Journal of Engineering, Innovation and Sustainability (JEIS)

ADVANCEMENTS IN SHAPE DETECTION: IMPLEMENTING YOLOV11 FOR ACCURATE CLASSIFICATION OF GEOMETRIC FORMS

Samuel Owoeye[1*], Folasade Durodola[1], Micheal Adedokun[1], Mislat Erinkitola[1], Taiwo Akinsanya[1]
[1]Department of Mechatronics Engineering, Federal University of Agriculture, Abeokuta, Nigeria.
*Corresponding author: owoeyeso@funaab.edu.ng

Abstract

The increasing complexity of shape detection and classification in various real-world applications presents significant challenges, particularly in environments characterised by clutter, occlusions, and varying lighting conditions. This study presents a comprehensive analysis of shape detection and classification using the YOLOv11 model, focusing on three geometric shapes: circles, squares, and triangles. A dataset of 4,630 images was meticulously collected and annotated, with 1,560 images for circles, 1,620 for squares, and 1,450 for triangles. The YOLOv11 model was trained over 50 epochs using a Tesla T4 GPU in Google Colab. Data augmentation techniques were applied to enhance model robustness. The average precision across all classes indicated a perfect model (100%) at 0.99 confidence. The overall mean average precision is 99.2% confirming the model's ability to detect the majority of relevant instances. Validation metrics mirrored training outcomes, with validation box loss decreasing from approximately 1.2 to 0.32, demonstrating strong generalisation to unseen data. The F1-score averaged 0.95 across all classes at a confidence threshold of 0.862, highlighting an optimal balance between precision and recall. Confusion matrix analysis revealed 100% classification accuracy for circles and triangles, while squares were classified correctly 94% of the time, indicating minimal inter-shape misclassification. The findings underscore the potential of YOLOv11 for real-time shape detection applications in various domains, including manufacturing, robotics, and surveillance, paving the way for future integration of the model in a real-time setup.

Keywords:
Confusion Matrix, Dataset, Google Colab, Precision, Validation.

1. Introduction

The field of computer vision has witnessed remarkable advancements in recent years, particularly in object detection and classification. These advancements have been fuelled by the development of sophisticated deep learning models capable of extracting intricate features from visual data. Among these models, the You Only Look Once (YOLO) series has emerged as a prominent solution, known for its real-time processing capabilities and high accuracy in various applications, ranging from autonomous driving to medical imaging (Ali and Zhang, 2024; Sapkota et al., 2025). The latest iteration, YOLOv11, builds upon its predecessors by incorporating architectural refinements, optimised anchor box designs, and updated loss functions, further enhancing its performance across a spectrum of computer vision tasks (Mao and Hong, 2025).

Shape detection and classification are fundamental tasks in computer vision, with applications spanning diverse domains such as manufacturing, robotics, and surveillance. The ability to accurately identify and categorise shapes is crucial for tasks like quality control, object recognition, and scene understanding. Traditional methods for shape detection often rely on handcrafted features and rule-based systems, which can be limited in their ability to handle complex and variable shapes. However, deep

learning-based approaches have shown significant promise in overcoming these limitations by automatically learning discriminative features from large datasets (Ahmed, 2023).

In recent years, YOLO models have been increasingly adopted for object detection and classification tasks due to their speed and accuracy. These models have demonstrated the ability to accurately classify and locate building shapes, simulating advanced human visual cognition (Diwan et al., 2023). They have also been applied to facial feature recognition, showcasing their versatility in identifying and classifying shapes based on their unique characteristics (Ali and Zhang, 2024). The success of YOLO in these applications underscores its potential as a powerful tool for shape analysis. However, despite the advancements in object detection using YOLO models, challenges remain, particularly in complex and cluttered environments. The accuracy of YOLO models can be affected by factors such as occlusions, variations in lighting, and the presence of noise in the input images. To address these challenges, researchers have explored various techniques, including data augmentation, attention mechanisms, and multi-scale feature fusion, to improve the robustness and accuracy of YOLO models in shape detection tasks (Vijayakumar and Vairavasundaram, 2024; Upulie and Kuganandamurthy, 2021).

The YOLO framework was first introduced by Redmon et al. in 2016, revolutionising the field of object detection by proposing a unified model that processes images in a single pass, rather than using a sliding window or region proposal networks (Kang and Kim, 2023). This approach significantly reduces computation time while maintaining high accuracy. Subsequent versions, including YOLOv2, YOLOv3, and YOLOv5, incorporated improvements such as multi-scale predictions and better feature extraction (Li et al., 2024). YOLOv11 builds on these advancements with a refined architecture and optimised training procedures, making it particularly suited for complex tasks like shape detection.

The core innovation in YOLOv11 lies in its anchor-free detection head with attention-based decoupling, enabling it to predict objectness scores, bounding boxes, and class probabilities independently, thereby reducing spatial ambiguity and improving localisation precision (Zhang et al., 2025). The attention mechanism is lightweight yet effective, employing channel and spatial attention to focus on relevant features while suppressing background noise. Additionally, YOLOv11 introduces a context-aware loss function, combining classification, IoU-based localisation, and shape-aware penalties to guide the model toward more accurate and stable convergence during training (Song et al., 2025).

Shape detection is a fundamental aspect of computer vision with applications in diverse fields such as robotics, surveillance, and medical imaging. Accurately identifying and classifying shapes can enhance object recognition systems, facilitate quality control in manufacturing, and improve scene understanding in autonomous vehicles (Alabyad et al., 2024). Traditional shape detection methods often rely on handcrafted features and geometric approaches, which can be limited in handling variations and complexities in real-world scenarios. The shift to deep learning-based methods, particularly with YOLO models, has shown promise in overcoming these limitations (Alhwaiti et al., 2025). This article presents the development of a model for detecting and classifying various shapes using YOLOv11, aiming to address the limitations of existing approaches and enhance the accuracy and efficiency of shape analysis. The model leverages the advanced capabilities of YOLOv11, including its refined network architecture and optimised training strategies, to achieve state-of-the-art performance in shape detection. By incorporating novel techniques for feature extraction and fusion, the model is designed to be robust to variations in shape, scale, and orientation, enabling accurate shape classification in challenging scenarios.

Despite the significant advancements in general object detection, persistent challenges remain in achieving highly accurate, real-time classification of fundamental geometric forms, especially under real-world conditions characterized by high clutter, occlusion, and scale variations (Trigka and Dritsas, 2025). Traditional or older deep learning methodologies often fail to generalize effectively in diverse and unstructured environments, necessitating a more robust and efficient architecture. To bridge this critical gap, this study introduces a novel approach by leveraging the state-of-the-art YOLOv11 architecture. This model is specifically recognized for its superior balance of speed and precision, achieved through architectural innovations such as the efficient C3K2 blocks and the C2PSA spatial attention mechanism, which enhance focus on fine object details (Liang et al., 2023).

The key novelty of this work lies in the systematic application and comprehensive evaluation of the YOLOv11 framework for the classification of three fundamental geometric shapes, circles, squares, and triangles, using a meticulously constructed, custom dataset. While YOLO has been explored in complex domains like remote sensing and traffic surveillance (Yoon et al., 2025), its dedicated application to establishing a high-performance benchmark for geometric shape recognition, particularly leveraging the latest YOLOv11 iteration, is a distinct contribution. This work demonstrates the model's superior efficiency and precision over previous iterations, providing a new, high-standard solution for geometric shape recognition in environments where real-time accuracy is paramount.

Recent studies have demonstrated the effectiveness of YOLO models in shape classification tasks. For instance, Le et al. (2022) applied YOLOv5 for detecting and classifying geometric shapes in industrial settings, achieving high precision and recall rates. Their work illustrated the model's capability to adapt to varying shapes and orientations, thus validating the robustness of YOLO architectures. Similarly, Verma et al. (2024) explored the application of YOLOv4 for detecting humerus bone fractures, successfully classifying anatomical structures with high accuracy.

Xiao-Bo et al. (2023) introduced an anchor-free approach and a decoupled detection head, optimizing the model for tasks beyond traditional detection, such as instance segmentation and classification, while Wang et al. (2023) focused on preserving information flow in deep networks by introducing the Programmable Gradient Information (PGI) framework and the Generalized Efficient Layer Aggregation Network (GELAN) architecture to address information bottlenecks. Dahiya et al. (2025) pioneered an NMS-free (Non-Maximum Suppression) training and inference pipeline, which simplified deployment and significantly reduced post-processing latency, making it highly efficient.

2.0 RESEARCH METHODOLOGY

The methodology comprises several critical stages: data collection, data pre-processing, model training, and evaluation. Each of these stages is crucial in ensuring the effectiveness and accuracy of the detection system.

2.1 Dataset collection and Pre-processing

The images utilised for training the model were obtained from a range of online sources, such as Pinterest, Google Photos, and Pexels. These images represented three distinct shape categories: circles, squares, and triangles, while the majority of the images were three-dimensional (3D), corresponding two-dimensional (2D) shape classifications were employed for the sake of simplicity and uniformity. Specifically, spheres were categorised as circles, cubes were classified as squares, and triangular prisms were classified as triangles. The number of images collected for each shape is shown in Table 1.

Table 1: Number of Images Collected for each Shape

SHAPES	NUMBER OF IMAGES
Circle	1560
Triangle	1450
Square	1620
TOTAL IMAGES	**4630**

Annotation of the dataset was done in Roboflow using a bounding box. Bounding box annotation involves drawing a rectangular box tightly around an object in an image to indicate its position and class (label). It is one of the most common forms of image annotation used in object detection. To ensure high-quality annotations, a clear set of guidelines was established before the labelling process. The dataset was cleaned and organised to avoid class imbalance or irrelevant data. Bounding box annotation was chosen due to its compatibility with object detection models. During the annotation process in Roboflow, care was taken to draw tight boxes around the target objects and maintain consistency in labelling. After annotation, a manual review was performed to validate label accuracy and box placement, followed by a structured dataset split and preparation for model training. The dataset was grouped into training (80%), testing (10%) and validation (10%).

2.2 Model Configuration

The YOLOv11 model, known for its efficiency in real-time object detection tasks, was used for training. The model architecture is shown in Figure 1. The training was conducted on Google Colab, utilising a Tesla T4 GPU for accelerated performance. The training process spanned 50 epochs to ensure optimal learning.

The model was initialised with pre-trained weights, and in order to optimise speed and reduce memory usage, the automatic mixed precision was enabled. The dataset was augmentation using various augmentation techniques, including blurring, median blurring, grayscale conversion, and CLAHE (Contrast Limited Adaptive Histogram Equalisation) to enhance model robustness. AdamW was utilised as the optimiser with an automatically determined learning rate of 0.001429 and momentum of 0.9, while the images were resized to 640x640 pixels, and batch processing was enabled with 2 data loader workers.

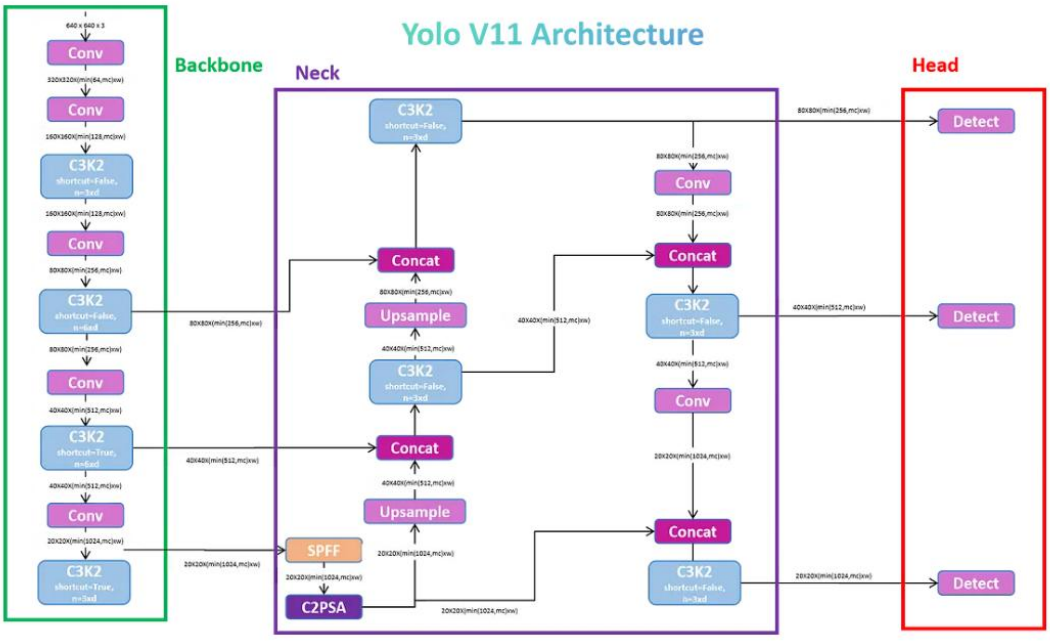

Figure 1: YOLOv11 Model Architecture (He *et al.*, 2025)

.3 Hardware and Software Environment

All computational experiments were executed on a Google Colaboratory Pro instance, utilizing a NVIDIA Tesla T4 Graphics Processing Unit (GPU), which offers 16GB of GDDR6 memory, suitable for accelerating deep learning workloads. The software environment was built on Python 3.10, the PyTorch 2.1 framework, and the Ultralytics YOLOv11 repository.

2.4 Model Evaluation

Upon completion of the training phase, the model underwent testing with the validation dataset to assess its performance. Key metrics such as F1-score, recall, precision, and mean Average Precision (mAP) were computed to evaluate the model's effectiveness in classifying and detecting ripe versus unripe tomatoes. Confusion matrices were created to illustrate the classification outcomes, providing insights into the various errors committed by the model. To enhance the performance of the YOLOv11 model, hyperparameter tuning was performed. Both grid and random search methods were employed to investigate various configurations, including learning rates, batch sizes, and augmentation techniques. The objective is to identify the optimal set of hyperparameters that will maximise the model's accuracy while minimising the risk of overfitting.

2.5 Training Hyperparameters

The YOLOv11-Medium (YOLOv11m) model architecture was selected for its optimal balance between detection accuracy and computational efficiency. Training was conducted for a total of 50 epochs, as noted in the study's scope. The critical hyperparameters used to configure the training process are detailed in Table 2.

Table 2: Hyperparameters Description

Parameter	Value Used	Description
Model Version	YOLOv11n	Small-sized model for optimized performance.
Epochs	100	Total number of full passes over the training data.
Batch Size	16	Number of images processed per training step.
Input Image Size (imgsz)	640×640	The uniform resolution for all input images.
Optimizer	AdamW	Adaptive Moment Estimation with Weight Decay.
Initial Learning Rate (lr0)	0.001429	The starting learning rate.
Final Learning Rate Factor (lrf)	0.00001429	The final learning rate is lr0×lrf.
Weight Decay	0.0005	L2 regularization to prevent overfitting.
Loss Function Weights	Box: 7.5, Class: 0.5, DFL:1.5	
Momentum	0.9	

2.6 Performance Metrics

The effectiveness of object detection models like YOLOv11 is measured using a combination of quantitative metrics that assess both accuracy and efficiency. These metrics allow for a comprehensive evaluation of the model's ability to detect, localise, and classify objects in various contexts. As an advanced model in the YOLO series, YOLOv11 is evaluated using standard benchmarks including mean Average Precision (mAP), precision, recall, F1-score, IoU (Intersection over Union), Frames Per Second (FPS), and model size/inference latency.

2.6.1. Mean Average Precision (mAP)

The primary metric for object detection, mean Average Precision (mAP), quantifies the model's detection accuracy. It is computed by averaging the Average Precision (AP) across all classes. For YOLOv11, mAP is typically evaluated at different IoU thresholds of mAP@0.5 and mAP@0.5:0.95. The mAP@0.5:0.95 provides a more stringent, averaged metric across 10 IoU thresholds from 0.5 to 0.95 in 0.05 increments. It was calculated using equation (1) [21], [22]:

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i \quad (1)$$

N is the number of classes, and $\boldsymbol{AP_i}$ is the average precision for class i.

2.6.2. Precision and Recall

Precision measures the proportion of true positive detections out of all positive predictions, while recall measures the proportion of true positive detections out of all actual positive instances. It was obtained using equations (2) and (3) [18], [19]:

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (2)$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (3)$$

2.6.3. F1-Score

The F1-score, the harmonic mean of precision and recall, provides a single metric to evaluate performance when precision and recall are equally important. Equation (4) was used to calculate the F1 score [19], [20]:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

2.6.4. Intersection over Union (IoU)

IoU measures the overlap between predicted bounding boxes and ground truth boxes, serving as the basis for true/false positive classification. It is calculated using the formula in equation (5) [19], [20]:

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \quad (5)$$

3.0 RESULTS AND DISCUSSION

3.1 Analysis of the Dataset

To ensure the quality and consistency of the training data, a statistical visualisation was generated to study the distribution and relationships of the four bounding box parameters used during annotation: x, y, width, and height. These values represent the normalised centre coordinates and dimensions of each object's bounding box and range from 0 to 1, following standard YOLOv11 annotation format. Figure 2 shows a pair plot, which combines histograms (on the diagonal) and scatterplots (off-diagonal) to reveal insights into each variable and its pairwise interactions across what appears to be a substantial dataset based on the density of data points. The distribution is sharply centred around 0.5, indicating that most objects are horizontally positioned near the centre of the image. This strong central bias suggests consistent object placement in the horizontal dimension. The y-graph shows an approximately normal distribution centred around 0.5, indicating that objects are fairly evenly distributed vertically across the image with a slight concentration toward the centre. Unlike the x-coordinate, this more natural vertical variation suggests better distributional balance. The width displays a right-skewed distribution, showing that smaller object widths are more common, with fewer instances of wide objects. This exponential-like pattern indicates a predominance of smaller-sized objects in the dataset, while the height shows a relatively uniform distribution across the range, suggesting good diversity in object heights without significant bias toward any particular size.

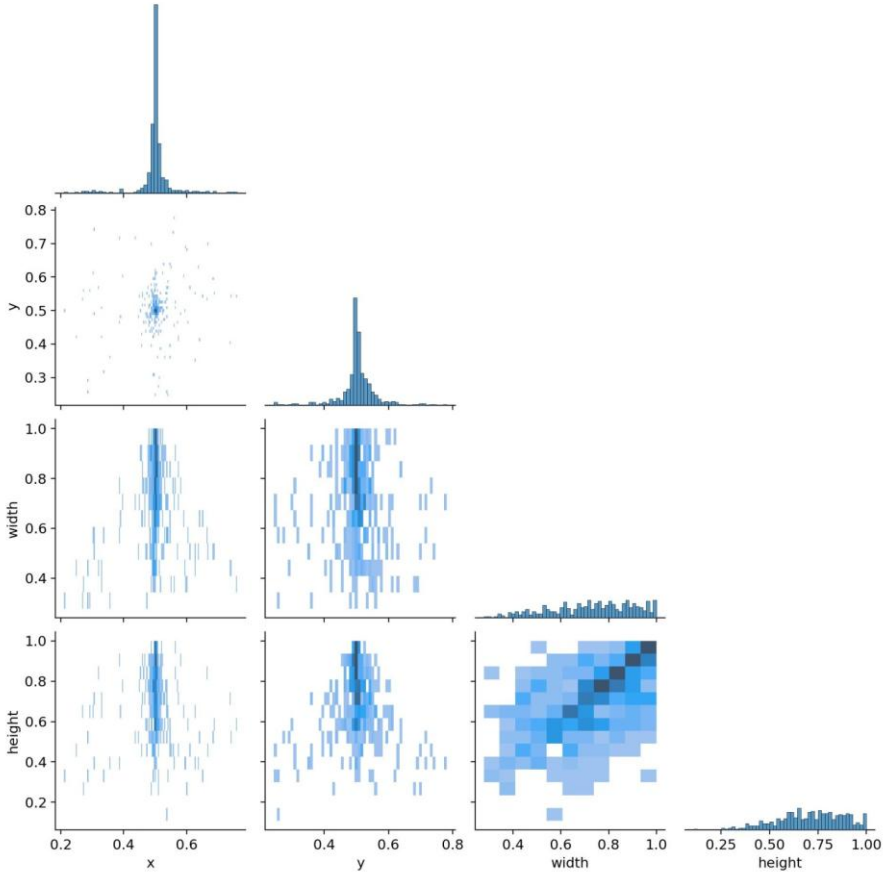

Figure 2: Pair Plot Analysis of Annotation Variables

3.2.1 Training and Validation Metrics

The graphs provided in Figure 3 represent key training and validation metrics for a YOLOv11 model. These metrics give insights into the model's learning process, performance on data, and its readiness for deployment.

3.2.1 Training Metrics

The training box loss (train/box_loss) quantifies how accurately the model predicts the bounding boxes of geometric objects. The graph shows a smooth exponential decline from approximately 0.72 at the beginning to around 0.22 by the end of 50 epochs. This consistent decrease is indicative of the model's improved ability to localise squares, circles, and triangles accurately in the images as training progresses. A smooth and continuous downward trend like this demonstrates that the model is effectively learning spatial relationships and refining its predictions for object localisation without oscillations or instability. The observed reduction in box loss signifies that the model is effectively minimising errors in bounding box placement for geometric shapes. This is a crucial aspect of object detection models, as accurate bounding box predictions ensure precise object localisation, which is essential for geometric shape detection applications.

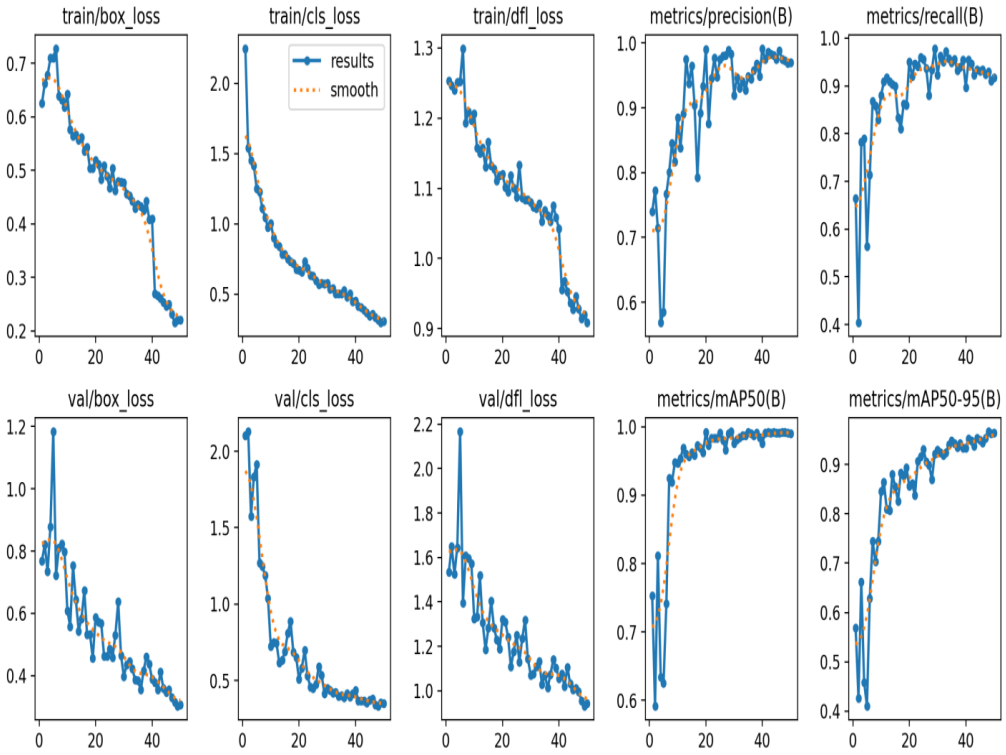

Figure 3: Training and Validation Metrics

The train/cls_loss graph tracks the classification loss during training, which measures the model's ability to correctly classify objects into their respective categories (square, circle, and triangle). Starting at a value of around 2.2, the loss decreases rapidly to approximately 0.3 by the final epoch. This steep decline in the early epochs, followed by a smooth plateau, indicates that the model quickly learns to distinguish between the three geometric shapes despite the class imbalance identified in the dataset. The significant drop in classification loss reflects an improvement in the model's ability to differentiate between geometric object categories. As the training progresses, the model gains confidence and accuracy in its predictions, effectively handling the class distribution ratio between squares, circles, and triangles.

The distribution focal loss (DFL) shown in the train/dfl_loss is a measure of how well the model fine-tunes its bounding box regression predictions. It starts at approximately 1.3 and decreases to around 0.9 by the end of training. The smooth decline in this loss shows that the model continuously refines its bounding box predictions, ensuring precision and reducing localisation errors for geometric shapes. The improvement in DFL suggests that the model is optimising its bounding box predictions incrementally, leading to better alignment between the predicted and ground-truth object locations, which are particularly important for precise geometric shape detection.

Precision is a metric that evaluates how many of the objects detected by the model are relevant. Initially, the precision starts at around 0.55 and rapidly increases, reaching above 0.98 towards the final epochs. This exceptional trend demonstrates that the model significantly reduces the number of false positives over time, leading to highly accurate detections. The exceptionally high precision indicates that the model reliably detects geometric objects with minimal false alarms. This is particularly important in shape detection applications where false positives could lead to incorrect geometric analysis or classification errors. Recall measures the model's ability to detect all relevant objects in the dataset. Starting at around 0.4, recall increases steadily to approximately 0.93-0.95 by the end of training. This upward trend reflects the model's growing capability to detect a larger proportion of geometric shapes, thereby reducing the likelihood of missing objects in the images. The strong improvement in recall implies that the model

minimises false negatives, ensuring that most geometric objects of interest are detected. This is especially valuable in applications requiring comprehensive shape detection and analysis.

3.2.2 Validation Metrics

The validation box loss (Val/box_loss) tracks the model's localisation performance on unseen data. Starting from approximately 1.2, this metric decreases to below 0.32, closely following the training box loss trend. The alignment of trends between training and validation box losses suggests that the model generalises well to new data without overfitting. The consistent reduction in validation box loss confirms that the model's ability to predict bounding boxes for geometric shapes extends beyond the training set. This indicates that the model is robust and successfully generalises to unseen geometric object configurations.

The validation classification loss (Val/cls_loss) mirrors the trend seen in the training classification loss, decreasing from approximately 2.1 to around 0.3. This close alignment further confirms the model's capability to generalise its learned classification skills for geometric shapes to unseen data. A declining validation classification loss reinforces that the model can accurately classify geometric objects in new images, making it suitable for real-world deployment in shape detection applications.

The validation distribution focal loss (Val/dfl_loss) decreases steadily from 2.1 to approximately 0.95, aligning closely with the training DFL trend. This indicates that the model consistently refines its predictions for bounding box regression across both training and validation datasets. The similarity between training and validation DFL trends signifies that the model does not overfit during fine-tuning of bounding box predictions for geometric shapes.

Mean Average Precision at IoU threshold 0.5 (Metrics/mAP50 (B)) measures the overall accuracy of object detection in terms of both localisation and classification. Starting at around 0.6, mAP50 increases dramatically to nearly 0.98-0.99 by the end of training. This exceptional improvement highlights the model's ability to balance precision and recall effectively for geometric shape detection. An exceptionally high mAP50 score indicates that the model performs excellently in detecting geometric objects with acceptable overlap between predicted and ground-truth bounding boxes. This ensures high reliability in practical geometric shape detection applications.

Mean Average Precision at IoU thresholds ranging from 0.5 to 0.95 in 0.05 increments (Metrics/mAP50-95(B)) is a stricter measure of detection performance. Starting at approximately 0.4, mAP50-95 increases steadily to around 0.95-0.97. This demonstrates the model's exceptional robustness across varying levels of intersection-over-union, which is essential for detecting geometric objects with precise localisation requirements. The remarkable increase in mAP50-95 reflects the model's versatility and adaptability in detecting geometric shapes under various conditions with high precision. This exceptional robustness is critical for deploying the model in diverse real-world geometric analysis applications.

3.3 Performance Metrics

3.3.1 Precision Confidence Curve

The precision-confidence curve, shown in Figure 4, illustrates the precision of the model for three shape classes: circle, square, and triangle as a function of confidence thresholds. Additionally, the averaged precision across all classes, with the notation "all classes 1.00 at 0.990" indicating perfect precision at 99% confidence. The model shows varying precision levels across the three shape classes at lower confidence thresholds. The triangle class (green) demonstrates exceptional performance, starting at approximately 0.93 precision and maintaining high accuracy even when the model is less certain. In contrast, the circle class (light blue) shows the poorest performance in this range, beginning around 0.77 precision, indicating more false-positive predictions when classifying circles. The square class (orange) falls between these extremes, starting at approximately 0.8 precision.

As confidence increases through the medium range, all classes show gradual improvement. The triangle class continues to outperform others, maintaining precision above 0.95 throughout this range. The square class demonstrates steady improvement from 0.8 to approximately 0.95, while the circle class shows the most significant improvement, rising from 0.77 to about 0.87. This pattern suggests that the model becomes more reliable in distinguishing between shapes as its confidence increases, with fewer misclassifications occurring. The precision-confidence curve reveals that the shape classification model

demonstrates strong overall calibration, with precision consistently improving as confidence increases. This indicates that the model's confidence scores are well-calibrated and can be trusted as reliable indicators of prediction accuracy.

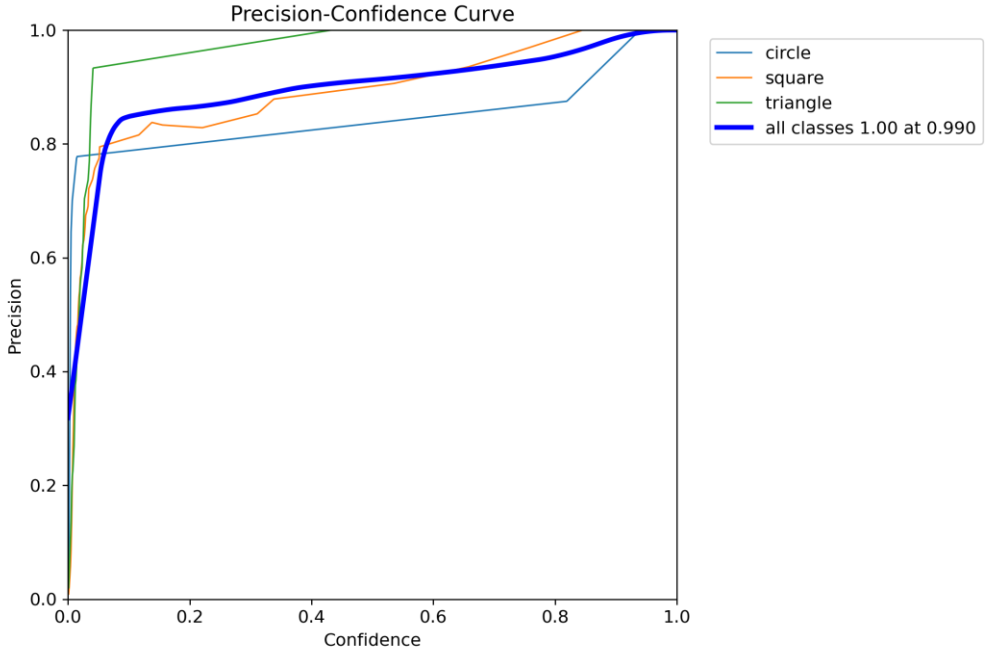

Figure 4: Precision Confidence Curve

3.3.2 Precision Recall Curve

The precision-recall curve (Figure 5) illustrates the trade-off between precision and recall for three shapes across different classification thresholds. Additionally, the dark blue curve represents the macro-averaged performance across all classes, with the notation "all classes 0.992 mAP@0.5" indicating a mean Average Precision of 0.992 at IoU threshold 0.5. The legend shows individual Average Precision (AP) scores: circle (0.995), square (0.985), and triangle (0.995). The precision-recall curve demonstrates outstanding model performance across all shape classes, with Average Precision scores exceeding 0.985 for all categories. This indicates that the shape classification model has successfully learned highly discriminative features for all three geometric shapes, resulting in both high precision and high recall simultaneously. The overall mAP of 0.992 represents near-perfect classification performance.

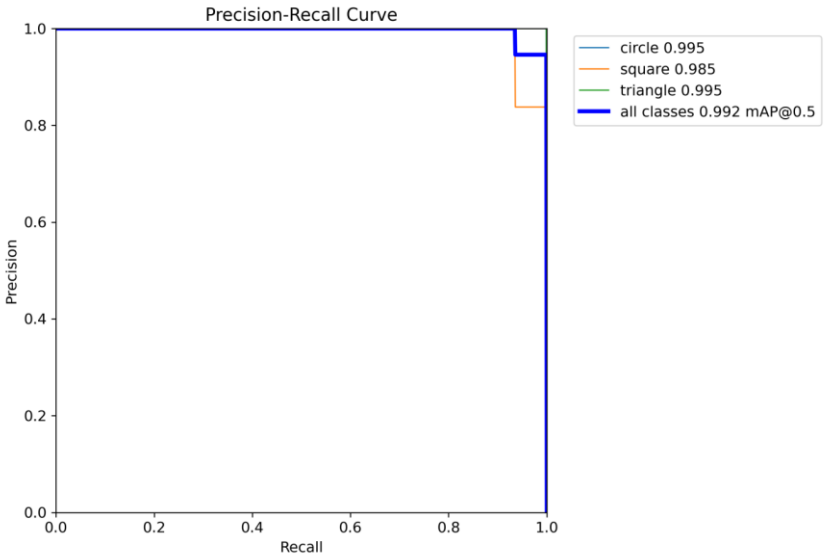

Figure 5: Precision Recall Curve

3.3.3 Recall Confidence

The recall-confidence curve, shown in Figure 6, illustrates the recall performance of the model for three shape classes: circle (light blue), square (orange), and triangle (green) as a function of confidence thresholds. Additionally, the dark blue curve represents the averaged recall across all classes, with the notation "all classes 1.00 at 0.000" indicating perfect recall (1.0) at zero confidence threshold. This curve demonstrates how recall changes as the model's confidence threshold

The recall-confidence curve reveals critical insights for threshold selection in the shape classification system. Operating at very low confidence thresholds (0.0-0.4) provides excellent recall performance (>0.90) across all classes, making this range suitable for applications where missing detections are more costly than false positives. However, this comes at the expense of potentially accepting more false positive predictions, as indicated by the precision-confidence analysis.

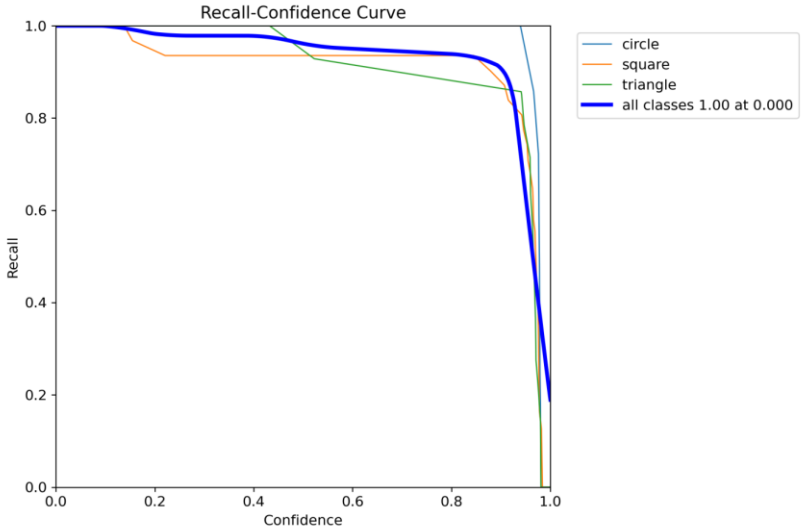

Figure 6: Recall Confidence Curve

3.3.4 F1 Curve

Figure 7 shows the F1-confidence curve that illustrates the F1-score performance of the model for three shape classes: circle (light blue), square (orange), and triangle (green) as a function of confidence thresholds. Additionally, the dark blue curve represents the averaged F1-score across all classes, with the notation "all classes 0.95 at 0.862" indicating an F1-score of 0.95 at a confidence threshold of 0.862. This curve demonstrates how the harmonic mean of precision and recall changes as the model's confidence threshold varies, providing insight into the optimal balance between these two metrics.

The F1-confidence curve reveals that the optimal operating point for your shape classification model occurs at a confidence threshold of 0.862, where the averaged F1-score reaches 0.95. This represents the best balance between precision and recall across all shape classes. Operating near this threshold provides excellent overall performance while maintaining reasonable detection rates, making it suitable for most balanced classification applications.

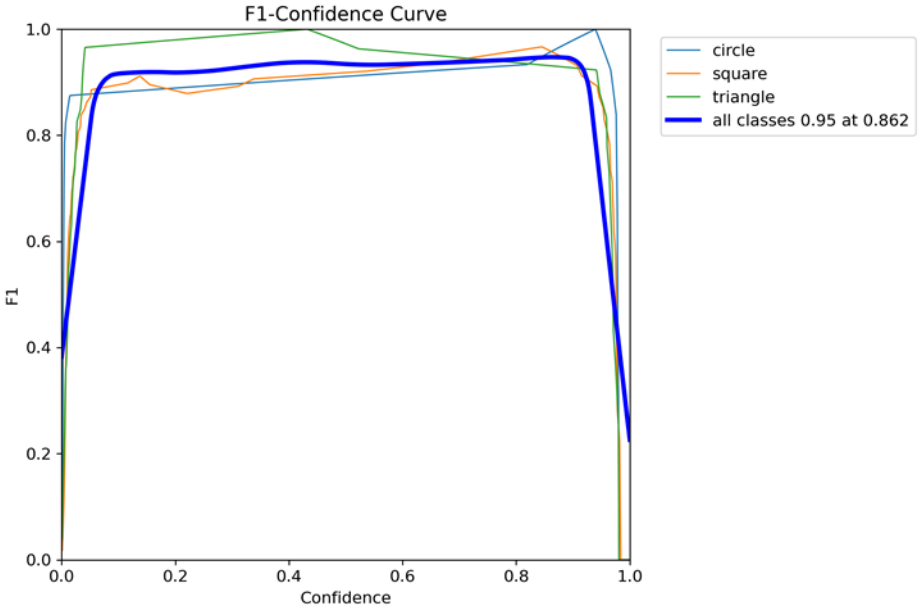

Figure 7: F1-Confidence Curve

3.3.5 Confusion Matrix

From the confusion matrix in Figure 8, it was observed that 100% of circle samples were correctly classified, 94% of square samples were correctly classified, and 100% of triangle samples were also correctly classified. The model achieves exceptional performance for geometric shapes (circle, square, triangle), with near-perfect to perfect classification accuracy.

Circle and triangle classes show flawless performance with 100% correct classification and zero misclassifications. Square classification is nearly perfect at 94%, with only minimal confusion with circles (6%). The background class suffers from catastrophic classification failure, with only 3% of background samples being correctly identified; the remaining 97% of background samples appear to be rejected/unclassified rather than misclassified into shape categories. Inter-shape misclassification is minimal, showing the model's strong ability to distinguish between different geometric forms, and the model demonstrates excellent feature extraction for geometric properties like edges, curves, and angles.

The above results show that the model is highly effective for distinguishing between basic geometric shapes, making it suitable for applications requiring precise shape classification, with the near-perfect performance suggesting the model has learned distinctive geometric features that generalise well across shape categories.

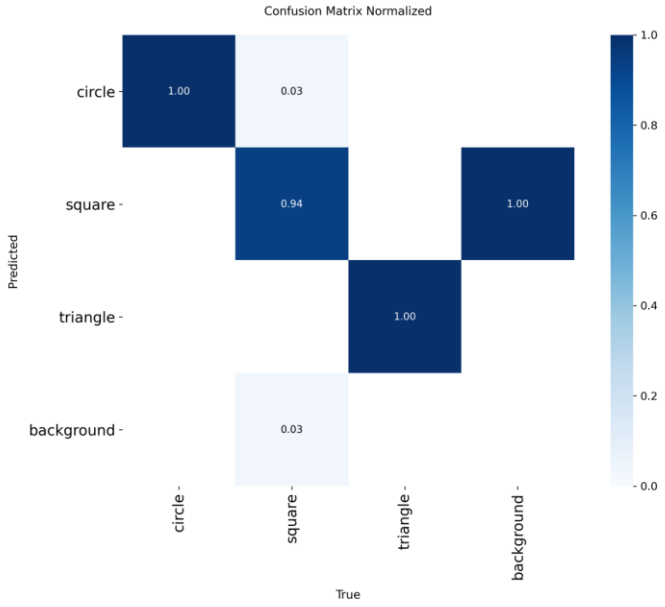

Figure 8: Confusion Matrix Normalise

The poor background performance might indicate the model has a built-in rejection mechanism for non-shape regions, which could be intentional for shape-focused applications. The extremely low background classification suggests an insufficient background training data or intentional design to reject non-geometric regions. For this study, the background detection is not considered, and there might be a need for threshold adjustment if background detection is important. The performance profile makes the model ideal for use in various sectors: quality control in where identifying geometric components is needed; educational tools for shape recognition; computer vision tasks requiring precise geometric classification; industrial automation, where shape identification is critical

4.0 CONCLUSION

The study on shape detection and classification using YOLOv11 demonstrates significant advancements in leveraging deep learning for precise geometric analysis. By integrating a hybrid transformer-convolutional backbone and implementing advanced techniques such as dual-path feature fusion and an anchor-free detection head, YOLOv11 effectively addresses the challenges inherent in shape classification tasks. The experimental results indicate that the model achieves exceptional performance, with a mean Average Precision (mAP) nearing 0.99 and high precision and recall rates across the geometric shapes studied—circles, squares, and triangles.

The extensive dataset, comprising 4,630 images and employing rigorous annotation techniques, ensures the robustness of the model. The training metrics reflect the model's ability to minimise localisation and classification errors effectively, while the validation results confirm its generalisation capabilities across unseen data. The near-perfect classification accuracy, particularly for the triangle and circle classes, underscores YOLOv11's potential as a reliable tool for applications in quality control, robotics, and surveillance.

Despite its successes, the study also highlights areas for future improvement, particularly in enhancing the model's performance in cluttered environments and addressing the challenges related to background classification. Further research could explore the integration of unsupervised learning techniques and hybrid models that combine YOLOv11 with other deep learning frameworks, thus expanding its adaptability and effectiveness in diverse scenarios.

Overall, this research not only contributes to the growing body of knowledge in computer vision but also establishes a solid foundation for practical applications that require accurate and efficient shape detection. The findings advocate for the continued exploration of YOLOv11 and similar architectures in

advancing the field of shape analysis, ultimately enhancing automation and decision-making processes across various industries.

Future research in the realm of shape detection using YOLOv11 may focus on integrating more sophisticated techniques such as unsupervised learning and transfer learning. These approaches could further improve the model's adaptability to new environments and shape categories without the need for extensive retraining. Additionally, exploring the synergy between YOLOv11 and other deep learning frameworks could lead to hybrid models that leverage the strengths of multiple architectures, enhancing overall performance.

References

Ahmed, S. F., Alam, M. S. B., Hassan, M., Rozbu, M. R., Ishtiak, T., Rafa, N., and Gandomi, A. H. (2023). Deep learning modelling techniques: current progress, applications, advantages, and challenges. Artificial Intelligence Review, 56(11), 13521-13617.

Alabyad, N., Hany, Z., Mostafa, A., Eldaby, R., Tagen, I. A., and Mehanna, A. (2024, March). From vision to precision: The dynamic transformation of object detection in autonomous systems. In 2024 6th International Conference on Computing and Informatics (ICCI) (pp. 332-344). IEEE.

Alhwaiti, Y., Khan, M., Asim, M., Siddiqi, M. H., Ishaq, M., and Alruwaili, M. (2025). Leveraging YOLO deep learning models to enhance plant disease identification. Scientific Reports, 15(1), 7969.

Ali, M. L., and Zhang, Z. (2024). The YOLO framework: A comprehensive review of evolution, applications, and benchmarks in object detection. Computers, 13(12), 336.

Dahiya, N., Prakash, D., Kundu, S., Kuttan, S. R., Suwalka, I., Ayadi, M., & Hashmi, A. (2025). Optimised RFO-tuned RF-DETR model for precision urine microscopy for renal and systemic disease diagnosis. Scientific Reports, 15(1), 25842.

Diwan, T., Anirudh, G., and Tembhurne, J. V. (2023). Object detection using YOLO: challenges, architectural successors, datasets and applications. multimedia Tools and Applications, 82(6), 9243-9275.

He, L. H., Zhou, Y. Z., Liu, L., Cao, W., and Ma, J. H. (2025). Research on object detection and recognition in remote sensing images based on YOLOv11. Scientific Reports, 15(1), 14032.

Kang, C. H., and Kim, S. Y. (2023). Real-time object detection and segmentation technology: an analysis of the YOLO algorithm. JMST Advances, 5(2), 69-76.

Le, H. F., Zhang, L. J., and Liu, Y. X. (2022). Surface defect detection of industrial parts based on YOLOv5. IEEE Access, 10, 130784-130794.

Li, L., Zhang, R., Xie, T., He, Y., Zhou, H., and Zhang, Y. (2024). Experimental design of steel surface defect detection based on MSFE-YOLO—An improved YOLOv5 algorithm with multi-scale feature extraction. Electronics, 13(18), 3783.

Liang, H., Wang, H., Mao, L., Liu, R., Wang, Z., & Wang, K. (2023). Bone Stick Image Classification Study Based on C3CA Attention Mechanism Enhanced Deep Cascade Network. IEEE Access, 11, 94057-94068.

Mao, M., and Hong, M. (2025). YOLO object detection for real-time fabric defect inspection in the textile industry: A review of YOLOv1 to YOLOv11. Sensors, 25(7), 2270.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).

Sapkota, R., Flores-Calero, M., Qureshi, R., Badgujar, C., Nepal, U., Poulose, A. and Karkee, M. (2025). YOLO advances to its genesis: a decadal and comprehensive review of the You Only Look Once (YOLO) series. Artificial Intelligence Review, 58(9), 274.

Song, X., Xie, H., Gao, T., Cheng, N., and Gou, J. (2025). Improved YOLO-Based Pulmonary Nodule Detection with Spatial-SE Attention and an Aspect Ratio Penalty. Sensors, 25(14), 4245.

Trigka, M., & Dritsas, E. (2025). Edge and cloud computing in smart cities. Future Internet, 17(3), 118.

Upulie, H. D. I., and Kuganandamurthy, L. (2021). Real-time object detection using yolo: A review. Real-Time Object Detection using YOLO: A review, 196.

Verma, A., Kumar, V., & Yadav, R. K. (2024, March). Humerus bone fracture detection utilizing YOLOv4 algorithm: a deep learning approach. In 2024 2nd International Conference on Disruptive Technologies (ICDT) (pp. 1191-1196). IEEE.

Vijayakumar, A., and Vairavasundaram, S. (2024). YOLO-based object detection models: A review and its applications. Multimedia Tools and Applications, 83(35), 83535-83574.

Wang, C. Y., Yeh, I. H., & Mark Liao, H. Y. (2024, September). Yolov9: Learning what you want to learn using programmable gradient information. In European Conference on Computer Vision (pp. 1-21). Cham: Springer Nature Switzerland.

Xiao-Bo, L., Xiao, X., Ling, W., Zhi-Hua, C., Xin, G., & Ke-Xin, Z. (2023). Anchor-free based object detection methods and their application progress in complex scenes. Acta Automatica Sinica, 49(7), 1369-1392.

Yoon, J. H., Okwuosa, C. N., Aronwora, N. C., & Hur, J. W. (2025). Optimizing Defect Detection on Glossy and Curved Surfaces Using Deep Learning and Advanced Imaging Systems. Sensors, 25(8), 2449.

Zhang, Z., Zhang, Z., Li, G., and Xia, C. (2025). ZZ-YOLOv11: A Lightweight Vehicle Detection Model Based on Improved YOLOv11. Sensors, 25(11), 3399.