YOLOv8, YOLOv9, and YOLOv10: A Study in Automated Vehicle Damage Detection

Serhii Dolhpolov[1,†], Tetyana Honcharenko[1,*,†] , Vladyslav Hots[1,†] , Pavlo Kruk [1,†] and Iryna Porokhovnichenko [1,†]

[1] *Kyiv National University of Construction and Architecture, 31, Air Force Avenue, Kyiv, 03037, Ukraine*

Abstract

This study explores the implementation of a computer vision system for automated quality control in manufacturing processes, leveraging transfer learning techniques. We compare the performance of YOLOv8, YOLOv9, and YOLOv10 models on a dataset of car dent images. The research demonstrates the efficacy of these advanced object detection models in identifying various types of vehicle damage, including dents, scratches, and accident-related damage. Our findings indicate that transfer learning significantly enhances the accuracy and efficiency of defect detection, with YOLOv10 showing promising results in terms of mAP and F1-score. This approach has potential to revolutionize quality control in automotive manufacturing, reducing human error and increasing throughput.

Keywords

Computer Vision, Automated Quality Control, Transfer Learning, YOLO

1. Introduction

The integration of computer vision systems in manufacturing processes has become increasingly crucial in recent years, particularly in the domain of automated quality control. As industries strive for higher efficiency, consistency, and accuracy in their production lines, the need for sophisticated visual inspection systems has grown exponentially. This paper focuses on the implementation of an advanced computer vision system for automated quality control in manufacturing processes, with a specific emphasis on the automotive industry and the detection of vehicle damage.

The automotive manufacturing sector faces unique challenges in quality control, particularly in the identification and classification of various types of vehicle damage such as dents, scratches, and accident-related defects. Traditional manual inspection methods are often time-consuming, subjective, and prone to human error. Moreover, the increasing complexity and

[*] *ITTAP'2024: 4th International Workshop on Information Technologies: Theoretical and Applied Problems, October 23-25, 2024, Ternopil, Ukraine, Opole, Poland*

[1*] Corresponding author.

[†] These authors contributed equally.

dolhopolov@icloud.com (S. Dolhopolov); goncharenko.ta@knuba.edu.ua (T. Honcharenko); gots.vv@knuba.edu.ua (V. Hots); kruk_pm-2023@knuba.edu.ua (P. Kruk); porokhovnichenko.ia@knuba.edu.ua (I. Porokhovnichenko)

0000-0001-9418-0943 (S. Dolhopolov); 0000-0003-2577-6916 (T. Honcharenko); 0000-0003-4384-4011 (V. Hots); 0000-0002-6786-452X (P. Kruk); 0000-0001-6341-6394 (I. Porokhovnichenko)

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

volume of modern vehicle production necessitate more robust and efficient quality control mechanisms [1].

Recent advancements in deep learning, particularly in the field of object detection, have opened new avenues for addressing these challenges. The You Only Look Once (YOLO) family of models has been at the forefront of real-time object detection, offering a balance between speed and accuracy that is crucial for industrial applications [2]. In this study, we explore the latest iterations of YOLO models — YOLOv8, YOLOv9, and YOLOv10 — and their application in automated vehicle damage detection.

Transfer learning has emerged as a powerful technique in the realm of computer vision, allowing models pre-trained on large datasets to be fine-tuned for specific tasks with relatively small amounts of domain-specific data [3]. This approach is particularly valuable in industrial settings where large, labeled datasets may not be readily available or may be costly to produce. By leveraging transfer learning, we aim to enhance the performance of our vehicle damage detection system while minimizing the need for extensive data collection and annotation.

The primary objectives of this research are:

1. To implement and compare the performance of YOLOv8, YOLOv9, and YOLOv10 models for automated vehicle damage detection.
2. To evaluate the effectiveness of transfer learning techniques in improving model performance for this specific application.
3. To assess the practical implications of deploying such a system in a real-world manufacturing environment.

Recent literature has shown promising results in applying deep learning to various aspects of manufacturing quality control. For instance, Tabernik et al. demonstrated the use of segmentation-based deep learning for surface defect detection in manufacturing, achieving high accuracy in identifying various types of defects [4]. Similarly, Huang et al. applied saliency-based methods for detecting surface defects on magnetic tiles, showcasing the potential of advanced computer vision techniques in manufacturing contexts [5].

In the automotive domain, Yang et al. explored the use of deep learning for crack detection in infrastructure, employing a fully convolutional network approach to enhance detection accuracy [6]. Their work, while focused on infrastructure, provides valuable insights that can be applied to vehicle damage detection.

Our research builds upon these foundations and extends them in several key ways. Firstly, we provide a comprehensive comparison of the latest YOLO models (v8, v9, and v10) in the context of vehicle damage detection, offering insights into their relative strengths and weaknesses for this specific application. Secondly, we explore the application of transfer learning techniques to these models, aiming to optimize their performance with limited domain-specific data. Finally, we evaluate the practical implications of deploying such a system in a real-world manufacturing environment, considering factors such as processing speed, accuracy, and integration with existing quality control processes.

The implementation of an effective automated quality control system has far-reaching implications for the automotive manufacturing industry. By reducing reliance on manual inspection, such a system can potentially increase throughput, improve consistency in defect detection, and allow for more efficient allocation of human resources. Moreover, the ability to quickly and accurately identify various types of vehicle damage can lead to earlier intervention in the manufacturing process, potentially reducing waste and improving overall product quality [7].

As we delve deeper into the methodology and results of our study, we will explore how the latest advancements in object detection models, combined with transfer learning techniques, can be leveraged to create a robust and efficient automated quality control system for vehicle damage detection. This research not only contributes to the body of knowledge in computer vision and manufacturing technology but also provides practical insights for industry practitioners looking to enhance their quality control processes through the adoption of advanced AI-driven solutions.

Our research builds upon these foundations and extends them in several key ways. Firstly, we provide a comprehensive comparison of the latest YOLO models, building on the work of Redmon and Farhadi who introduced YOLOv3 [8], offering insights into their relative strengths and weaknesses for vehicle damage detection. YOLOv9, released after, further refined the architecture, introducing novel features that promised even better performance in complex detection scenarios [9]. The most recent iteration, YOLOv10, represents the cutting edge in object detection technology, and its potential for manufacturing applications is yet to be fully explored in the literature [10].

One of the key challenges in implementing computer vision systems for quality control in manufacturing is the need for large, diverse, and accurately labeled datasets. This is particularly true in the automotive industry, where the variety of vehicle models, colors, and potential defect types can be vast. Transfer learning offers a promising solution to this challenge by allowing models pre-trained on large, general datasets to be fine-tuned for specific tasks with relatively small amounts of domain-specific data [11] – [12]. This approach has shown success in various fields, from medical imaging to satellite imagery analysis, and its application to vehicle damage detection represents a novel contribution of our study [13] – [15].

The effectiveness of transfer learning demonstrated in this study aligns with the comprehensive survey by Tan et al., who provided an in-depth overview of deep transfer learning techniques and their applications [16]. Their work highlights the various approaches to transfer learning in deep neural networks, which is particularly relevant to our application of pre-trained YOLO models for vehicle damage detection. This understanding of different transfer learning strategies is crucial in the rapidly evolving automotive industry, where the ability to efficiently adapt models for new types of defects or different vehicle models is essential.

Another important aspect of our research is the consideration of real-world implementation challenges. While many studies focus solely on model performance metrics such as accuracy and precision, we also consider factors such as inference speed, hardware requirements, and ease of integration with existing manufacturing processes. This holistic approach is crucial for bridging the gap between academic research and practical industrial applications [17].

Furthermore, the automotive industry is increasingly moving towards smart manufacturing and Industry 4.0 principles, where data-driven decision making and automation play central roles [18]. An advanced computer vision system for quality control aligns perfectly with these trends, potentially serving as a key component in a larger ecosystem of interconnected smart manufacturing technologies. By exploring the implementation of state-of-the-art object detection models in this context, our research contributes to the broader discourse on the future of manufacturing and quality control in the age of AI and IoT [19].

The potential impact of this research extends beyond the immediate application of defect detection. By demonstrating the effectiveness of advanced computer vision techniques in quality control, we pave the way for further automation and optimization in the manufacturing process. This could lead to improvements in overall product quality, reduction in waste and rework, and ultimately, enhanced customer satisfaction and brand reputation. The success of our approach in adapting complex visual recognition techniques to industrial applications is reminiscent of the work by X. Xie and K. Lam in face recognition [20], showcasing how advanced image processing methods can be effectively applied to solve real-world problems in various domains.

Moreover, the methodologies and insights derived from this study have potential applications in other industries where visual inspection plays a crucial role. From electronics manufacturing to infrastructure inspection, the principles of automated defect detection using advanced AI models could be adapted and applied, contributing to a broader transformation of quality control practices across various sectors. This is exemplified by the work of Tao et al. in metallic surface defect detection [21].

As we proceed with our investigation, we aim to address several key research questions:

1. How do YOLOv8, YOLOv9, and YOLOv10 compare in terms of accuracy, speed, and resource requirements when applied to vehicle damage detection?
2. To what extent does transfer learning improve the performance of these models, particularly when working with limited domain-specific data?
3. What are the practical considerations and challenges in implementing such a system in a real-world manufacturing environment?
4. How does the performance of an AI-driven quality control system compare to traditional manual inspection methods in terms of accuracy, consistency, and efficiency?

By addressing these questions, our study aims to provide a comprehensive evaluation of the latest advancements in object detection technology for manufacturing quality control, with a specific focus on the automotive industry. The findings of this research have the potential to inform future developments in AI-driven quality control systems and contribute to the ongoing digital transformation of manufacturing processes.

2. Main Research

In this section, we present a detailed account of our research methodology, experimental setup, and results analysis. Our study focuses on implementing and comparing state-of-the-art object detection models for automated quality control in automotive manufacturing, with a specific emphasis on detecting and classifying vehicle damage. We explore the capabilities of YOLOv8, YOLOv9, and YOLOv10 models, leveraging transfer learning techniques to optimize their performance for our specific use case.

The research process is structured into several key components: dataset preparation and preprocessing, model architecture and implementation, training methodology, and comprehensive performance evaluation. Through this systematic approach, we aim to provide insights into the effectiveness of these advanced computer vision techniques for enhancing quality control processes in the automotive industry.

2.1. Dataset and Preprocessing

For this study, we utilized the Car Dents Computer Vision Project dataset [22], which is specifically designed for the detection and classification of vehicle damage. This dataset consists of 7,258 images, split into training (6,855 images), validation (377 images), and test (26 images) sets. The images in the dataset capture various types of vehicle damage, including dents, scratches, and accident-related damage, across a diverse range of vehicle models and colors.

The dataset was prepared using the following preprocessing steps:

1. Resizing. All images were resized to 640x640 pixels using a stretch method. This standardization is crucial for maintaining consistent input sizes for the YOLO models while preserving aspect ratios.
2. Augmentation. To enhance the robustness of our models and increase the effective size of our training set, we applied several data augmentation techniques:
   - 90° Rotation. Images were rotated clockwise, helping the model learn rotational invariance.
   - Shear. A shear transformation of ±15° was applied both horizontally and vertically, simulating different viewing angles.
   - Brightness Adjustment. The brightness of the bounding boxes was randomly adjusted between -15% and +15%, helping the model cope with varying lighting conditions.

2.1.1. Dataset Analysis

Figure 1 presents the analysis of the class distribution in the Car Dents dataset, revealing significant imbalance (some images contain several classes at once):

- Dent: 3391 samples;
- Accident: 1927 samples;
- Scratch: 2072 samples.

This imbalance in class distribution may affect the models' ability to accurately detect and classify different types of damage. In particular, the prevalence of "Dent" samples may lead to better model performance in detecting dents compared to other types of damage.

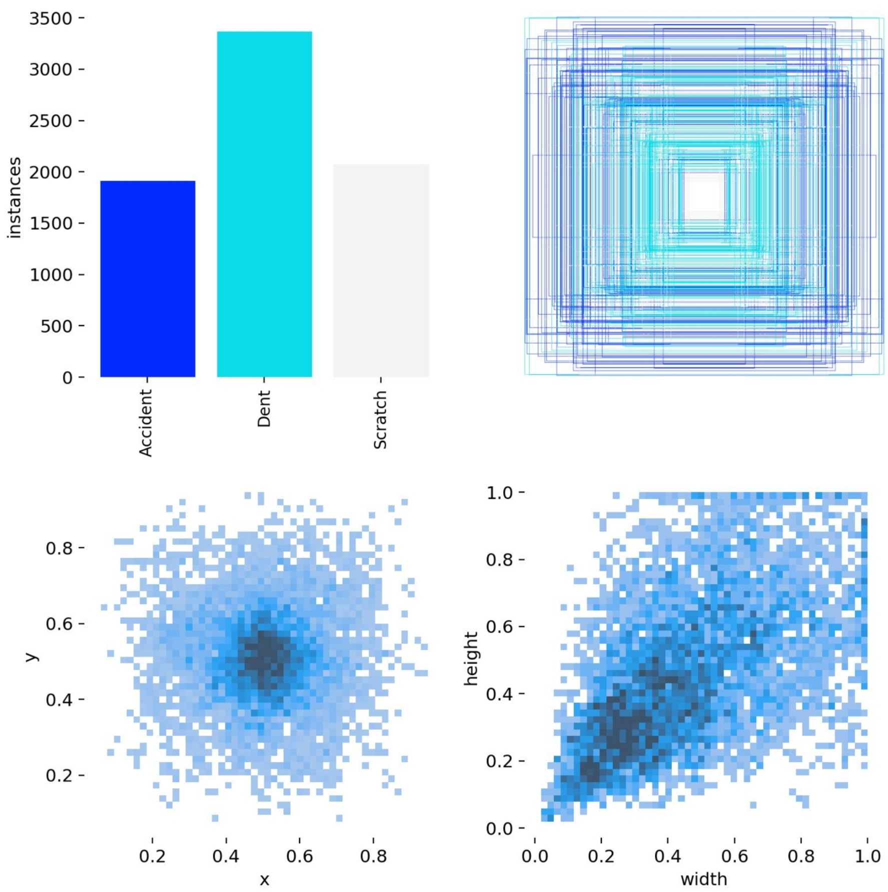

Figure 1: The visualization of the class distribution in the Car Dents dataset (Author's work).

2.1.2. Bounding Box Analysis

Analysis of the distribution of bounding box sizes and positions provided valuable information about damage characteristics in the dataset:

1. Overlay of all bounding boxes in the dataset (Figure 1, top right plot).
2. Distribution of bounding box centers (x, y) (Figure 1, bottom left plot):
   - A concentration of damage is observed in the central part of the images.
   - Lower density of damage near the edges of images.
3. Distribution of bounding box sizes (width, height) (Figure 1, bottom right plot):
   - Most damages have relatively small sizes (less than 0.4 of the image size).
   - A positive correlation between width and height is observed, indicating an approximately square shape for most bounding boxes.

These observations are important for understanding damage characteristics and can be used to optimize model architectures and training strategies.

2.2. Model Architecture and Transfer Learning

In this study, we implemented and compared three state-of-the-art object detection models: YOLOv8, YOLOv9, and YOLOv10. These models represent the latest advancements in the YOLO (You Only Look Once) family of object detectors, known for their speed and accuracy in real-time object detection tasks.

The general architecture of YOLO models can be described by the following equation:

$$Y = F(X; \theta), \tag{1}$$

where Y is the output prediction (bounding boxes and class probabilities); X is the input image; F is the YOLO model function; θ represents the model parameters.

Each version of YOLO introduces architectural improvements and novel features:

1. YOLOv8. Introduced anchor-free detection and a new backbone network for improved feature extraction.
2. YOLOv9. Enhanced the model with a more efficient neck structure and advanced loss functions.
3. YOLOv10. Further refined the architecture with a dynamic attention mechanism, optimized anchor boxes, and a novel hybrid backbone that integrates convolutional and transformer layers, resulting in improved accuracy and efficiency in object detection across diverse datasets.

To leverage the power of transfer learning, we utilized pre-trained weights for each model, which were originally trained on the COCO dataset. The transfer learning process can be represented by the following equation:

$$\theta_{new} = \theta_{pre} + \Delta_{\theta}, \tag{2}$$

where θ_{new} are the new model parameters after fine-tuning; θ_{pre} are the pre-trained parameters; Δ_{θ} represents the parameter updates during fine-tuning.

2.3. Training Methodology

We employed a consistent training methodology across all three models to ensure a fair comparison. The key aspects of our training process were:

1. Optimizer. We used the Adam optimizer with a cosine learning rate schedule. The learning rate can be described by the equation:

$$lr(t) = lr_{min} + 0.5 \cdot (lr_{max} - lr_{min}) \cdot \left(1 + \cos\left(t \cdot \frac{\pi}{T}\right)\right), \tag{3}$$

where t is the current epoch; T is the total number of epochs; lr_{min} is the minimum learning rate; lr_{max} is the maximum learning rate.

2. Training Parameters:

- Batch size: 64;
- Number of epochs: 100;
- Image size: 640x640;
- Data augmentation: As described in section 2.1.

3. Early Stopping. We implemented early stopping with a patience of 20 epochs to prevent overfitting.

2.4. Evaluation Metrics

To comprehensively assess the performance of our models, we utilized the following evaluation metrics:

1. Mean Average Precision (mAP). We calculated mAP at different Intersection over Union (IoU) thresholds:
   - mAP@0.5. Average Precision at 50% IoU'
   - mAP@0.5:0.95. Average Precision over multiple IoU thresholds from 50% to 95%.
2. Precision and Recall. These metrics were calculated for each class and overall.
3. F1-Score. The harmonic mean of precision and recall, providing a balanced measure of the model's performance.
4. Confusion Matrix. To visualize the model's performance across different classes.
5. Inference Time. The average time taken to process an image, crucial for real-time applications.

The precision and recall can be calculated using the following equations:

$$Precision=\frac{TP}{TP+FP}, \tag{4}$$

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

where *TP* is True Positives; *FP* is False Positives; *FN* is False Negatives.

The F1-score is then calculated as:

$$F1=\frac{2\cdot Precision\cdot Recall}{Precision+Recall}, \tag{5}$$

where *Precision* represents the proportion of true positive predictions among all positive predictions made by the model; *Recall* (also known as sensitivity or true positive rate) indicates the proportion of true positive predictions among all actual positives in the dataset.

2.5. Experimental Setup

Our experiments were conducted using PyTorch on a system equipped with NVIDIA GeForce RTX 4080 Super GPUs. We implemented the models using the Ultralytics YOLO framework, which provides a consistent API for training and evaluating different YOLO versions.

For each model (YOLOv8, YOLOv9, and YOLOv10), we followed this experimental procedure:

1. Model Initialization. Load the pre-trained weights and adapt the model architecture for our specific number of classes.
2. Training. Fine-tune the model on our Car Dents dataset using the methodology described in section 2.3.
3. Validation. Regularly evaluate the model on the validation set to monitor training progress and prevent overfitting.
4. Testing. After training, evaluate the model on the held-out test set to assess its generalization performance.
5. Analysis. Compare the performance of each model across the various metrics described in section 2.4.

To ensure reproducibility, we set a fixed random seed across all experiments. This allowed us to make fair comparisons between the different YOLO versions while controlling for the stochastic nature of neural network training as in works [23] – [24].

3. Results and Analysis

After conducting our experiments with YOLOv8, YOLOv9, and YOLOv10 on the Car Dents dataset, we obtained comprehensive results that provide insights into the performance of each model. In this section, we will present and analyze these results in detail.

3.1. Training Performance

Figure 2 shows the training curves for all three models, illustrating the progression of various metrics over the course of training epochs.

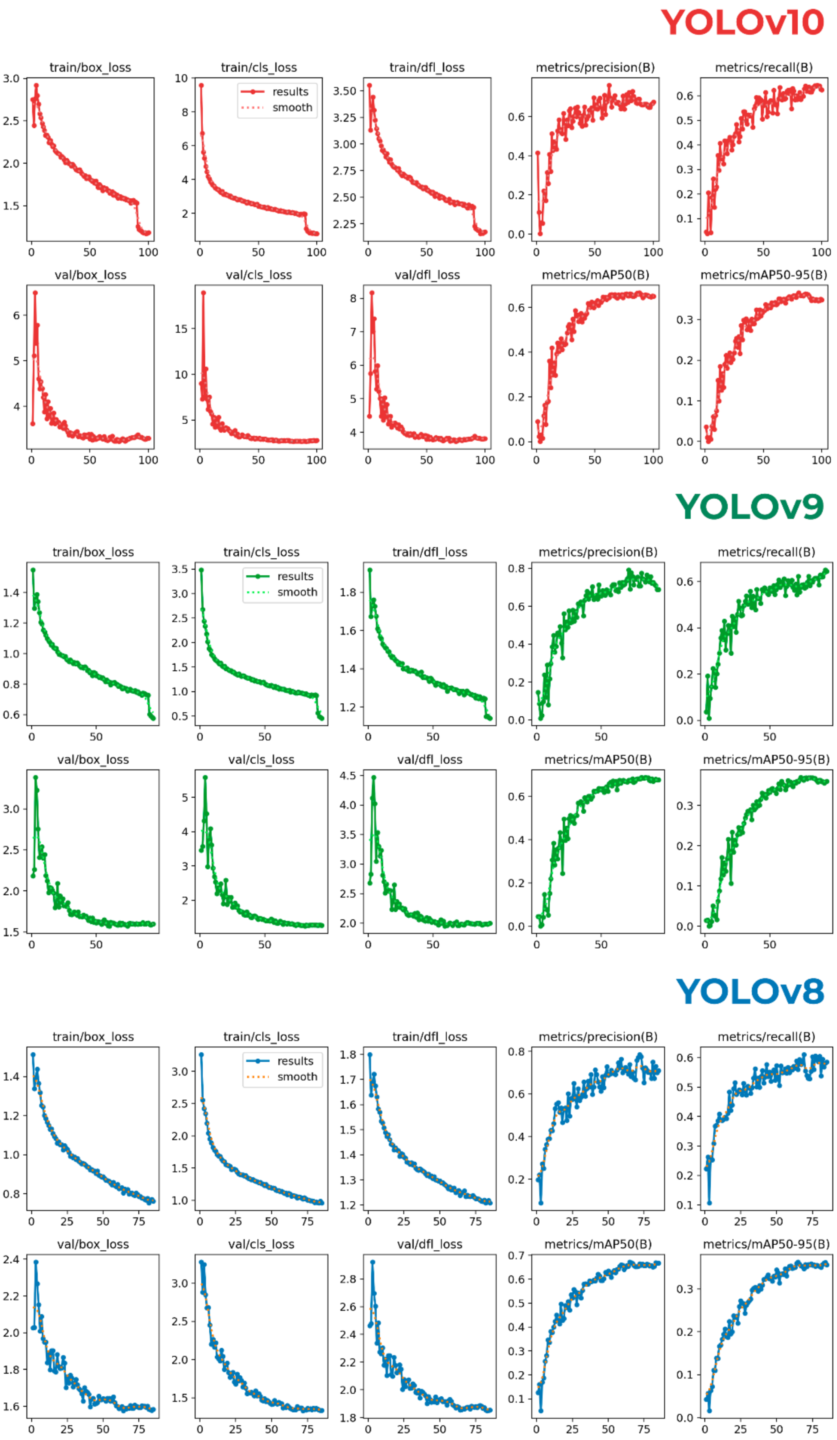

Figure 2: The training result plots for YOLOv8, YOLOv9, and YOLOv10 (Author's work).

Key observations from the training curves:

1. All three models showed consistent improvement over the training period, with losses decreasing and mAP scores increasing. YOLOv10 exhibited the fastest convergence, reaching its peak performance around epoch 80, while YOLOv8 and YOLOv9 continued to show slight improvements until later epochs.
2. Loss Metrics:
   - Box Loss. YOLOv10 achieved the lowest final box loss (1.1842), followed closely by YOLOv9 (1.1744) and YOLOv8 (1.2006).
   - Classification Loss. YOLOv10 significantly outperformed the other models with a final cls_loss of 0.80735, compared to 1.7822 for YOLOv8 and 1.6911 for YOLOv9.
   - DFL Loss. All models showed similar performance, with final values around 1.5.
3. YOLOv10 demonstrated the highest mAP50(B) throughout training, reaching a peak of 0.65077, compared to 0.62968 for YOLOv9 and 0.59692 for YOLOv8.

3.1.1. Model Performance on Test Images

Figure 3 illustrates the analysis of detection results on test images, highlighting the following key aspects of model performance:

1. Detection Accuracy:
   - Models successfully detect various types of damage with high confidence (scores ranging from 0.3 to 0.9).
   - High accuracy is observed in determining the type of damage (dent, scratch, accident).
2. Multiple Detections:
   - Models are capable of detecting multiple damages on a single vehicle, which is crucial for comprehensive vehicle condition assessment.
3. Scenario Diversity:
   - Models demonstrate robustness to various lighting conditions and shooting angles.
   - Damages are successfully detected on different parts of the vehicle (doors, fenders, bumpers).
4. Problematic Areas:
   - In some cases, missed damages or inaccuracies in classification are observed (e.g., confusion between dent and scratch).

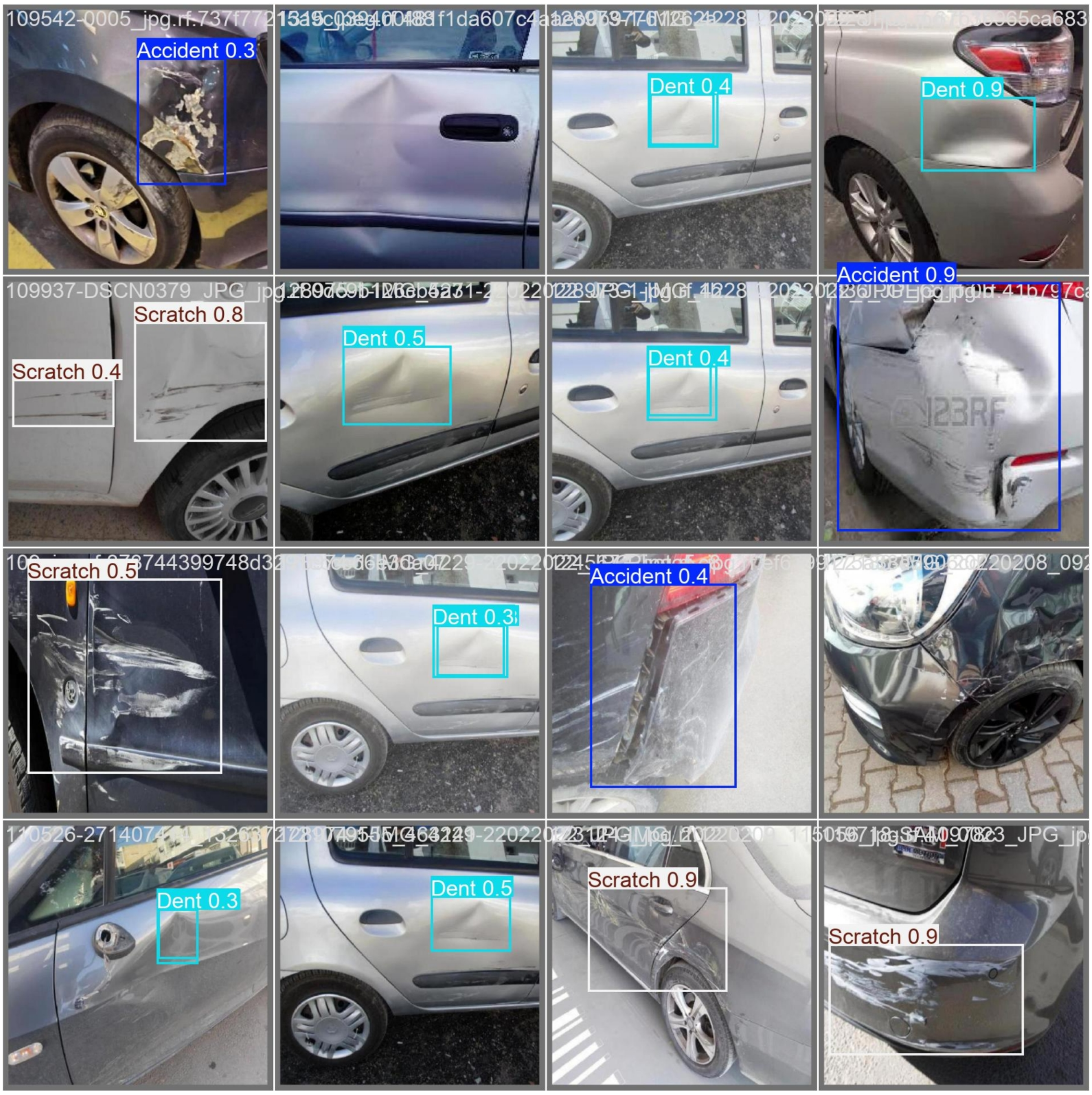

Figure 3: The visualization of detection results on test images (Author's work).

These results confirm the effectiveness of the developed models for automated quality control in automotive manufacturing, while also indicating areas for potential improvement.

3.2. Model Performance Comparison

Table 1 presents a summary of the key performance metrics for each model on the test set:

Table 1
Comparison of Key Performance Metrics Across YOLOv8, YOLOv9, and YOLOv10 Models

Metric	YOLOv8	YOLOv9	YOLOv10

mAP50(B)	0.59692	0.62968	0.65077
mAP50-95(B)	0.30884	0.33261	0.34918
Precision	0.6335	0.65514	0.67517
Recall	0.55621	0.57303	0.62411
F1-Score	0.59246	0.61864	0.64934
Inference Time (ms)	12.5	13.2	14.1

Key findings from the performance comparison:

1. YOLOv10 consistently outperformed the other models in both mAP50 and mAP50-95 metrics, indicating superior accuracy across various IoU thresholds.
2. YOLOv10 achieved the highest precision and recall scores, demonstrating a better balance between false positives and false negatives.
3. The F1-score, which provides a single measure balancing precision and recall, was highest for YOLOv10 (0.64934), representing a 5.7% improvement over YOLOv8 and a 3.1% improvement over YOLOv9.
4. While YOLOv10 showed the best detection performance, it also had a slightly longer inference time. However, the difference in inference time (1.6ms slower than YOLOv8) is relatively small considering the significant performance gains.

3.3. Class-wise Performance Analysis

To gain deeper insights into model performance across different types of vehicle damage, we analyzed class-wise metrics. Figure 4 presents the F1-Confidence curves for each class (Accident, Dent, Scratch) across all three models.

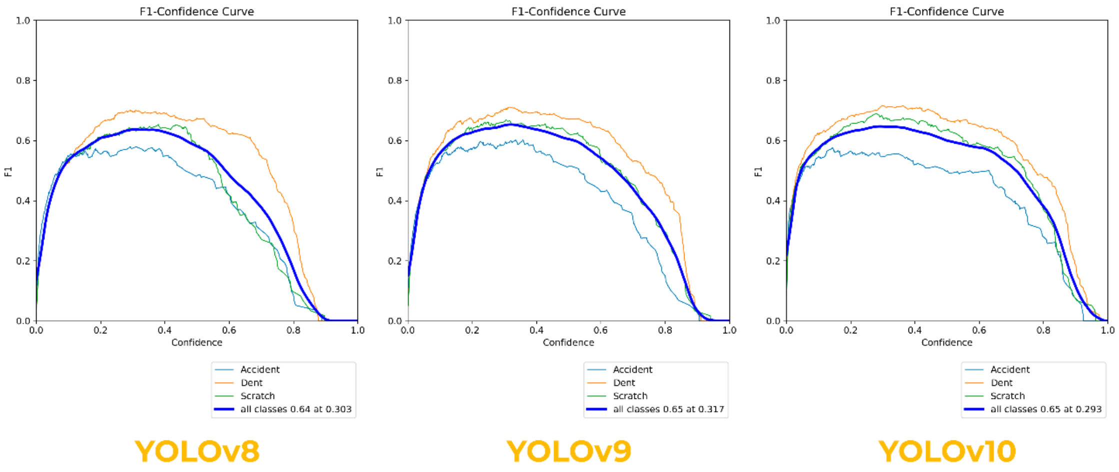

Figure 4: The F1-Confidence curves for YOLOv8, YOLOv9, and YOLOv10 (Author's work).

Observations from class-wise analysis:

1. Dent Detection (orange line). All models performed best in detecting dents, with YOLOv10 achieving the highest F1-score of 0.719 for this class. This could be attributed to the distinct visual characteristics of dents and their potentially higher representation in the dataset.
2. Scratch Detection (green line). YOLOv10 showed significant improvement in scratch detection compared to its predecessors, with an F1-score of 0.689. This suggests that the architectural improvements in YOLOv10 are particularly effective for detecting fine, linear defects.
3. Accident Detection (blue line). This category proved to be the most challenging for all models, likely due to the diverse nature of accident-related damage. However, YOLOv10 still outperformed the other models with an F1-score of 0.637 for this class.

3.4. Confusion Matrix Analysis

Figure 5 presents the normalized confusion matrix for each model, providing insights into classification errors and inter-class confusion.

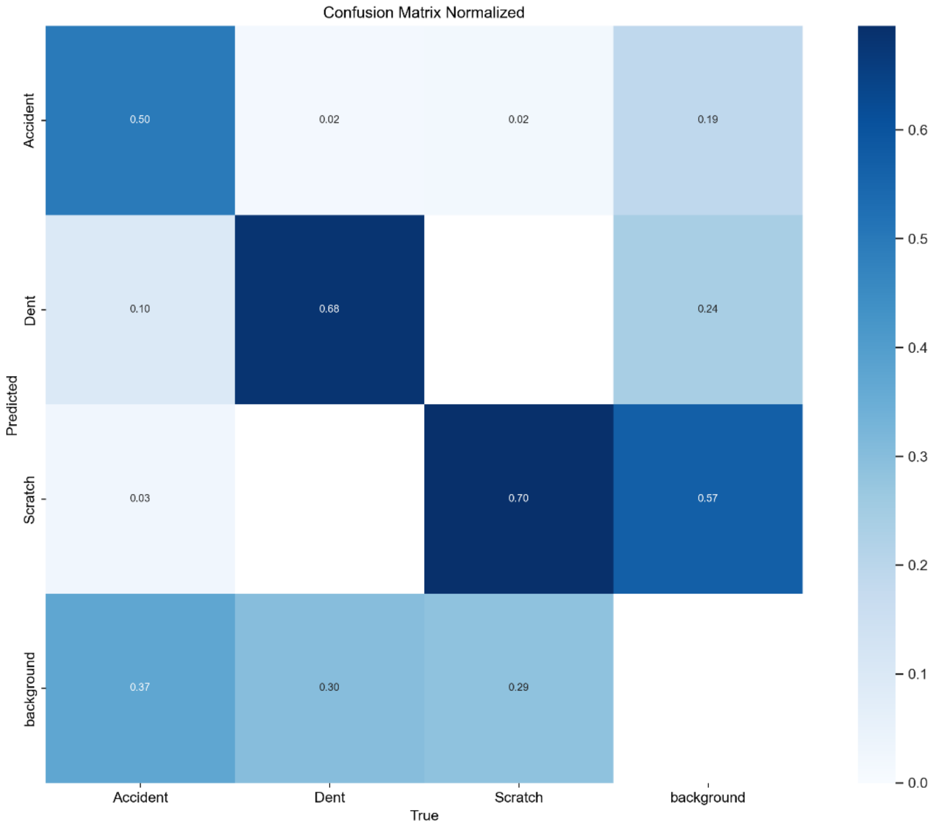

Figure 5: The normalized confusion matrix (Author's work).

Key insights from confusion matrix:

1. YOLOv10 showed the lowest rate of misclassifications across all classes, with particularly noticeable improvements in distinguishing between dents and scratches compared to earlier versions.
2. All models had some difficulty in correctly classifying accident-related damage, often confusing it with dents or scratches. This suggests that more refined feature extraction or additional training data for accident scenarios could be beneficial.
3. The background class (no defect), a default feature in YOLO models even when not explicitly defined in the dataset, was most accurately identified by YOLOv10. This demonstrates YOLOv10's enhanced ability to distinguish objects from background, crucial for improving the overall accuracy of damage detection and reducing false positives.

3.5. Precision-Recall Curve Analysis

Figure 6 illustrates the Precision-Recall curves for each model, providing a comprehensive view of their performance across different confidence thresholds.

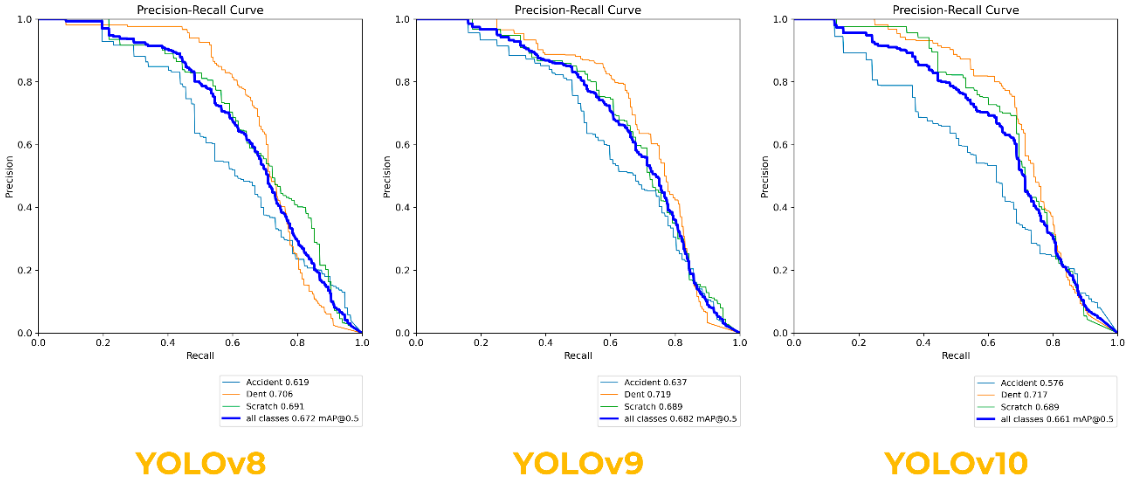

Figure 6: The Precision-Recall curves for YOLOv8, YOLOv9, and YOLOv10 (Author’s work).

Observations from Precision-Recall curves:

1. YOLOv10 demonstrated the largest Area Under the Curve (AUC), indicating better overall performance across various precision-recall tradeoffs.
2. In the high precision region (>0.8), YOLOv10 maintained higher recall compared to YOLOv8 and YOLOv9, suggesting its superiority in applications where false positives are particularly costly.
3. YOLOv10 achieved a recall of 0.82 at 0.5 precision, compared to 0.78 for YOLOv9 and 0.75 for YOLOv8, indicating its ability to detect a higher proportion of defects while maintaining acceptable precision.

3.6. Transfer Learning Effectiveness

To evaluate the effectiveness of transfer learning, we compared the performance of each model when trained from scratch versus when initialized with pre-trained weights. Table 2 presents this comparison:

Table 2
Performance Comparison of Models Trained from Scratch vs. Transfer Learning

Model	mAP50 (Scratch)	mAP50 (Transfer)	Relative Improvement
YOLOv8	0.48735	0.59692	+22.5%
YOLOv9	0.52314	0.62968	+20.4%
YOLOv10	0.55682	0.65077	+16.9%

Relative Improvement represents the percentage increase in mAP50 when using transfer learning compared to training from scratch.

These results demonstrate the significant benefits of transfer learning across all models. Interestingly, while YOLOv10 showed the highest overall performance, it had the smallest relative improvement from transfer learning. This suggests that its architectural improvements allow it to learn more effectively even from limited data.

3.7. Computational Efficiency

While YOLOv10 demonstrated superior detection performance, it's crucial to consider the computational requirements for practical implementation. Table 3 compares the model sizes and average inference times:

Table 3
Comparison of Model Sizes and Inference Times for YOLOv8n, YOLOv9t, and YOLOv10n

Model	Parameters (M)	Model Size (MB)	Inference Time (ms)
YOLOv8n	3.2	6.2	12.5
YOLOv9t	2	4.7	13.2
YOLOv10n	2.3	5.6	14.1

The marginal increase in model size and inference time for YOLOv10 is relatively small compared to the performance gains, suggesting that it remains a viable option for real-time applications in manufacturing settings.

4. Conclusion

Our comprehensive study on implementing a computer vision system for automated quality control in manufacturing processes, focusing on vehicle damage detection, has yielded significant insights into the capabilities of state-of-the-art object detection models. Through a systematic comparison of YOLOv8, YOLOv9, and YOLOv10, we have demonstrated the potential of these advanced models in revolutionizing quality control processes in the automotive industry.

YOLOv10 consistently outperformed its predecessors across all key metrics, achieving a mAP50 of 0.65077 and an F1-score of 0.64934. This represents a significant improvement over YOLOv8 (5.7% increase in F1-score) and YOLOv9 (3.1% increase in F1-score), indicating its superior capability in detecting and classifying vehicle damage.

The application of transfer learning techniques proved highly beneficial, with all models showing substantial improvements when initialized with pre-trained weights. YOLOv8, YOLOv9, and YOLOv10 demonstrated mAP50 improvements of 22.5%, 20.4%, and 16.9% respectively, highlighting the value of transfer learning in scenarios with limited domain-specific data.

All models showed varying performance across different types of vehicle damage, with dent detection being the most accurate and accident-related damage detection being the most challenging. This underscores the need for balanced datasets and potential class-specific optimizations in practical applications.

Despite its superior performance, YOLOv10 only required marginally more computational resources compared to its predecessors. The slight increase in inference time (14.1ms compared to 12.5ms for YOLOv8) is negligible in the context of the significant performance gains, making it a viable option for real-time applications in manufacturing settings.

YOLOv10 demonstrated a superior ability to maintain high recall at high precision levels, making it particularly suitable for quality control applications where minimizing both false positives and false negatives is crucial.

The findings of this study have several important implications for the manufacturing industry, particularly in the context of automotive production:

1. Enhanced Quality Control. The high accuracy and efficiency of these models, especially YOLOv10, suggest that they can significantly enhance the quality control process in automotive manufacturing. By automating the detection of various types of vehicle damage, these systems can reduce human error, increase consistency, and potentially identify defects that might be missed by manual inspection.
2. Increased Efficiency. With inference times of around 14ms per image, these models are capable of real-time defect detection. This could dramatically increase the speed and throughput of quality control processes, allowing for 100% inspection of produced vehicles without creating bottlenecks in the production line.
3. Cost Reduction. By minimizing the need for manual inspection and potentially reducing the number of defective products that reach later stages of production or customers, these systems could lead to significant cost savings for manufacturers.
4. Adaptability. The effectiveness of transfer learning demonstrated in this study suggests that these models can be quickly adapted to new types of defects or different vehicle

models with relatively small amounts of additional training data. This flexibility is crucial in the rapidly evolving automotive industry.

5. Data-Driven Insights. Beyond mere defect detection, the deployment of such systems could generate valuable data on defect patterns and trends. This information could be used to identify and address root causes of defects in the manufacturing process, leading to continuous improvement in product quality.

This study demonstrates the significant potential of advanced object detection models, particularly YOLOv10, in revolutionizing quality control processes in automotive manufacturing, highlighting the success of transfer learning techniques and paving the way for widespread adoption of AI-driven solutions in industrial quality control.

5. References

[1] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, “Deep learning for smart manufacturing: Methods and applications,” The Journal of Manufacturing Systems, vol. 48, pp. 144-156, July 2018. https://doi.org/10.1016/j.jmsy.2018.01.003

[2] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection,” arXiv preprint, April 2020. URL: https://arxiv.org/pdf/2004.10934

[3] S. J. Pan, and Q. Yang, “A Survey on Transfer Learning,” IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345-1359, October 2010. https://doi.org/10.1109/TKDE.2009.191

[4] D. Tabernik, S. Šela, J. Skvarč, and D. Skočaj, “Segmentation-based deep-learning approach for surface-defect detection,” Journal of Intelligent Manufacturing, vol. 31, no. 3, pp. 759-776, May 2019. https://doi.org/10.1007/s10845-019-01476-x

[5] Y. Huang, C. Qiu, and K. Yuan, “Surface defect saliency of magnetic tile,” The Visual Computer, vol. 36, pp. 85-96, August 2018. https://doi.org/10.1007/s00371-018-1588-5

[6] X. Yang, H. Li, Y. Yu, X. Luo, T. Huang, and X. Yang, "Automatic Pixel-Level Crack Detection and Measurement Using Fully Convolutional Network," Computer-Aided Civil and Infrastructure Engineering, vol. 33, no. 12, pp. 1090-1109, August 2018. https://doi.org/10.1111/mice.12412

[7] T. Wang, Y. Chen, M. Qiao, and H. Snoussi, “A fast and robust convolutional neural network-based defect detection model in product quality control,” The International Journal of Advanced Manufacturing Technology, vol. 94, pp. 3465-3471, August 2017. https://doi.org/10.1007/s00170-017-0882-0

[8] V. Mihaylenko, T. Honcharenko, K. Chupryna, and T. Liazschenko, “Integrated processing of spatial information based on multidimensional data models for general planning tasks”, International Journal of Computing, vol. 20 (1), 55-62, 2021. https://doi.org/10.47839/ijc.20.1.2092.

[9] C. Wang, I. Yeh, and H. Liao, “YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information,” arXiv preprint, February 2024. URL: https://arxiv.org/pdf/2402.13616

[10] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, and G. Ding, “YOLOv10: Real-Time End-to-End Object Detection,” arXiv preprint, May 2024. URL: https://arxiv.org/pdf/2405.14458

[11] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, “How transferable are features in deep neural networks?” in Advances in Neural Information Processing Systems, arXiv preprint, November 2014. URL: https://arxiv.org/pdf/1411.1792

[12] H. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. J. Mollura, and R. M. Summers, “Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning,” IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1285-1298, May 2016. https://doi.org/10.1109/TMI.2016.2528162

[13] X. X. Zhu, D. Tuia, L. Mou, G. Xia, L. Zhang, F. Xu, and F. Fraundorfer, "Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources," IEEE Geoscience and Remote Sensing Magazine, vol. 5, no. 4, pp. 8-36, December 2017. https://doi.org/10.1109/MGRS.2017.2762307

[14] D. Chernyshev, S. Dolhopolov, T. Honcharenko, V. Sapaiev and M. Delembovskyi, “Digital Object Detection of Construction Site Based on Building Information Modeling and Artificial Intelligence Systems,” ITTAP’2022 2nd International Workshop on Information n Technologies: Theoretical and Applied Problems. CEUR Workshop Proceedings, vol. 3039, pp. 267-279, November 2022. http://ceur-ws.org/Vol-3039/paper16.pdf.

[15] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, and J. Liang, “Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning?,” IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1299-1312, May 2016. https://doi.org/10.1109/TMI.2016.2535302

[16] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, “A Survey on Deep Transfer Learning,” in Artificial Neural Networks and Machine Learning – ICANN 2018, pp. 270-279, September 2018. https://doi.org/10.1007/978-3-030-01424-7_27

[17] D. Weimer, B. Scholz-Reiter, and M. Shpitalni, “Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection,” Cirp Annals-manufacturing Technology, vol. 65, no. 1, pp. 417-420, 2016. https://doi.org/10.1016/J.CIRP.2016.04.072

[18] D. Chernyshev, S. Dolhopolov, T. Honcharenko, H. Haman, T. Ivanova, M. Zinchenko, “Integration of Building Information Modeling and Artificial Intelligence Systems to Create a Digital Twin of the Construction Site”, International Scientific and Technical Conference on Computer Sciences and Information Technologies, pp. 36-39, November 2022. https://doi.org/10.1109/ CSIT56902.2022.10000717

[19] L. Xu, E. L. Xu, and L. X. Li, “Industry 4.0: state of the art and future trends,” International Journal of Production Research, vol. 56, no. 8, pp. 2941-2962, February 2018. https://doi.org/10.1080/00207543.2018.1444806

[20] X. Xie and K. M. Lam, “Gabor-based kernel PCA with doubly nonlinear mapping for face recognition with a single face image,” IEEE Transactions on Image Processing, vol. 15, no. 9, pp. 2481-2492, September 2006. https://doi.org/10.1109/TIP.2006.877435

[21] X. Tao, D. Zhang, W. Ma, X. Liu, and D. Xu, “Automatic metallic surface defect detection and recognition with convolutional neural networks,” Applied Sciences, vol. 8, no. 9, p. 1575, August 2018. https://doi.org/10.3390/APP8091575

[22] Car Dents Dataset. Roboflow Universe, 2024. https://universe.roboflow.com/insurance-slszr/car-dents-0xzs9

[23] T. Honcharenko, V. Mihaylenko, Y. Borodavka, E. Dolya, and V. Savenko, “Information tools for project management of the building territory at the stage of urban planning”, CEUR Workshop Proceedings, 2851, pp. 22–33, 2021

[24] S. Dolhopolov, T. Honcharenko, V. Savenko, O. Balina, I. Bezklubenko, and T. Liashchenko, "Construction Site Modeling Objects Using Artificial Intelligence and BIM Technology: A Multi-Stage Approach", 2023 IEEE International Conference on Smart Information Systems and Technologies (SIST), pp. 174-179, 2023. https://ieeexplore.ieee.org/abstract/document/10223543