considerations, reproducibility, and the implications of the findings. Finally, Section 8 concludes the paper by summarizing the contributions and suggesting directions for future research.

2 Related Literature

Advances in object detection and segmentation have been critical to computer vision research. Several basic and recent works have influenced the discussion of Shape Language classification in meetings.

Object detection has advanced dramatically over the last two decades, with deep learning algorithms making significant contributions. Zhao et al. [8] presented a detailed overview of object detection strategies, emphasising the transition from traditional approaches to new deep learning frameworks. Similarly, Zou et al. [9] described the progress in object detection over the last 20 years, emphasising the challenges and breakthroughs in dynamic contexts such as real-world meetings.

Among the multitude of options when it comes to object detection models, four particular ones have been chosen, after careful consideration of their different characteristics:

- **YOLOv8 (You Only Look Once):** Yolov8 builds upon the foundation of the YOLO family of object detection models, originally introduced by Redmon et al. (2016) [6]. It is a single-stage object detection model that uses a single CNN architecture to divide images into grid cells and predict bounding boxes and classes directly for each cell. Yolov8 is characterized by its speed and accuracy, making it suitable for real-time applications, as well as imbalanced datasets. These qualities make it a strong candidate for applications requiring efficient and accurate detection in dynamic environments. The main idea of how Yolo works can be seen in figure 2.

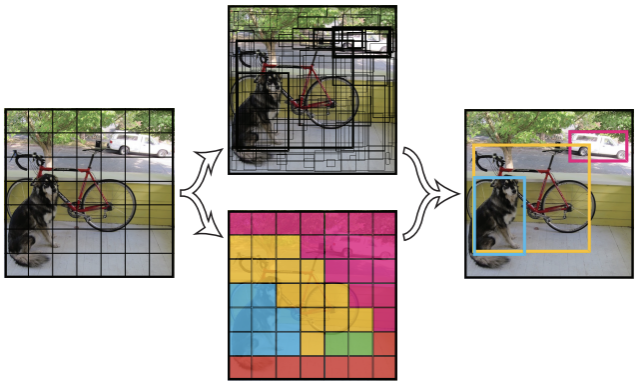

Figure 2: Main idea of Yolo [6]

- **SSD: (Single Shot Multibox Detector):** SSD is a single-stage object detection model that employs a single CNN to predict bounding boxes and classes for multiple scales in a single pass [4]. It is fast and efficient, with a simple architecture, making it ideal for real-time applications. However, SSD struggles with small objects and has lower accuracy compared to RCNN variants, which can limit its effectiveness in scenarios with fine object details. The architecture of SSD can be seen in figure 3.

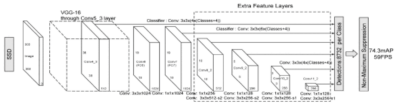

Figure 3: SSD architecture [4]

- **RCNN:** is a two-stage object detection model proposed by Girshick et al. [2] in 2014, that uses region proposals generated via Selective Search for per-region feature extraction and classification. It achieves high accuracy and performs well with complex and small objects, making it suitable for varied object sizes. However, RCNN is slow, computationally expensive, and not end-to-end trainable, which limits its practicality for real-time applications. The RCNN architecture is described in figure 4.

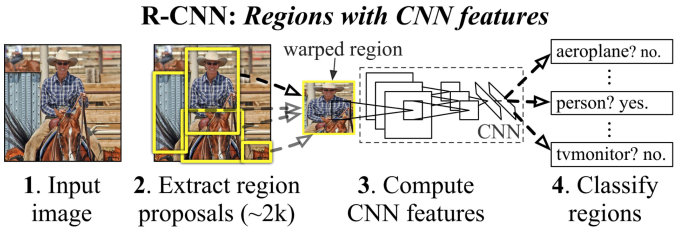

Figure 4: RCNN Architecture [2]

- **RetinaNet:** is a single-stage object detection model that uses a single CNN with a Feature Pyramid Network (FPN) for multi-scale feature extraction.[7] It predicts bounding boxes and classes while achieving a balance between accuracy and speed. RetinaNet performs well with imbalanced datasets and offers strong performance overall. However, it is slower than YOLO and SSD and has a higher computational cost compared to simpler single-stage models, making it less suitable for real-time scenarios. The RetinaNet architecture is shown in figure 5.

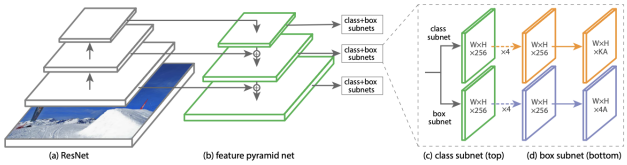

Figure 5: RetinaNet architecture [7]

A summary of the features of these four models, as well as the tradeoffs and differences between them, which con-

tributed to the choice of fine-tuning these specific models can be found in figure 6. The comparison of speed, and accuracy of the models was made relative to eachother, in order to give a qualitative estimate of how the models perform when tested on generic datasets.

	RCNN (Region-based Convolutional Neural Network)	SSD (Single Shot MultiBox Detector)	YOLOv8 (You Only Look Once)	RETINANET
Paradigm	Two-Stage	Single-Stage	Single-Stage	Single-Stage
Architecture	• Region proposals via Selective Search. • Per-region feature extraction. • Separate classification model.	• Single CNN. • Predicts bounding boxes and classes for multiple scales in one pass.	• Single CNN. • Divides image into grid cells. • Predicts bounding boxes and classes directly for each cell.	• Single CNN. • Feature Pyramid Network (FPN) for multi-scale feature extraction. • Predicts boxes and classes.
Speed	Slow	Fast	Very fast	Moderate
Accuracy	High	Moderate	High	High
Advantages	• High accuracy. • Strong for complex and small objects. • Good for varied object sizes.	• Fast and efficient. • Simple architecture. • Real-time capable.	• Balances accuracy and speed. • Strong performance for imbalanced datasets.	• Balances accuracy and speed. • Strong performance for imbalanced datasets.
Disadvantages	• Very slow. • Computationally expensive. • Not end-to-end trainable.	• Struggles with small objects. • Lower accuracy than RCNN variants.	• Early versions struggle with small objects and crowded scenes. • Localization issues.	• Slower than YOLO and SSD. • Higher computational cost than simpler single-stage models.

Figure 6: Summary of comparison of different features of the four models chosen

3 Methodology

In figure 7, a high-level overview of the workflow, from annotating images, until analysing results is provided. This chapter will now provide a detailed explanation of each of these phases.

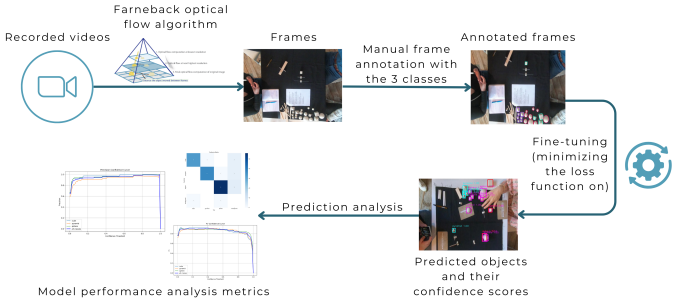

Figure 7: High-level workflow pipeline [1]

3.1 Frame Annotation Process

To facilitate effective analysis of the target models, the videos in the dataset were preprocessed by converting them into frames, which serve as discrete images for further analysis. Rather than uniformly sampling frames across the entire video, optical flow algorithms were employed to guide the frame extraction process. This approach prioritizes sampling frames in segments of the video where significant motion is detected, ensuring that dynamic and relevant content is captured with higher temporal resolution. By focusing on areas of substantial motion, this method not only enhances the efficiency of the process, but also significantly reduces the memory footprint required for storing the frames. Additionally, it minimizes the manual annotation workload by avoiding the inclusion of redundant or static frames. For computing the optical flow, the Farneback algorithm is calculated using the cv2.calcOpticalFlowFarneback method, of the cv2 python library. This works by modeling the pixel intensity neighborhood using a quadratic polynomial approximation. The intensity I(x) at a point x in the image is expressed as:

$$I(x) \approx x^\top Ax + b^\top x + c$$

where:

- x is the pixel location,
- A is a symmetric matrix representing the quadratic terms,
- b is a vector for the linear terms,
- c is a scalar constant for the intensity offset.

After that, given two consecutive frames,

$$I_1(x), I_2(x+d)$$

the algorithm estimates the displacement d (optical flow) by minimizing the squared difference between the two frames:

$$E(d) = \int [I_2(x+d) - I_1(x)]^2 \, dx$$

This is achieved by expanding the neighborhood polynomial model and using an iterative process to compute the motion field across the image.

Following frame extraction, a thorough annotation process was carried out. Each object relevant to the study, specifically the geometric shapes from the Shape Language (sphere, cube, and pyramid) was manually annotated and assigned its corresponding class label. These annotations provide the necessary ground truth for training and evaluating the models. An example of an annotated frame, depicting the three object classes, is shown in Figure 8. These annotated images form the foundational dataset for training the target models, enabling them to learn object detection and classification tasks.

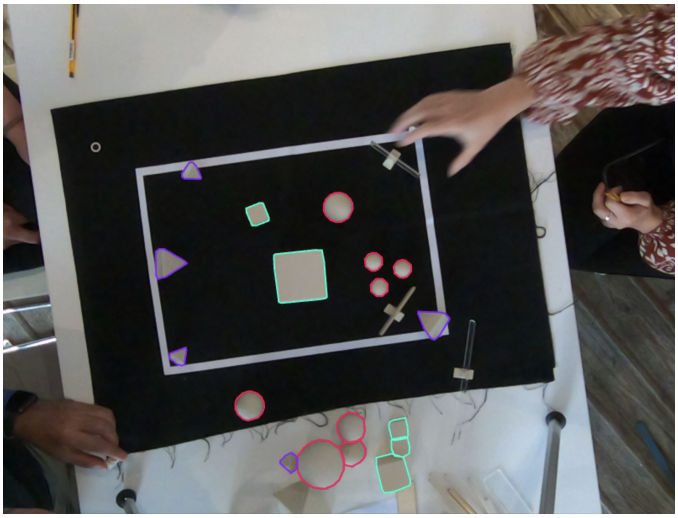

Figure 8: Example of an annotated image containing the three geometric object classes: sphere, cube, and pyramid.

3.2 The Target Models

With the annotated dataset prepared, four object detection and classification models were selected for fine-tuning: YOLOv8, SSD (Single Shot Multibox Detector), R-CNN and RetinaNET. The purpose of employing multiple models is to evaluate their comparative performance in detecting and classifying the three geometric object classes from the Shape Language, as annotated in the dataset.

During training, the models utilize the annotated images to learn the distinguishing features of each object class, ultimately producing bounding box predictions and class labels, along with confidence scores for each label, for unseen test images. When choosing the models, their different characteristics, along with their strengths and weaknesses were considered in order to make a decision.

3.3 Model Fine-Tuning

These models were then fine-tuned to identify and classify the geometric shapes using the annotated dataset. In order to evaluate the models, the predictions of the test set were plotted, along with some metrics (confusion matrix, F1 score, precision-confidence curve, recall-confidence curve, and precision-recall curve). The comparative analysis of these models provides insight into how well they perform in terms of precision, and flexibility in recognizing the Shape Language. An example of how the predictions of the models looked after fine-tuning can be found in figure 9

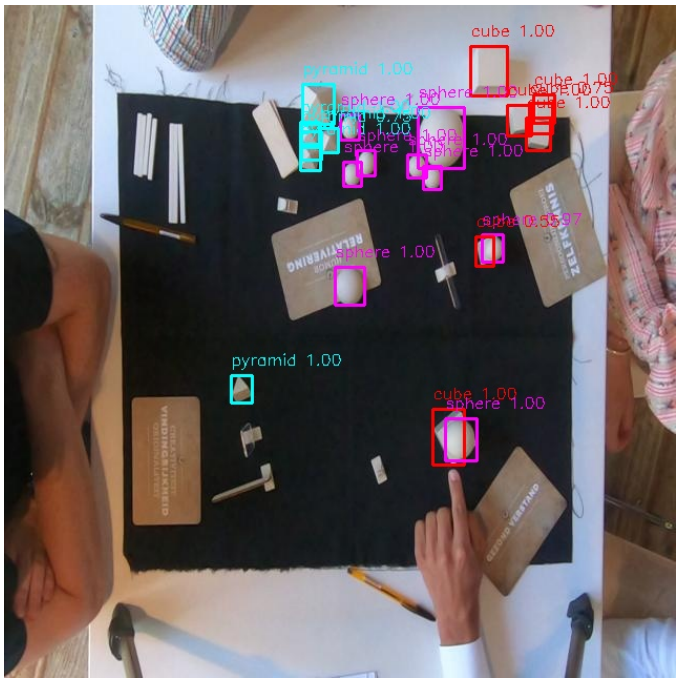

Figure 9: An example of a very confident RetinaNet test frame with predictions

.

4 Experimental Setup and Results

4.1 Experimental Setup

To ensure a robust and fair comparison of the four models, the following standardized experimental conditions were established:

- **Dataset Composition:** The dataset comprised 90 annotated images derived from four meeting scenarios, representing the Shape Language objects (sphere, cube, and pyramid). These images were selected after applying the optical flow algorithm on the 4 recordings, and they were distributed into three sets: 74 for training, 8 for validation, and 8 for testing. The training set was comprised of 1322 shapes, out of which: 330 cubes, 406 pyramids and 586 spheres.
- **Model Training Parameters:** Each model was fine-tuned over 150 epochs, and a confidence threshold of 0.5 was uniformly applied to filter predictions.
- **Data Representation:** To maximize diversity, the images captured objects in varied orientations, positions, and interactions within the meeting environment. An example on how the setup of the data recorded looks in the video can be seen in Figure 10.

Figure 10: Example of a frame from the dataset containing the Shape Language objects

- **Evaluation Metrics:** Performance was evaluated using standard metrics, including:
  - Confusion Matrix: It provides a detailed breakdown of model performance for each class. Useful for identifying which classes are being confused and where the model is struggling.
  - Precision-Recall Curves: Analyzes the trade-off between precision and recall at different classification thresholds. For example, high precision with low recall means the model is conservative, predicting fewer but more accurate positives.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

– F1 Score: Used to evaluate the model's performance when both precision and recall are important, especially in imbalanced datasets where one metric alone might be misleading. A high F1 score indicates that the model has a good balance of precision and recall, making it reliable for classification tasks.

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

4.2 Results - Confusion Matrices

The confusion matrices for the four models can be seen in Figure 11.

Figure 11: The confusion matrices showing the performance of the 4 models on the test set

In evaluating the performance of the four object detection models against the ground truth, RetinaNet demonstrated the most consistent and accurate results. RetinaNet detected 165 objects, slightly exceeding the ground truth of 154, but had the lowest number of misclassifications (17). It performed particularly well in detecting spheres (67), cubes (39), and pyramids (42), closely aligning with the respective ground truth values of 70, 41, and 43. RCNN can be thought of as the second best, detecting 163 objects with 26 misclassifications. Although it performed well in pyramid detection (40), it struggled with cube detection (31), which was significantly below the ground truth. YOLOv8 detected 146 objects, which is slightly under the ground truth, and achieving relatively low misclassification rates (20). However, it underperformed in detecting spheres (56) and pyramids (37). SSD, despite detecting 155 objects, had the poorest overall performance, with 30 misclassifications and substantial underperformance in detecting spheres (55) and pyramids (37). These results highlight RetinaNet as the most reliable model for accurate object detection in this context, while SSD showed the most limitations.

A summary of the numbers of objects detected for each model can be found in the table below:

Metric	Ground Truth	RCNN	SSD	YOLOv8	RetinaNet
Number of objects detected	154	163	155	146	165
Number of spheres detected	70	66	55	56	67
Number of cubes detected	41	31	33	33	39
Number of pyramids detected	43	40	37	37	42
Number of wrong objects	-	26	30	20	17

Table 1: Performance comparison of RCNN, SSD, YOLOv8, and RetinaNet models against the ground truth.

4.3 Results - Quantitative Performance Analysis using Recall, Precision and F1 Score

The performance of four object detection models was evaluated using confidence-F1, precision-recall, precision-confidence, and recall-confidence curves. Each model's capability to detect and classify the three Shape Language objects (sphere, cube, and pyramid) was analyzed. Below, a performance analysis per metric is provided.

F1-Confidence Curves

These curves show the F1 score, a harmonic mean of precision and recall, plotted against confidence thresholds. Higher F1 scores indicate better balance between precision (accuracy of positive predictions) and recall (coverage of actual positives).

The F1-Confidence scores per class, and combined can be seen in figure 12.

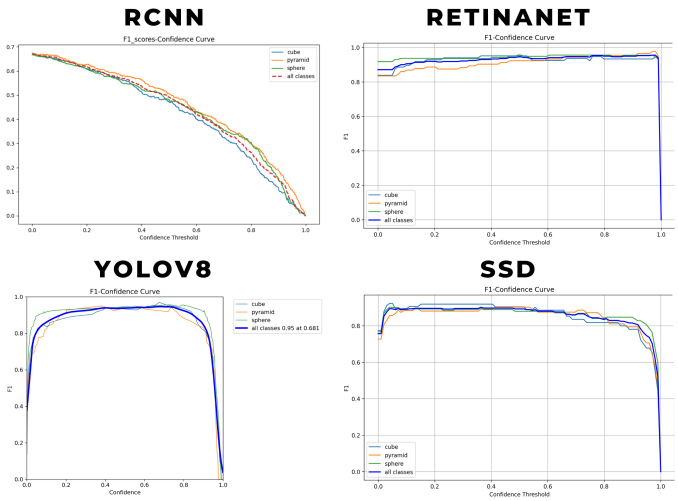

Figure 12: The F1-confidence scores on the 4 models on the test set

- **RetinaNet:** Exhibits high F1 scores across the majority of confidence thresholds, maintaining consistency across classes. The performance remains stable until very high confidence thresholds, indicating a robust model.
- **Yolov8:** Demonstrates very high F1 scores within an optimal confidence range (0.2–0.8). Peaks at a confidence threshold around 0.6-0.8 before slightly declining, indicating it is highly effective in precise detections within this confidence range.
- **SSD:** Shows similar results to RetinaNet in general trends, but with slightly lower performance and more variability in F1 scores among classes.
- **RCNN:** Performance decreases consistently with increasing confidence thresholds. Shows balanced but low F1 scores across all classes, suggesting weak detection capability for specific shapes at high confidence levels.

Precision

The following graphs illustrate how the model's precision (ratio of correct positive predictions to total positive predictions) changes with increasing confidence thresholds. Higher precision at a broader range of confidence thresholds indicates fewer false positives. The precision-confidence scores per class, and combined can be seen in figure 13.

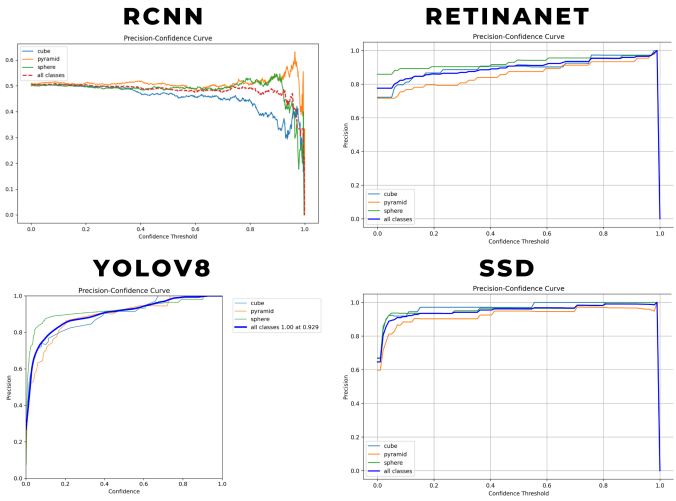

Figure 13: The precision-confidence scores on the 4 models on the test set

- **RetinaNet:** Precision increases consistently with higher confidence thresholds, showcasing robust performance at higher thresholds. Displays very good precision across all classes until extreme thresholds.
- **Yolov8:** Achieves high precision even at low confidence thresholds, with a peak near 0.8 confidence. Outperforms other models in precision consistency, especially at moderate thresholds.
- **SSD:** Precision trends are similar to RetinaNet but slightly lower across all confidence levels. Suffers from sharper precision drops at the highest thresholds.
- **RCNN:** Precision trends gradually decline as confidence thresholds increase, reflecting high false positives for lower-confidence predictions. Performance is generally weaker and more variable compared to other models.

Recall

The following graphs depict the recall (ratio of true positives to the sum of true positives and false negatives) at different confidence thresholds. Models with high recall across thresholds are better at detecting positive instances. The recall-confidence scores per class, and combined can be seen in figure 14.

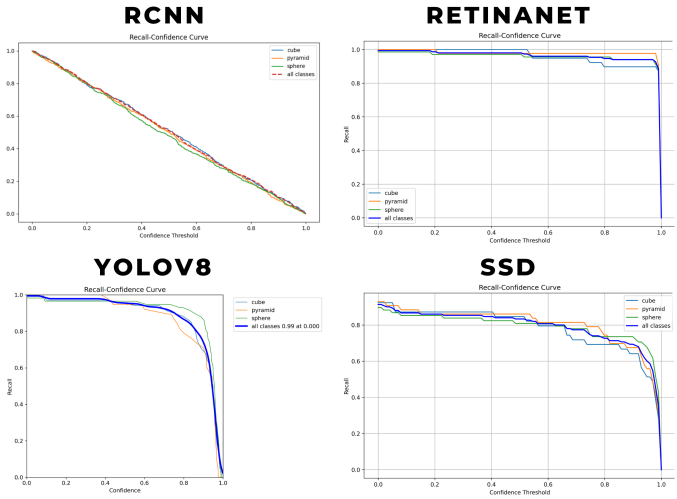

Figure 14: The recall-confidence scores on the 4 models on the test set

- **RetinaNet:** Displays high recall across most thresholds, with only a sharp drop at extreme confidence levels, indicating good sensitivity for all classes.
- **Yolov8:** Maintains high recall at lower thresholds but declines rapidly at higher thresholds, highlighting limitations in exhaustive detections, and a strong preference for lower confidence thresholds in maintaining sensitivity.
- **SSD:** Shows moderate recall, with some difficulty in detecting specific objects like spheres and pyramids.
- **RCNN:** Displays consistently declining recall as confidence thresholds increase.

Precision-Recall

These curves analyze the trade-off between precision and recall. A curve closer to the top-right corner indicates better performance, as it suggests high precision and recall across multiple thresholds. The precision-recall scores per class, and combined can be seen in figure 15.

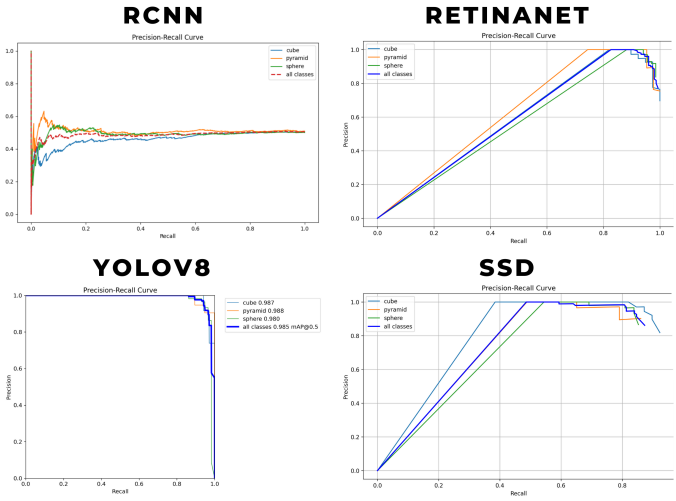

Figure 15: The precision-recall scores on the 4 models on the test set

- **RetinaNet:** High precision is maintained across a wide range of recall values, showing reliable detection across all classes. Sharp drop in precision occurs at high recall values for certain shapes, showing limitations under exhaustive detection conditions.
- **Yolov8:** Near-perfect precision is achieved at most recall values, with steep declines only at extreme recall levels. This indicates its suitability for highly accurate detections, with low tolerance for false positives.
- **SSD:** Precision trends align with those of RetinaNet but with sharper declines at high recall values. Shows reduced reliability compared to RetinaNet and YOLOv8, especially for specific classes.
- **RCNN:** Precision remains relatively low across all recall values, showing inefficiency in identifying objects without false positives. Performance is erratic at lower recall values, indicating inconsistency.

The comparative evaluation of the four models shows some differences in their ability to detect and classify the three Shape Language objects. RetinaNet consistently demonstrated the best overall performance, maintaining high F1 scores, precision, and recall across various confidence thresholds and object classes. YOLOv8 did well in precision-recall metrics, achieving near-perfect precision at moderate confidence thresholds, but its recall declined sharply at extreme thresholds, indicating sensitivity to specific detection conditions. SSD showed moderate performance, with acceptable F1 scores but lower precision and recall, particularly for certain shapes like spheres and pyramids, making it less reliable overall. RCNN exhibited the weakest performance across metrics, with significant variability in precision and recall, indicating challenges in detecting objects accurately, especially smaller or less prominent ones. Overall, RetinaNet emerged as the most balanced and robust model, followed by YOLOv8, while SSD and RCNN had some limitations in handling this custom dataset. A summary of the results can be seen in table 2:

Metric	RCNN	RetinaNet	YOLOv8	SSD
F1 Score (Overall)	Low	High	Very High (Optimal Range)	Moderate
Precision	Low	High	Very High	Moderate
Recall	Moderate	High	High (Declines at Extremes)	Moderate
Performance Consistency	Low	High	High	Moderate
Best Confidence Range	Low-Mid	Mid-High	Mid	Mid-High

Table 2: Performance Comparison of Object Detection Models

5 Discussion

By using advanced models such as Yolov8, SSD, RCNN, and RetinaNet, this study adds to the growing body of research on how computer vision may improve working environments. However, in order to fully comprehend the importance and limitations of the findings, they must be placed within a broader context.

5.1 Comparison to Previous Work

Traditional object detection research relies on generic datasets like COCO or ImageNet, which may not effectively capture the complexities of dynamic, real-world scenarios, such as meetings involving human-object interaction. Unlike previous studies, this study fine-tunes novel models, in order to test their effectiveness specifically in identifying geometric shapes in interactive situations. This approach emphasises the distinct issues presented by small dataset sizes, varying object orientations, and dynamic participant-object interactions.

While previous models such as RCNN and SSD have been verified on big, well-annotated datasets, the outcomes of this work highlight their limits in terms of computing efficiency and accuracy on small, specific datasets. This study adds to the previous literature by extending the evaluation of these models to specialised scenarios, revealing RetinaNet and YOLOv8 as particularly effective in balancing precision and recall in Shape Language contexts.

5.2 Insights from Model Performance

The models' performance study reveals numerous crucial insights into their capabilities and limitations when detecting Shape Language objects in meeting scenarios. RetinaNet consistently outperformed all measures, thanks to its capacity to handle multi-scale features and reduce class imbalance, making it ideal for datasets with varying object sizes and orientations. YOLOv8 achieved very high precision and F1 scores within an optimal confidence range, demonstrating its usefulness for precise and efficient detections; nevertheless, recall decreased at extreme thresholds, revealing sensitivity to specific detection conditions. SSD demonstrated moderate performance, with significant trouble preserving consistency across object classes, most likely because to its simpler architecture and less advanced loss optimisation. RCNN, while useful for detailed detections in other contexts, fell short due to its computational complexity and reliance on region recommendations, which struggled with the dataset's low size and diversity. Overall, these findings highlight the

importance of model architecture, dataset features, and task-specific obstacles in predicting detection performance. RetinaNet and YOLOv8 appear as the most reliable models for this task, with RetinaNet providing balanced performance and YOLOv8 excelling in precision-critical cases.

5.3 Reflection on Methodology and Results

The performance differences of RCNN, RetinaNet, YOLOv8, and SSD can be attributed to their respective architectures, dataset restrictions, scene complexity, and computation constraints. RetinaNet outperformed thanks to its Feature Pyramid Network (FPN) and focus loss, which allowed for robust multi-scale identification and management of class imbalances, resulting in high F1 scores, precision, and recall. YOLOv8's single-stage approach, which was optimised for speed and precision, worked well in most cases but suffered with overlapping objects and high recall thresholds. SSD, while ideal for real-time applications, performed poorly for small or overlapping objects due to its simplified architecture and loss function. RCNN's two-stage technique, while powerful for complex objects, was computationally expensive and unable to generalise on the small dataset, resulting in lower precision and recall.

Most likely, the dataset's limited size and diversity further exacerbated challenges, impacting models like RCNN and SSD more significantly than RetinaNet and YOLOv8. Additionally, computational resource constraints likely hindered optimization, particularly for resource-intensive models such as RCNN and RetinaNet. These parameters explain the observed performance differences among the models.

6 Limitations

While this study provides useful insights into the use of computer vision technologies for identifying, segmenting, and tracking the Shape Language in meetings, certain limitations must be noted in order to motivate the findings and guide future research paths.

6.1 Limited Dataset

The dataset used in this study is limited to a few recordings, which represent a narrow range of meeting situations and object interactions. This restriction may have an impact on the models' generalisability because it does not fully reflect the diversity of object locations, lighting circumstances, and participant behaviours. Furthermore, due to time constraints, only 90 frames were annotated, reducing the variety and scale of the training data. A larger and more diversified dataset is likely to improve model resilience and allow for a more comprehensive evaluation of performance in different scenarios.

6.2 Computational Constraints and Memory Usage

The models were trained and evaluated with restricted GPU power, limiting the experiment's size and complexity. For example, resource constraints prevented hyperparameter adjustment and testing with bigger batch sizes or deeper models. This may have had an impact on the performance of some models, particularly those that require a lot of computing power, such as R-CNN.

Moreover, the preprocessing pipeline, which splits videos into frames and stores them for annotation and model training, requires a substantial amount of RAM. Although the application of optical flow techniques reduced the amount of redundant frames, the memory requirements for storing and processing the dataset remained high. Future studies may investigate more memory-efficient techniques, such as on-the-fly frame generation or leveraging compressed video formats.

7 Responsible Research

7.1 Ethical Considerations

While the study focusses on improving decision-making processes and working together efficiency, it is also required to critically assess the potential ethical consequences of implementing such technologies in real-world settings. There are two main ethical considerations that are relevant to this study:

1. **Privacy:** The use of computer vision in meetings could involve acquiring and analysing sensitive video data containing individuals. Ensuring participants' privacy is critical. To overcome this, the study follows tight data privacy procedures. All datasets were anonymised, thus no personal identifiers were included in the analysis. Any use of these technologies in real-world contexts must follow existing privacy rules, such as GDPR, and include mechanisms for participants' informed consent.
2. **Bias and Fairness:** The models used in this study rely on annotated datasets for training, which may introduce biases depending on the dataset composition. This bias may have an impact on the model's accuracy and fairness when applied to different surroundings or people. To address this, the dataset was carefully chosen to include a wide range of object placements and settings. Future research should investigate the models' generalisability by testing them on different datasets and finding any biases in their predictions.

7.2 Reproductibility

Reproducibility is a key component of responsible research. This study focusses on transparency and reliability by:

1. **Open-source technology:** The preprocessing, annotation, training, and evaluation frameworks, as well as versions of the five object detection models, are openly available. This ensures that other researchers can reproduce and evaluate the results.
2. **Detailed Methodology:** The methodology section contains detailed information on the data preparation, annotation, and model training methods. The use of standardised datasets and uniform evaluation processes means that the findings may be independently validated. Moreover, all hyperparameters, training epochs, and experimental settings are documented to ensure reproducibil-

ity. Any deviations from standard practices are clearly stated.

In conclusion, this study follows responsible research procedures by addressing ethical concerns, assuring transparency, and promoting repeatability. These principles drive the study's contribution to the progress of computer vision applications in the workplace, while also protecting participant rights while promoting trust in the use of new technology.

8 Conclusions and Future Work

8.1 Conclusion

The purpose of this study was to assess the performance of four current computer vision models: YOLOv8, SSD, RCNN, and RetinaNet in recognising and categorising Shape Language objects in meeting scenarios. These geometric shapes serve an important role in promoting collaboration in meetings by acting as boundary objects between abstract and concrete concepts.

The findings show that RetinaNet regularly outperforms competing models on all evaluation criteria, particularly precision, recall, and F1 score. However, YOLOv8 appeared as a strong contender, providing great precision and robust performance while maintaining an optimal confidence level. In contrast, SSD displayed moderate reliability, and RCNN experienced issues due to its computational complexity and inefficiency on the restricted dataset. These findings highlight the need of balancing model architecture and dataset features when using computer vision models in specialised situations such as Shape Language recognition.

Despite these achievements, some limitations were discovered in the study. The short dataset limited the generalisability of the conclusions, while computing restrictions limited the opportunity for hyperparameter adjustment and deeper evaluations. These issues require more research, in order to provide accurate results.

8.2 Future Work

This research opens several areas for future investigation. Some of these come from finding solutions to the limitations described in Section 7. Other future improvements can include the following:

1. **Addressing occlusions and variable object sizes:** Real-world situations frequently include items of different sizes because of variations in position, orientation, or distance from the camera, as well as occlusions, in which objects partially or totally block one another. Accurate detection and classification are showing limitations in these circumstances. To improve the models' capacity to manage these complexities, future research can investigate different strategies like transformer-based models such as DETR, in order to deal with these cases better. Furthermore, robustness and generalisation can be enhanced by using artificial datasets with improved occlusion scenarios and gathering data with a wider variety of object sizes.
2. **Semi-supervised and active learning:** To address the limitations of manual annotation, future work could explore semi-supervised or active learning frameworks to reduce annotation effort while maintaining high-quality training datasets.

In conclusion, this study provides a framework for using computer vision techniques to improve workplace collaboration by automating the analysis of interactions with the Shape Language. Researchers can improve these tools and broaden their application to various and complicated real-world circumstances by resolving the identified limitations and implementing the outlined recommendations in the future.

References

[1] Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Image Analysis: 13th Scandinavian Conference, SCIA 2003 Halmstad, Sweden, June 29–July 2, 2003 Proceedings 13*, pages 363–370. Springer, 2003.

[2] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[3] Alexander Kirillov et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

[4] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 21–37. Springer, 2016.

[5] N. Ravi, V. Gabeur, Y. T. Hu, R. Hu, C. Ryali, T. Ma, and C. Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

[6] J Redmon. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[7] T-YLPG Ross and GKHP Dollár. Focal loss for dense object detection. In *proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2980–2988, 2017.

[8] Vormtaal. Vormtaal website. https://www.vormtaal.com/, 2025. Accessed: 2025-01-23.

[9] Zhong-Qiu Zhao, Peng Zheng, Shou-Tao Xu, and Xindong Wu. Object detection with deep learning: A review. *IEEE Transactions on Neural Networks and Learning Systems*, 30(11):3212–3232, 2019.

[10] Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, and Jieping Ye. Object detection in 20 years: A survey. *Proceedings of the IEEE*, 111(3):257–276, 2023.