Evaluating the Impact of Image Compression and Resolution on YOLOv11 Performance for Residential Occupancy Verification

Farendi Giotivano[1] , Wiyli Yustanti[2*] , Ricky Eka[3]
Informatics Department, Faculty of Engineering, Universitas Negeri Surabaya
Kampus Ketintang, Jalan Ketintang, Kecamatan Gayungan, Surabaya, Jawa Timur (Kode Pos 60231)
E-mail: farendigiotivano@unesa.ac.id, wiyliyustanti@unesa.ac.id, rickyeka@unesa.ac.id

***Abstract*-The implementation of computer vision in educational assistance verification systems is frequently hindered by high storage requirements and heavy server bandwidth consumption. Despite this, empirical research mapping the tolerance threshold of image degradation against object detection reliability in this domain remains limited. This study aims to evaluate the impact of JPEG compression and resolution reduction on the performance of the YOLOv11 model in identifying residential physical components, specifically roofs, walls, and floors. We conducted evaluations using various JPEG quality scenarios and image resolution adjustments. The experimental results demonstrate that YOLOv11 exhibits high robustness to compression. At a 75% quality level, file size was reduced by up to 70.6% while maintaining stable performance, achieving a precision of 0.7605, recall of 0.6651, mAP50 of 0.7341, and an inference time of 8.80 milliseconds. Regarding the resolution experiments, a 480-pixel dimension provided the most optimal outcome, yielding a precision of 0.7309, recall of 0.6711, mAP50 of 0.7333, and an inference time of 3.60 milliseconds. Conversely, extreme resolution reduction triggered a significant drop in accuracy. These findings confirm that the YOLOv11 architecture is more tolerant of compression quality than it is of resolution scaling. We recommend a 75% JPEG compression or a 480-pixel resolution as a standard operational baseline for efficient and accurate residential verification.**

Keywords— YOLOv11, image compression, image resolution, computer vision, residential verification.

I. INTRODUCTION

The ongoing digital transformation in the education sector has driven higher education institutions to adopt various information technologies to enhance the efficiency of academic and administrative services. A critical process requiring digitalization is the verification of residential conditions for prospective recipients of tuition fee subsidies (Uang Kuliah Tunggal or UKT). Conventional verification processes, which rely heavily on field surveys, are often time-consuming, costly, and labor-intensive. Consequently, the adoption of Computer Vision (CV) technology has emerged as a promising alternative, enabling more rapid and objective identification of residential physical conditions [1].

Recent advancements in Deep Learning have introduced various object detection methods capable of automating recognition tasks with high accuracy. The "You Only Look Once" (YOLO) family, in particular, has become one of the most widely adopted approaches due to its well-balanced trade-off between detection speed and accuracy [2], [3]. Continuous evolution of the YOLO architecture has further improved its feature extraction capabilities and computational efficiency, allowing for deployment across diverse real-world scenarios, ranging from environmental monitoring to object identification and building condition analysis [4], [5], [6].

In the context of residential verification, object detection models can be leveraged to identify key physical building components—such as roofs, walls, and floors—which serve as indicators of a student's living conditions. This approach allows for automated evaluation based on user-uploaded photographs. Existing studies have demonstrated that YOLO-based models deliver robust detection performance across various application domains while maintaining relatively low computational requirements. Furthermore, the integration of Feature Pyramid Network (FPN) mechanisms enables the model to detect objects across multiple scales, making it highly adaptive to variations in object size within an image [7].

Despite these benefits, implementing CV-based systems within a university environment poses significant infrastructure challenges. During each registration or scholarship application cycle, institutions must manage thousands of student-uploaded images. As these images are typically captured using various smartphones, they often feature high resolutions and large file sizes, which place substantial strain on server storage capacity and bandwidth consumption. Such conditions can compromise system operational efficiency, particularly for institutions with constrained infrastructure resources.

A common strategy to mitigate storage burdens is to implement image compression and resolution adjustment before data storage or processing. From a multimedia perspective, image compression aims to reduce file size while maintaining a level of visual quality acceptable to human observers. However, images processed by AI systems possess distinct characteristics compared to those intended for human consumption. Visual information that may appear insignificant to the human eye can serve as critical features for a Deep Learning model's object detection process [8].

Previous research has indicated that image quality degradation can impact the performance of object detection systems. Delbracio et al. [9] highlighted that the impact of compression on detection models is not always linear; at certain compression levels, model performance can be maintained even as file sizes are significantly reduced. Conversely, aggressive resolution reduction can result in the loss of essential spatial information, thereby impairing the model's ability to recognize objects accurately [10], [11]. These findings suggest a critical trade-off between storage efficiency and detection accuracy that warrants further investigation.

Based on the literature reviewed, most existing studies have focused on developing new object detection architectures or improving model accuracy using general datasets such as Microsoft COCO [12]. Research specifically evaluating the influence of image compression and resolution on YOLO performance within the context of residential verification systems remains relatively scarce. Such information is crucial for establishing image upload policies in campus information systems that rely on computer vision technology.

Accordingly, this study aims to evaluate the impact of image compression and resolution on the performance of YOLOv11 in the verification process of student housing for tuition aid applicants. We conducted evaluations using various JPEG quality levels and image resolutions to measure changes in Precision, Recall, mAP50, inference time, and storage efficiency. The primary contribution of this research is to provide a recommended image configuration that strikes an optimal balance between server resource efficiency and object detection reliability. We expect these findings to serve as a benchmark for educational institutions in designing more efficient and scalable AI-based residential verification systems.

II. MATERIALS AND METHODS

A. Dataset

This study utilizes a residential image dataset compiled from the verification process of prospective recipients of the Single Tuition Fee (Uang Kuliah Tunggal or UKT) subsidy. The dataset consists of documentation depicting the physical conditions of houses, collected from various smartphones characterized by differing camera qualities, lighting conditions, shooting angles, and image resolutions. The target detection objects comprise three primary residential components: roofs, walls, and floors.

The dataset was designed to accurately reflect the real-world operational conditions of the student housing verification system. The diverse sources of imagery introduce heterogeneity in image quality, illumination, perspective, and resolution, which is essential for replicating the practical environment of the UKT assistance verification system. Table I outlines the specifications of the dataset.

TABLE I. DATASET SPECIFICATIONS

Parameter	Vakue
Total Dataset	3.000 images
Testing Data	300 images
Object Classes	Roof, Wall, Floor
Label Format	YOLO TXT
Data Source	Student Housing Photographs
Image Format	JPG/JPEG
Device Variance	Multi-smartphone

As shown in Table I, the dataset exhibits a high degree of heterogeneity, making it suitable for testing the model's generalization capabilities against varied image conditions. This diversity is crucial because the performance of Computer Vision systems in real-world deployments is generally affected by non-uniform image quality.

The total dataset comprises 3,000 images that have been annotated using the YOLO format. Out of this total, 300 images (10%) were allocated as a testing set to evaluate the impact of image degradation on model performance. The use of a separate test set ensures that the evaluation results accurately reflect the model's ability to generalize to unseen data.

B. Object Detection Model

The model employed in this study is YOLOv11n (You Only Look Once Version 11 Nano), a variant within the YOLO family designed to achieve an optimal balance between inference speed, computational efficiency, and detection accuracy. YOLO architecture utilizes a single-stage detector approach, where object localization and classification are performed simultaneously in a single forward pass, yielding significantly faster processing speeds than two-stage detectors such as Faster R-CNN [13]. These characteristics make YOLO widely favored in Computer Vision applications requiring rapid response times and real-time detection capabilities.

The selection of YOLOv11n was not arbitrary; rather, it was grounded in prior research focusing on the development of a residential verification system for UKT applicants using object detection. In that previous study, several YOLO models were trained and evaluated to detect physical home components (roofs, walls, and floors). The test results demonstrated that YOLOv11n delivers stable performance with high accuracy on student housing data featuring diverse lighting, shooting angles, and camera qualities.

Beyond accuracy, this research prioritizes practical implementation within a campus information system environment. During each UKT application cycle, the system must process thousands of images uploaded by students. Consequently, the chosen model must not only possess robust detection capabilities but also maintain high efficiency in computational resource utilization and processing time. YOLOv11n fits these requirements due to its relatively low model complexity compared to larger YOLO variants, facilitating easier deployment on servers or web-based applications with constrained resources.

The model utilized in this study is the best-performing pre-trained model (best.pt) from previous research, yielding a Precision of 0.7599, a Recall of 0.6709, and an mAP50 of 0.7352. Utilizing a pre-trained model ensures that any performance variations observed in this study stem strictly from image compression and resolution reduction rather than discrepancies during the model training phase. Thus, the research focus is directed specifically toward evaluating the robustness of YOLOv11n against various degrees of image degradation commonly encountered during image uploads in campus information systems.

C. Image Degradation Scenarios

This study applies two primary image degradation scenarios to evaluate model robustness against visual quality variations: JPEG compression and image resolution reduction. Both scenarios were chosen because they represent the most common storage optimization techniques deployed in web and mobile applications.

TABLE II. TESTING SCENARIOS

Scenario	Variations
JPEG Quality	100%, 75%, 50%, 25%

Resolution	640×640, 480×480, 320×320, 160×160

The parameter variations listed in Table II were selected to simulate typical image upload conditions found in campus information systems. Consequently, the evaluation results can serve as a foundation for establishing standard, efficient image upload policies that preserve object detection performance.

a) JPEG Compression Scenario

In the first scenario, all test images were converted to JPEG format across four distinct quality levels: 100%, 75%, 50%, and 25%. JPEG compression operates by discarding high-frequency information deemed visually insignificant, thereby significantly reducing file size [14]. Each quality variation was stored in a separate testing directory while preserving identical annotation labels.

b) Resolution Reduction Scenario

In the second scenario, images were tested across four spatial resolution variations: 640×640, 480×480, 320×320, and 160×160 pixels. Downsampling was executed using OpenCV's INTER_AREA interpolation method to minimize aliasing artifacts and preserve visual feature integrity. To prevent geometric distortion and object proportion changes during resizing, the aspect ratio was maintained through letterbox padding—incorporating neutral gray borders (stride-compliant) in alignment with the default Ultralytics YOLOv11 preprocessing standards.

D. Testing Procedure

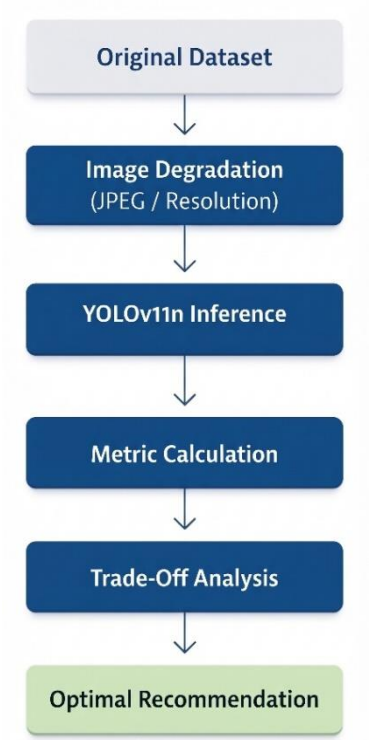


Fig. 1. Research workflow illustrating the evaluation of image compression and resolution impacts on YOLOv11 performance.

Each degraded dataset was evaluated using the identical YOLOv11n model without retraining. Testing was conducted using the validation mode within the Ultralytics framework to ensure uniform treatment across all scenarios.

Fig. 1 illustrates the research stages, beginning with the collection of student housing datasets, followed by image degradation via JPEG quality variations and resolution scaling. Subsequently, each modified dataset was evaluated using the same YOLOv11n model to ensure that performance shifts were exclusively driven by image quality alterations. Inference outcomes were then analyzed using Precision, Recall, mAP50, inference time, and file size metrics to identify the most optimal configuration.

The overall research stages follow an evaluative approach oriented toward measuring the correlation between image quality and object detection system performance [15].

E. Evaluation Metrics

Model performance was evaluated using standard object detection metrics: Precision, Recall, mean Average Precision at an IoU threshold of 0.5 (mAP50), and inference time.

$$mAP = \frac{1}{N}\sum_{i=1}^{N} A\,P_i \tag{1}$$

Precision measures the proportion of correct positive predictions relative to all positive predictions generated by the model.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

Furthermore, this study adopts mAP50 as the primary indicator of detection performance, as it is widely utilized in evaluations of YOLO models and the COCO dataset.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

In addition to performance metrics, storage efficiency was calculated based on file size reductions resulting from compression or resolution adjustment. The percentage of storage savings was computed using the following equation:

$$\text{Storage Saving } \% = \frac{S_{\text{awal}} - S_{\text{akhir}}}{S_{\text{awal}}} \times 100\% \tag{4}$$

F. Data Analysis

Data analysis was performed by comparing shifts in Precision, Recall, mAP50, inference time, and file size across each testing scenario. The experimental results were then visualized as trade-off graphs to identify configurations that strike the best balance between storage efficiency and detection accuracy.

Configurations delivering the largest file size reductions alongside minimal drops in accuracy were established as optimal recommendations for implementing AI-based residential verification systems within higher education environments.

III. RESULT AND DISCUSSION

A. *Impact of Image Compression on YOLOv11 Performance*

Table III presents the experimental results regarding the impact of JPEG compression on YOLOv11 performance in detecting residential components.

TABLE III. YOLOv11 PERFORMANCE UNDER VARYING JPEG COMPRESSION LEVELS

JPEG Quality	mAP50	Precision	Recall	Inference (ms)	File size (KB)
100%	0.7352	0.7599	0.6709	10.39	1442.71
75%	0.7341	0.7605	0.6651	8.80	423.49
50%	0.7259	0.6970	0.6983	8.84	326.59
25%	0.7158	0.7177	0.6665	8.22	207.42

As shown in Table III, JPEG compression has a relatively minor impact on the detection capability of YOLOv11. The mAP50 value decreased only marginally from 0.7352 at 100% quality to 0.7158 at 25% quality. In other words, even under aggressive image degradation, the model retained approximately 97.4% of its initial detection performance.

This phenomenon demonstrates that YOLOv11 is highly robust to JPEG compression artifacts. This robustness can be theoretically explained by the signal processing mechanisms within the frequency domain and the model's hierarchical feature extraction process. JPEG compression employs Discrete Cosine Transform (DCT) to convert spatial image representations into frequency matrices, where quantization eliminates high-frequency coefficients representing fine texture details and noise. Although this lossy compression triggers blocking artifacts at lower quality levels, the YOLOv11 backbone (e.g., C3k2 modules) and neck (SPPF) architectures operate via deep hierarchical convolutions that primarily extract global semantic features, edges, contours, and high-level spatial representations. Since the feature aggregation mechanism in the Feature Pyramid Network (FPN) relies more on object geometry and contours than on the high-frequency texture details lost during DCT quantization, the semantic features of residential components—such as roof patterns, wall lines, and floor areas—remain effectively encoded for classification and bounding box regression.

Another significant finding is the storage efficiency achieved. The average file size was reduced from 1,442.71 KB to 423.49 KB at 75% quality, representing a storage saving of approximately 70.6%. Even at 25% quality, the file size was only 207.42 KB, a reduction of 85.6%. These results indicate that decreasing JPEG quality provides far greater benefits for storage efficiency than the losses incurred in detection accuracy [16].

Furthermore, inference time decreased from 10.39 ms to 8.22 ms. While the improvement is modest, it suggests that smaller file sizes can reduce data processing overhead prior to inference. These findings align with the concept of "Visual Coding for Machines," which asserts that more efficient image representations can enhance the performance of AI-based systems without requiring maximum visual fidelity [17], [18].

B. *Impact of Image Resolution on YOLOv11 Performance*

Table IV shows the experimental results across various image resolution variations.

TABLE IV. YOLOv11 PERFORMANCE UNDER VARYING RESOLUTIONS

Resolution	mAP50	Precision	Recall	Inference (ms)
640×640	0.7185	0.7225	0.7018	4.90
480×480	0.7333	0.7309	0.6711	3.60
320×320	0.6583	0.6727	0.6362	2.58
160×160	0.3378	0.4320	0.3380	1.62

Unlike JPEG compression, changes in resolution have a much more significant impact on model performance. Reducing the resolution to 160×160 pixels caused the mAP50 to drop to 0.3378, a decrease of over 53% compared to the best configuration.

Theoretically, resolution is directly correlated with the amount of spatial information available in an image. When image dimensions are reduced, object edge details, surface textures, and boundaries between components become increasingly difficult to distinguish. Consequently, features previously utilized by the model to recognize objects are lost or distorted [19].

Interestingly, the 480×480 resolution yielded the highest mAP50 at 0.7333, outperforming the standard 640×640 resolution (0.7185). This phenomenon can be explained by the "spatial downsampling regularization effect." At a 640×640 input, the image contains redundant high-frequency details and background noise, such as rough wood textures, wall fibers, or irregular soil patterns, which are irrelevant to the primary localization task. In the anchor-free YOLOv11 architecture, this high-level visual noise can interfere with bounding box regression and object center-ness estimation. Moderate downsampling to 480×480 acts as a natural smoothing filter, suppressing redundant details without discarding crucial spatial information. Consequently, the multi-scale feature fusion module in the FPN is better able to extract cleaner, well-generalized feature representations, thereby enhancing localization precision and component classification.

Regarding computational efficiency, smaller resolutions improve inference speed, with latency dropping from 4.90 ms (640×640) to 1.62 ms (160×160). However, the resulting significant drop in accuracy renders these lower resolutions unsuitable for residential verification systems that demand high reliability.

C. *Trade-off Analysis between Accuracy and Efficiency*

Following the compression process, the majority of the visual information required by YOLOv11 can still be preserved, as illustrated in Fig. 2, despite a relatively gradual decline in mAP50 values, as shown by the graph in Fig. 2. Although JPEG artifacts at low quality begin to emerge, the model can still recognize primary object features such as walls, floors, and roofs. The results indicate that YOLOv11 is highly robust against image quality degradation. JPEG compression is an extremely beneficial tool to use because it is capable of significantly reducing file size with a relatively small drop in accuracy. Consequently, utilizing lower JPEG quality can be an excellent way to save bandwidth and storage capacity without significantly

diminishing the performance of the residential verification system.

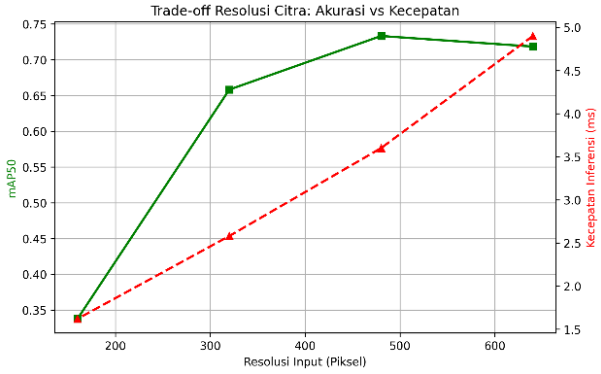


Fig. 2. Impact of JPEG compression on mAP50 values.

In contrast to JPEG compression, the graph in Fig. 3 illustrates a clearer trade-off relationship between accuracy and speed. Resolution reduction yields increasingly faster inference times, but it also degrades the model's ability to recognize objects. The optimal balance point was found at a resolution of 480×480 pixels because it yields the highest accuracy while simultaneously achieving a lower inference speed compared to the 640×640 pixel resolution.

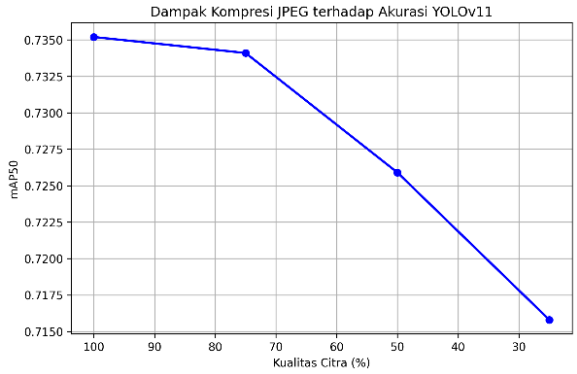


Fig. 3. Trade-off between image resolution, accuracy, and inference speed.

D. Visual Analysis of Detection Results

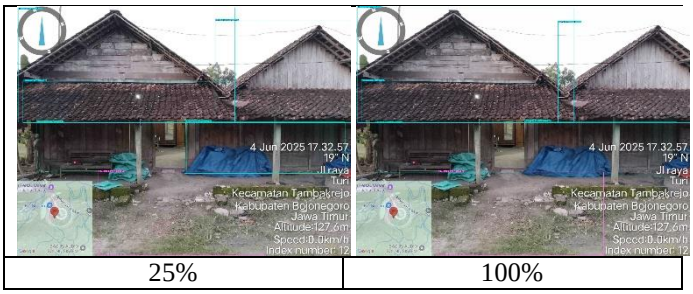


Fig. 4. Comparison of detection results at 100% and 25% JPEG quality.

As shown in Fig. 4, primary objects remain recognizable even when JPEG quality is reduced to 25%. The bounding boxes generated by the model are still able to identify the roof and wall areas with a relatively high degree of accuracy. This indicates that the primary structural information of the objects can still be preserved despite the loss of some texture details due to the compression process.



Fig. 5. The impact of compression artifacts on residential component detection..

Fig. 5 illustrates the emergence of blocking artifacts, which are a primary characteristic of JPEG compression at low quality levels. These artifacts cause a loss of texture details on wall surfaces and floor areas. Nevertheless, the YOLOv11 model remains capable of maintaining most of its detection performance because feature extraction relies more heavily on global shape patterns than on local texture details.



Fig. 6. Object localization failure on high-compression images.

Fig. 6 shows that several objects experience localization errors or fail to be detected altogether. This condition occurs because excessive image degradation causes the loss of essential spatial features, such as object edge boundaries and contrast differences between building components. Consequently, the feature representations received by the model become less informative, thereby simultaneously degrading the model's classification and object localization capabilities.

The visual analysis demonstrates that strong JPEG compression induces blocking artifacts and a loss of texture details across several object areas. However, the majority of bounding boxes are still successfully generated because the model can still globally identify the shapes of roofs, walls, and floors. Conversely, when spatial information is lost, object boundaries become blurred, which increases the likelihood of localization errors. The quantitative results from the mAP50 testing are consistent with these visual findings.

E. Implications for UKT Residential Verification Systems

The research results demonstrate that mandates for high-resolution image uploads do not always yield proportional benefits to system performance improvements. From an operational perspective, utilizing a 75% JPEG quality level can reduce storage requirements by over 70% with an almost negligible drop in accuracy. Meanwhile, a 480×480 pixel resolution delivers the optimal combination of accuracy and inference speed.

Based on these findings, educational institutions can establish a standard image upload policy at 75% JPEG quality and a 480×480 pixel resolution to enhance storage and bandwidth efficiency without compromising the reliability of AI-based residential verification processes. These findings provide an empirical foundation for developing more scalable and economical automated verification systems to support educational aid programs.

IV. CONCLUSION

This study has comprehensively evaluated the impact of image compression and resolution on the performance of the YOLOv11 model for residential verification systems within higher education tuition subsidy (Uang Kuliah Tunggal or UKT) programs, revealing that the architecture is considerably more tolerant of JPEG compression artifacts than of spatial resolution reduction. Specifically, the experimental findings demonstrated that a 75% JPEG quality level successfully reduced file sizes by up to 70.6% while incurring a negligible decline in mAP50, whereas extreme resolution downscaling drastically degraded accuracy due to the loss of vital spatial features, with a 480×480 pixel resolution ultimately delivering the most optimal performance balance with an mAP50 of 0.7333 and an inference time of 3.6 ms. Consequently, the combination of 75% JPEG compression and a 480×480 pixel resolution is strongly recommended as a standard operational baseline for campus information systems, successfully achieving an efficient equilibrium among storage capacity, bandwidth consumption, processing speed, and object detection reliability. Furthermore, these empirical insights provide a scalable and economical foundation for automated educational aid verification, while opening avenues for future research to explore modern compression formats like WebP and HEIC, evaluate alternative detection architectures, and incorporate more diverse mobile upload scenarios.

REFERENCES

[1] M. F. J. Permana, J. C. R. B. Gani, N. A. Fauzan, and A. Adiwilaga, "YOLOv11 Model as a Smart Solution for Waste Identification and Classification in Automated Waste Management System," *Jurnal Ilmu Komputer dan Informasi*, vol. 18, no. 2, pp. 207–216, Jun. 2025, doi: 10.21609/jiki.v18i2.1490.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.

[3] J. Huang *et al.*, "Speed/Accuracy Trade-Offs for Modern Convolutional Object Detectors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 7310–7319. doi: 10.1109/CVPR.2017.351.

[4] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 6517–6525. doi: 10.1109/CVPR.2017.690.

[5] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv preprint arXiv:2004.10934*, 2020, doi: 10.48550/arXiv.2004.10934.

[6] L. Chen, Y. Zhang, and H. Liu, "BFA-YOLO: Improved YOLO Architecture for Facade Detection," *Autom. Constr.*, vol. 154, p. 105012, 2024, doi: 10.1016/j.autcon.2023.105012.

[7] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2117–2125. doi: 10.1109/CVPR.2017.106.

[8] L. Wang, H. Wang, and S. Wang, "A Survey on Image Quality Assessment," *arXiv preprint arXiv:2104.09325*, 2021, doi: 10.48550/arXiv.2104.09325.

[9] M. Delbracio, H. Talebi, and P. Milanfar, "First Gradually, Then Suddenly: Understanding the Impact of Image Compression on Object Detection Using Deep Learning," *Sensors*, vol. 22, no. 3, p. 1104, 2022, doi: 10.3390/s22031104.

[10] Y. Hao, X. Zhang, and Y. Wang, "Understanding the Impact of Image Quality and Distance of Objects to Object Detection Performance," *arXiv preprint arXiv:2205.14820*, 2022, doi: 10.48550/arXiv.2205.14820.

[11] A. K. Venkataramanan, R. Wu, and A. C. Bovik, "Assessing the Impact of Image Quality on Object-Detection AI," *arXiv preprint arXiv:2203.15112*, 2022, doi: 10.48550/arXiv.2203.15112.

[12] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," in *European Conference on Computer Vision (ECCV)*, 2014, pp. 740–755. doi: 10.1007/978-3-319-10602-1_48.

[13] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 91–99. [Online].

[14] M. Ehrlich, L. Davis, S.-N. Lim, and A. Shrivastava, "Analyzing and Mitigating JPEG Compression Defects in Deep Learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2021, pp. 2357–2367. doi: 10.1109/ICCVW54120.2021.00267.

[15] P. Chapman *et al.*, "CRISP-DM 1.0: Step-by-Step Data Mining Guide," 2000. [Online].

[16] W. He, Z. Chen, and K. N. Ngan, "Blind Quality Assessment of Images Containing Objects Using Detection Transformer Features," *Sensors*, vol. 23, no. 5, p. 2500, 2023, doi: 10.3390/s23052500.

[17] A. Marie, P. Le Callet, and others, “Evaluation of Image Quality Assessment Metrics for Visual Coding for Machines,” in *Proceedings of the International Conference on Quality of Multimedia Experience (QoMEX)*, 2023. doi: 10.1109/QoMEX58391.2023.10178456.

[18] D. Gummadi, S. S. Mukherjee, and others, “Evaluating Machine Vision Performance Under Compressed Camera Data,” *arXiv preprint arXiv:2303.04123*, 2023, doi: 10.48550/arXiv.2303.04123.

[19] P. Beniwal, S. K. Singh, and others, “Image Quality Assessment Using Deep Features for Object Detection,” in *Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP)*, 2022. doi: 10.5220/0010887300003124.