STU
SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA

Journal of Electrical Engineering, Vol. 76, No. 5, 2025, pp. 429-442
https://reference-global.com/journal/JEE

# Robustness of YOLO models for object detection in remote sensing images

**Touati Adli, Dimitrije M. Bujaković, Boban P. Bondžulić *, Mohammed Zouaoui Laidouni, Milenko S. Andrić**

Remote sensing imagery enables object detection systems to localize and classify targets for critical applications like surveillance and autonomous driving. However, distortions introduced during image acquisition, transmission, or compression degrade the detection performance, posing challenges for real-world applications. This study conducts a comprehensive robustness evaluation of seven state-of-the-art YOLO models, including YOLOv5, YOLOv7, YOLOv8, YOLOv9, YOLOv10, YOLOv11, and the modified YOLOv5 against four common distortions: Additive White Gaussian Noise (AWGN), JPEG and JPEG2000 compressions, and Gaussian blurring. Using the DOTA-v1.0 dataset, we generated 40 distortion test sets (10 levels per distortion type). The obtained results demonstrate that all distortions degrade performance across all evaluated models. YOLOv9 outperforms others YOLO models in terms of mean average precision under different distortions. YOLOv7 and YOLOv10 exhibit the weakest robustness, whereas YOLOv11 shows low resistance to AWGN distortion.

Keywords: object detection, image distortion, deep learning, remote sensing images, YOLO, DOTA dataset

## 1 Introduction

Object detection in remote sensing images is critical task for a wide range of real-world applications, such as surveillance, environmental monitoring, self-driving, emergency rescue, and geographic information system (GIS) updating [1]. To accomplish this task, recently deep learning object detection based-methods received much attention, due to their ability to handle large amount of data and the strong feature representation capability [2]. The deep learning-based methods can be categorized into two main categories, two-stage object detection algorithms and single-stage object detection algorithms. The two-stage algorithms, such as Faster R-CNN [3] and Mask R-CNN [4], are region-based algorithms. In the first stage, they generate regions of interest (RoIs) from the input image, while in the second stage, they perform object localization and classification. Despite their high accuracy, these methods require significant computational resources due to the large number of RoIs candidates, resulting in lower detection speeds [5]. The single-stage algorithms, such as RetinaNet [6] and the YOLO family (YOLOv5, YOLOv7–YOLOv11) [7-12], directly predict bounding boxes and class probabilities from input image through single architecture, offering trade-off between detection accuracy and real-time inference speed.

Despite the success of deep learning-based object detection algorithms in remote sensing images, their performance remains dependent on high-quality input data, making it vulnerable to image degradation [13, 14]. Numerous studies have explored the impact of image distortions on the performance of deep learning-based object detection and classification algorithms, revealing their performance sensitivity to image distortions and highlighting its limitations in terms of robustness to common signal degradations, data corruptions, and adversarial examples [15]. Early work by Dodge and Karam [16] evaluated the impact of Gaussian blurring, noise, contrast, JPEG, and JPEG2000 compression on the performance of deep learning-based classification models, demonstrating that convolutional neural networks (CNNs) are particularly vulnerable to blur and Gaussian noise, which significantly affect their accuracy. Bakir & Bakir [5] demonstrated that Gaussian noise reduces YOLOv5's performance, with a significant drop when the injected noise proportion reaches 50% into the original input image. Authors in [17] investigated the impact of various noise types and brightness changes on object detection performance of deep CNNs algorithms, including YOLOv5, YOLOv8, and Faster R-CNN. The results revealed that injecting noise and/or reducing image brightness generally degrades the performance of these detection methods. However, in certain scenarios, adding a small amount of noise or reducing illumination levels can enhance object detection. In [18] it is investigated the impact of compression artifacts on the accuracy of CNN-based object detectors, showing their susceptibility to compression degradations. In [19], authors evaluated the

Military Academy, University of Defence in Belgrade, Serbia
adlitouati94@gmail.com, dimitrijebujakovic@gmail.com, * bondzulici@yahoo.com,
mohammedz.laidouni@gmail.com, andricsmilenko@gmail.com

https://doi.org/10.2478/jee-2025-0045, Print (till 2015) ISSN 1335-3632, On-line ISSN 1339-309X

robustness of several object detection methods such as YOLOv4 [20], Mask R-CNN and EfficientDet [21], under various global and local distortions using a new distorted benchmark dataset derived from MS-COCO dataset [22]. Furthermore, the authors in [23] investigated the impact of JPEG compression on the performance of object detection algorithms, such as RetinaNet and Faster R-CNN, revealing a significant drop in detection performance at high compression ratios, following a knee-shaped curve. Further, in [24] it is analyzed the effect of sensor noise on CNN classification performance. The study found that Poisson noise had a minimal impact, while speckle and salt-and-pepper noise, combined with global illumination variations, significantly degrade accuracy. Gaussian and uniform noise had a moderate effect on CNN performance.

In remote sensing context, image distortions pose additional challenges, including geometric distortions, variation in illumination, blurring, noise, and compression artifacts, arising from low-light sensors and bandwidth-constrained transmissions [25]. These degradations severely impact image quality and disturb learning of high-level semantic features of objects within image, leading to reduced detection accuracy [17], and the models' robustness in these cases remains uncertain. Bayerl et al. [26] investigated the impact of remote sensing image compression, including JPEG and JPEG2000 on object classification performance. They demonstrated that mismatched compression encoding between the training and inference phases can significantly degrade classification performance. Similarly, in [27] it is observed a consistent decline in classification accuracy with increasing JPEG2000 compression ratios across multi-source satellite imagery. Authors in [28] showed that compression methods like SPIHT and JPEG2000 reduce classification accuracy in hyperspectral data by destroying spectral-spatial discriminative information. To mitigate such effects, in [29] is proposed a convolutional sparse coding method to quantitatively assess the impact of JPEG2000 compression on remote sensing image classification. The authors in [30] evaluated the impact of JPEG2000 compression at a 12:1 ratio, on a proposed YOLOv5 algorithm for spaceborne targets detection using DOTA-v1.0 dataset [31]. It achieves a balance between computational efficiency and detection accuracy, though with a slight increase in false alarms and missed detection. In [32] it is analyzed the impact of camera calibration parameters (camera distortion correction and gamma correction) and five image parameters (quantization, JPEG compression, resolution, color model, and additional channels) on the performance of object detection algorithms, including Faster R-CNN, EfficientDet, CenterNet [33] and YOLOv4. In [34], the authors investigated the effect of image noise on object localization and classification using Faster RCNN, RetinaNet, YOLOv5 and YOLOv8, finding that higher noise levels significantly reduce detection performance, particularly for detecting small objects.

However, existing studies remain limited in addressing remote sensing-specific challenges. Most existing studies examine distortion effects on object detection and classification performance using natural images (MS-COCO dataset [22]), which lack the unique complexities of remote sensing such as arbitrary object orientations from top-down views, large scale variations, nonuniform object densities, and large aspect ratios [1]. Moreover, researches on distortion effects in remote sensing imagery remains limited to comprehensively cover real-world remote sensing scenarios. In particular, blurring significantly degrades the visibility of small and densely targets within image, making it one of the most critical distortions for object detection in remote sensing images.

Despite these efforts, a few studies have comprehensively evaluated YOLO models under controlled distortions specific in remote sensing images. While authors in [30, 32, 34] addressed certain distortions types, they did not cover critical distortion like blurring, which frequently occur in remote sensing images, due to motion artifacts or atmospheric interference. To bridge this gap, and recognizing the critical need for precise detection in remote sensing images across wide range of scenarios, this study presents a comprehensive analysis of four distortion types, including AWGN, JPEG and JPEG2000 compressions, and blurring on detection performance of state-of-the-art single-stage object detection algorithms: YOLOv5 [7], YOLOv7 [8], YOLOv8 [9], YOLOv9 [10], YOLOv10 [11], YOLOv11 [12], and the modified YOLOv5 [35]. The evaluation is conducted on the DOTA-v1.0 remote sensing dataset [31], across multiple distortion levels.

To this end, the key contributions of this research are:

- Test bed creation: Distorted testing sets are derived from DOTA-v1.0 remote sensing dataset [31], including four distortion types: AWGN, JPEG and JPEG2000 compressions, and blurring with 10 severity levels each.
- Comparative analysis: Robustness evaluation of seven YOLO algorithms on the created distorted testing sets is performed. This analysis highlights the strengths and limitations of each algorithm under various distortion scenarios.

The remain of this paper is organized as follows. Section 2 outlines the materials and methods, describing the selected distortion types and the evaluated YOLO models. Section 3 details the data preparation and the creation of distorted test sets. Section 4 presents the experimental results, analysis, and discussion. Section 5

summarizes the key findings and suggests directions for future research.

## 2 Materials and methods

This section is divided into two parts: (1) distortion types, which includes AWGN, JPEG and JPEG2000 compressions, and blurring used to analyze the distortion effect on object detection performance, and (2) object detection algorithms, that presents an overview of the evaluated YOLO models.

### *2.1 Distortion types*

JPEG is a lossy image compression method widely used to reduce digital image storage size while maintaining acceptable visual quality. It compresses image by transforming it into the frequency domain using the Discrete Cosine Transform (DCT), which separates image details into frequency components. The resulting DCT coefficients are then quantized to reduce storage requirements. JPEG is favored for its efficient trade-off between image quality, file size, and processing speed. Compression quality is controlled by a quality factor ($Q$), typically ranging from 1 to 100, where higher values correspond to lower compression and better image quality [32].

JPEG2000 is an advanced image compression standard that employs wavelet-based technique instead of the DCT used in JPEG. It provides lossy-to-lossless compression, providing higher efficiency and superior image quality, particularly at high compression ratios. JPEG2000 also offers features such as progressive transmission, region-of-interest coding, and scalability, making it well-suited for applications requiring adaptability and interoperability in networked and mobile environments [36]. Compression quality is controlled by the compression ratio ($CR$), where smaller values correspond to lower compression and better image quality.

In this study, Gaussian noise is employed to assess the robustness of YOLO algorithms due to its resemblance to naturally noise types [5]. Gaussian noise typically arises in images due to factors such as the noise of electronic circuits and sensors caused by poor illumination and/or high temperatures. To simulate noise corruption, Gaussian noise with a zero mean and a standard deviation $\sigma_N$ was applied independently to each pixel within each image channel.

Gaussian blurring is a global distortion used in image processing to reduce noise and smooth image details by attenuating high-frequency components. This is achieved through convolution with a Gaussian low-pass filter of standard deviation ($\sigma_B$), applied to each image channel. The standard deviation controls the blur intensity (larger $\sigma_B$ results in severe blurring) [16].

### *2.2 Object detection algorithms*

This study evaluates the robustness of seven YOLO-based object detection models: YOLOv5 [7], YOLOv7 [8], YOLOv8 [9], YOLOv9 [10], YOLOv10 [11], YOLOv11 [12], and the modified YOLOv5 [35].

YOLOv5 is an anchor-based detector. It uses the CSPDarknet53 (Cross Stage Partial Network) in the Backbone part for feature extraction, and combines the Path Aggregation Network (PANet) and Feature Pyramid Network (FPN) for multi-scale feature aggregation in the Neck part. YOLOv5 is available in five scalable variants.

YOLOv7 introduces the Extended Efficient Layer Aggregation Networks (E-ELAN) for efficiently guides computational blocks to capture a diverse range of features. In addition, a new scaling strategy and training optimizations are adopted to balance speed and accuracy.

YOLOv8 is an anchor-free object detection algorithm, which introduced C2f (Cross-Stage Partial Bottleneck with Two Convolutions) modules and a decoupled head for improved gradient flow and accuracy.

YOLOv9 is an anchor-free object detection algorithm, that improves gradient propagation through Programmable Gradient Information (PGI) and employs a GELAN (General Efficient Layer Aggregation Network) Backbone for higher efficiency.

YOLOv10 is a real-time object detection algorithm. It introduces a fully NMS-free (Non-Maximum Suppression-free) with a dual assignment strategy for better label assignment, reducing latency while improving accuracy.

YOLOv11 is a real-time, anchor-free object detection model introduced by Ultralytics in 2024. It builds on YOLOv8 and YOLOv9 by incorporating enhancements such as C3K2 modules, C2PSA (advanced attention mechanism) blocks, and SPPF module (Spatial Pyramid Pooling-Fast), which improves its capacity to handle spatial information while preserving fast inference.

The modified YOLOv5 [35] integrates Swin Transformer v2 into the Backbone, a bi-directional feature pyramid network (BiFPN) [21] in the Neck, and an additional detection head for improving small target detection in remote sensing images.

## 3 Data preparation

This section presents the dataset used in the evaluation of YOLO models robustness. At the beginning, the original DOTA-v1.0 dataset is introduced. Next, the adaptation process made to DOTA-v1.0 dataset for YOLO training/testing is described in detail. Finally, the generation details and parameters of four distortions are provided.

### *3.1 Dataset description and partitioning*

DOTA-v1.0 [31] is a large-scale geospatial object detection dataset, which consists of 15 different object categories: baseball diamond, basketball court, bridge, harbor, helicopter, ground track field, large vehicle, plane, ship, small vehicle, soccer ball field, storage tank, swimming pool, tennis court, and roundabout. This dataset contains a total of 2806 aerial images obtained from different sensors and platforms with multiple resolutions. There are 188282 object instances labeled by an oriented bounding box. The sizes of images range from 800×800 to 4000×4000 pixels. Each image contains multiple objects of different scales, orientations and shapes. The proportions of the training set, validation set, and testing set in DOTA-v1.0 are 3/6, 1/6, and 2/6, respectively.

To adapt the DOTA-v1.0 dataset for YOLO training/testing, several preprocessing steps are applied. Firstly, the original training set of DOTA-v1.0 was split into a training, validation, and test set, preserving the division ratio of the original dataset: 3/6, 1/6, and 2/6, respectively. Secondly, a patches of 1024×1024 pixels were cropped from the original images with a stride of 824 pixels (overlapping of 200 pixels), due to the high resolution of the original images. Thus, the resulting dataset contains 7874 images (93152 instances) for training, 2519 images (30600 instances) for validation, and 5356 images (65352 instances) for testing. Since most YOLO models are not designed for oriented bounding boxes, we generated horizontal bounding boxes by calculating the axis-aligned bounding boxes over original (oriented) annotated bounding boxes of DOTAv-1.0.

### *3.2 Distortion generation*

To assess the robustness of the algorithms under varying degrees of image degradation, distorted test sets were created, focusing on distortions commonly encountered in real-world applications. These include JPEG and JPEG2000 compressions, AWGN, and Gaussian blurring. For each type of distortion, 10 testing sets were generated, with each set representing a specific level of distortion as shown in Tab. 1.

For AWGN, 10 levels were generated in our experiments by varying the variance parameter $\sigma_N^2$ from 0.001 to 0.05 using the MATLAB *imnoise* function, while a zero-mean Gaussian noise was added independently to each pixel.

For JPEG compression, 10 levels of compression were generated in our experiments by varying the quality parameter ($Q$ factor) from 5 to 50 in steps of 5 using the MATLAB *imwrite* function. The $Q$ factor, ranging from 1 to 100, controls the compression level, where value $Q$=100 corresponds to low compression and high quality, and lower values of $Q$ result in higher compression and reduced image quality.

For JPEG2000 compression, 10 levels of compression were generated in our experiments by varying the parameter Com-pression Ratio ($CR$) from 50 to 500 in steps of 50 using the MATLAB *imwrite* function and the ". j2k" file extension.

**Table 1.** Distortion levels for JPEG, JPEG2000, AWGN, and blurring distortions

| Distortion \ Levels | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| JPEG compression ($Q$) | 50 | 45 | 40 | 35 | 30 | 25 | 20 | 15 | 10 | 5 |
| JPEG2000 compression ($CR$) | 50 | 100 | 150 | 200 | 250 | 300 | 350 | 400 | 450 | 500 |
| AWGN ($\sigma_N^2$) | 0.001 | 0.002 | 0.004 | 0.006 | 0.008 | 0.01 | 0.02 | 0.03 | 0.04 | 0.05 |
| Blurring ($\sigma_B$) | 0.5 | 1 | 1.5 | 2 | 2.5 | 3 | 3.5 | 4 | 4.5 | 5 |

For Gaussian blurring distortion, 10 levels were generated in our experiments by varying the standard deviation $\sigma_B$ from 0.5 to 5 with a step of 0.5 using the MATLAB *imgaussfilt* function. Each pixel was blurred through convolution with a Gaussian low-pass filter of standard deviation $\sigma_B$.

Figure 1 presents examples of image distortion levels for the four distortion types analyzed in this study. The images are extracted from the created testing sets. Each row corresponds to a specific distortion type, while the columns represent three selected distortion levels: low, medium, and high. For each level, a representative parameter value is provided, such as JPEG quality factor ($Q$ = 50 for low, $Q$ = 25 for medium, and $Q$ = 5 for high). It provides a visual overview of how distortions increase in severity for each type, helping to understand the impact on image quality.

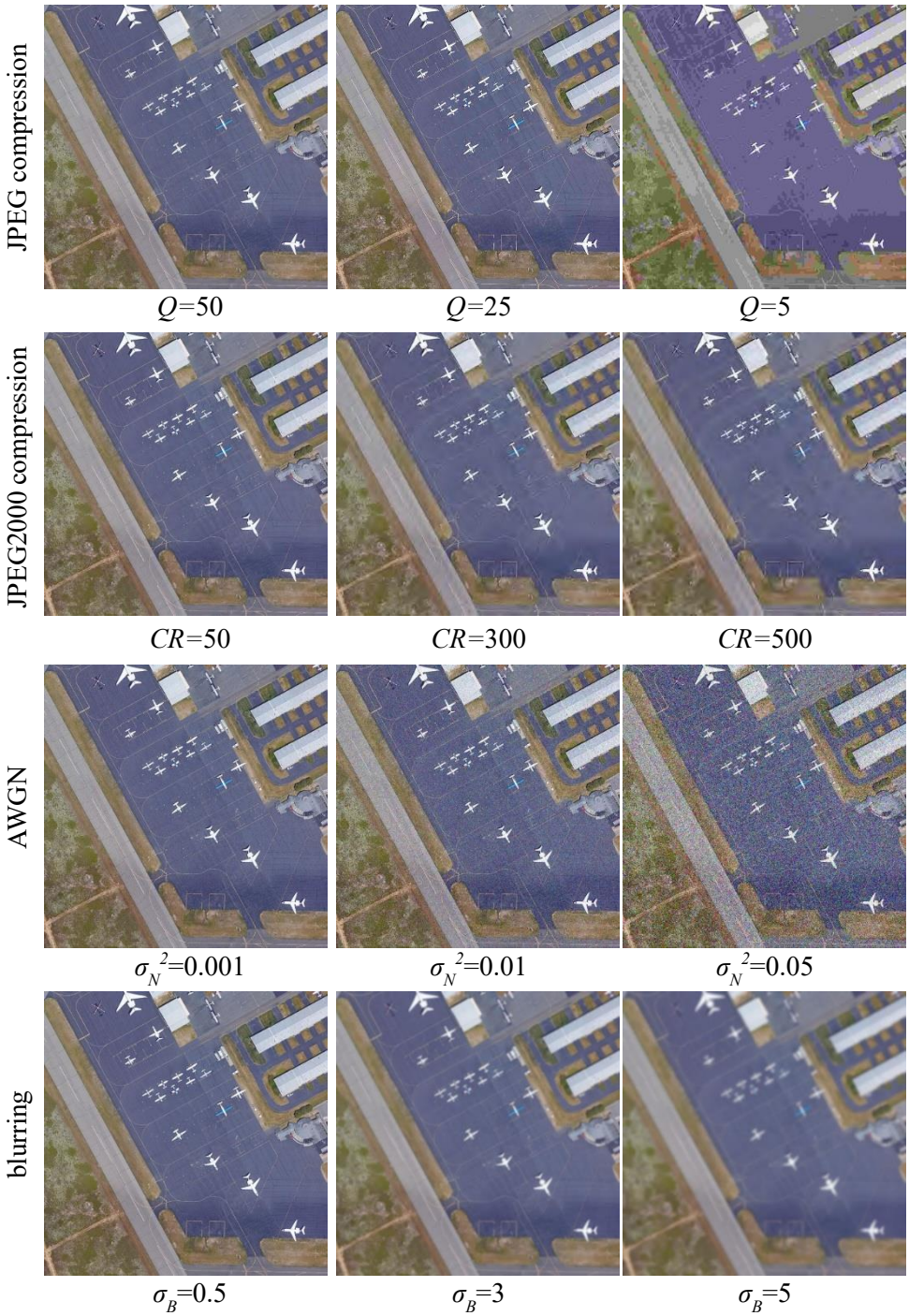



**Fig. 1.** Examples of four distortion types (JPEG, JPEG2000, AWGN, blurring) at low, medium, and high distortion levels

## 4 Experimental results and analysis

### *4.1 Implementation details*

The experimental setup operates on Windows 11, equipped with an Intel® Core™ i7-12650H/16GB @2.30GHz, and NVIDIA RTX 3050Ti/4GB. The algorithms were implemented using the PyTorch framework (version 2.0.1) and accelerated with CUDA Toolkit (version 11.8).

To evaluate YOLO models robustness, the common widely used evaluation metrics are employed, including precision (P), recall (R), mean average precision (mAP) averaged for intersection over union IoU∈{0.5, 0.55, 0.6, …, 0.95} (COCO's standard metric) and mean average precision ($mAP_{0.5}$) at IoU=0.5 (PASCAL VOC's metric) [3].

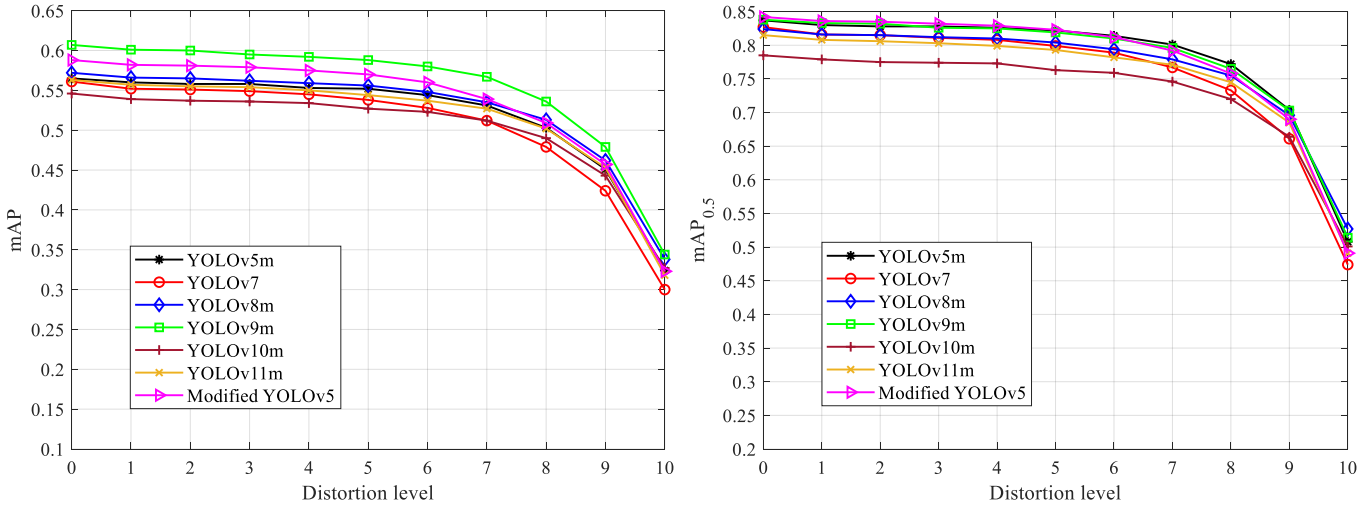
Initially, the algorithms are trained on the original high-quality images in the training sets. Then, the trained models are tested on a specifically created distorted test sets containing distorted images to assess performance degradation relative to the applied distortions. The training process used the Stochastic Gradient Descent (SGD) algorithm for optimizing the loss function. Key training hyperparameters include a momentum of 0.937, a weight decay coefficient of 0.0005, and an initial learning rate of 0.01. The batch size is set to 4, and the number of epochs is 300. The input image resolution is 640×640 pixels.

### *4.2 Detection performance under individual distortions*

In this part, the effect of four degradation types on the performance of target detection in remote sensing images is examined. Each degradation was applied at ten intensity levels, enabling a comprehensive assessment of algorithm robustness under conditions that simulate real-world scenarios. The performance evaluation uses mAP and $mAP_{0.5}$ quantitative measures.

#### *4.2.1 JPEG compression analysis*

Figure 2 illustrates the impact of JPEG distortion levels on the object detection performance mAP and $mAP_{0.5}$, for seven object detection models, including YOLOv5m, YOLOv7, YOLOv8m, YOLOv9m, YOLOv10m, YOLOv11m, and a modified YOLOv5.




(a) (b)

**Fig. 2.** Impact of JPEG distortion on object detection performance for YOLOv5m, YOLOv7, YOLOv8m, YOLOv9m, YOLOv10m, YOLOv11m, and the modified YOLOv5, (a) mAP and (b) $mAP_{0.5}$

The distortion levels are indexed from 1 to 10, where 1 represents the minimal distortion level and 10 the maximal, while zero indicates the case without distortion. The detection performance, measured in terms of mAP and $mAP_{0.5}$, decreases as JPEG distortion levels increase. For the first six levels of degradation (up to the quality factor $Q$=25), this decline is small, but beyond that, performance drops significantly. This trend is consistent across all models, highlighting the significant challenges posed by JPEG compression on detection performance.

For $mAP_{0.5}$, the modified YOLOv5 achieves the best results at lower distortion levels, indicating its greater robustness in these conditions. At higher distortion levels, the original YOLOv5m and YOLOv9m slightly outperforms the modified YOLOv5.

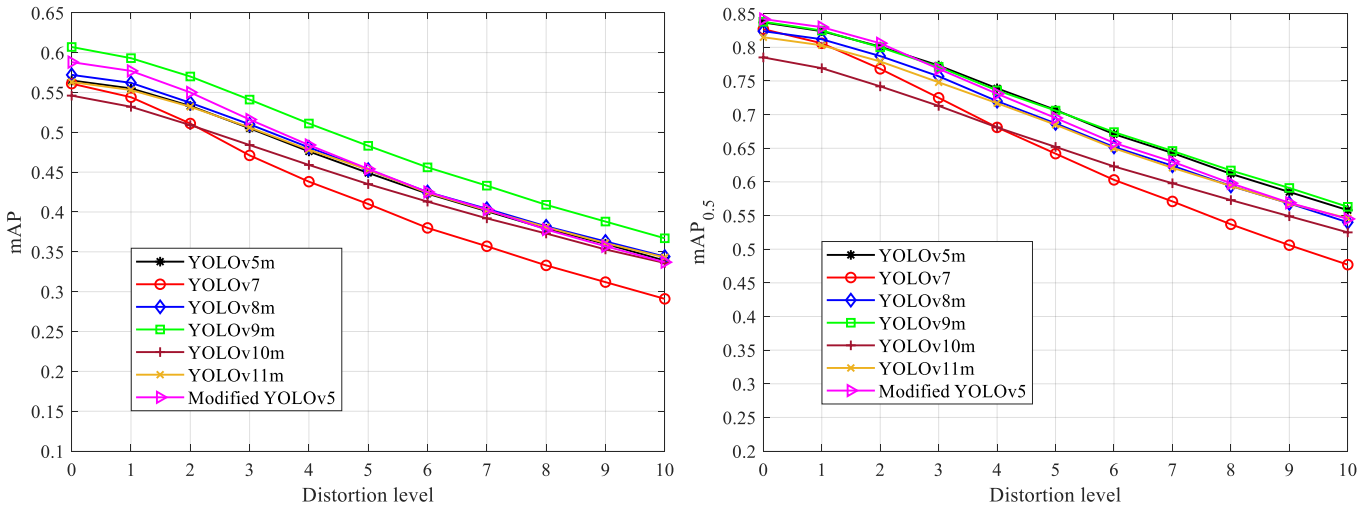
For mAP, YOLOv9m outperforms all other algorithms across all distortion levels, showing a clear performance gain and strong robustness to JPEG compression. YOLOv5m, YOLOv8m and YOLOv11m show comparable performance across all distortion levels. From level 1 to 7, the modified YOLOv5 ranks second, while at higher distortion levels, it performs similarly to YOLOv5m, YOLOv11m, and YOLOv8m, with no model exhibiting a clear dominance. Notably, for both mAP and $mAP_{0.5}$, YOLOv10m records the lowest performance at low compression levels, while YOLOv7 shows the weakest results at high compression levels, indicating its vulnerability to high JPEG compression.

### *4.2.2 JPEG2000 compression analysis*

Figure 3 displays a comparison of the effect of JPEG2000 compression on object detection performance, measured in terms of mAP and $mAP_{0.5}$, for seven object detection algorithms. For all detection algorithms, both mAP and $mAP_{0.5}$ decrease progressively with increasing JPEG2000 distortion level, following a linear degradation trend. This reflects a proportional relationship between JPEG2000 compression severity and the drop in detection performance.

For $mAP_{0.5}$, the modified YOLOv5 demonstrates the best detection results at low distortion levels (1–2), outperforming all other models. Beyond these levels, YOLOv5m and YOLOv9m slightly surpasses the others, while the modified YOLOv5, YOLOv8m, and YOLOv11m show comparable performance. In contrast, YOLOv10m consistently records the weakest performance from levels 1 to 3, while at higher JPEG2000 compression levels (4–10), YOLOv7 shows the lowest results indicating its susceptibility to JPEG2000 compression artifacts.

For mAP, YOLOv9m surpasses all other algorithms across all distortion levels, demonstrating the highest robustness to JPEG2000 distortion. The modified YOLOv5 slightly outperforms YOLOv7, YOLOv8m, YOLOv10m, and YOLOv11m from distortion levels 1 to 4, while at higher distortion levels (5–10), the modified YOLOv5, YOLOv5, YOLOv8m and YOLOv11m achieve similar performance. Furthermore, YOLOv7 shows a sharper decline in mAP from levels 2 to 10, indicating lower resistance to JPEG2000 compression.



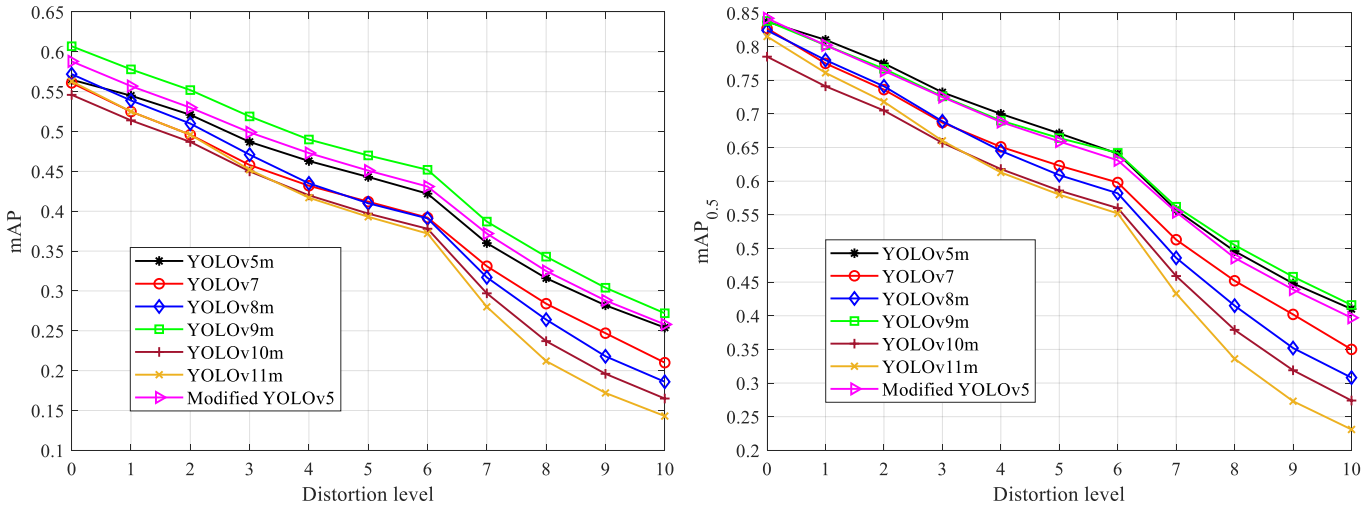

(a) (b)

**Fig. 3.** Impact of JPEG2000 distortion on object detection performance for YOLOv5m, YOLOv7, YOLOv8m, YOLOv9m, YOLOv10m, YOLOv11m, and the modified YOLOv5, (a) mAP and (b) $mAP_{0.5}$

### *4.2.3 AWGN analysis*

Figure 4 highlights the robustness of the YOLOv5m, YOLOv7, YOLOv8m, YOLOv9m, YOLOv10m, YOLOv11m, and the modified YOLOv5 algorithms in maintaining detection performance under ten levels of AWGN distortion, evaluated in terms of mAP and $mAP_{0.5}$.

The mAP and $mAP_{0.5}$ exhibit a gradual decline as the AWGN distortion level increases across all algorithms, following a piecewise linear degradation trend. A noticeable drop in both mAP and $mAP_{0.5}$ is observed, particularly at higher noise levels, indicating that greater noise severity significantly reduces detection performance (for $\sigma_N^2$ greater than 0.01).




(a) (b)

**Fig. 4.** Impact of AWGN distortion on object detection performance for YOLOv5m, YOLOv7, YOLOv8m, YOLOv9m, YOLOv10m, YOLOv11m, and the modified YOLOv5, (a) mAP and (b) $mAP_{0.5}$

YOLOv5m, the modified YOLOv5, and YOLOv9m demonstrate comparable $mAP_{0.5}$ performance with YOLOv5 showing a slight advantage in low levels (1–6) and beyond these levels YOLOv9m exhibit a slight gain, consistently outperforming other models. YOLOv7 and YOLOv8m perform similarly during the initial distortion levels (1–4). However, at higher levels, YOLOv7 demonstrates superior robustness, maintaining better $mAP_{0.5}$ performance than YOLOv8m, YOLOv10m and YOLOv11m.

For mAP, YOLOv9m and the modified YOLOv5 show superior robustness to AWGN distortion, achieving the highest mAP results across all distortion levels. This highlights its exceptional detection capability even under high noise conditions, making them the most resilient algorithms among the evaluated models. In contrast, for both mAP and $mAP_{0.5}$, YOLOv10m and YOLOv11m show the lowest resistance to AWGN distortion.

*4.2.4 Blurring analysis*

Figure 5 illustrates the impact of blurring distortion on the object detection performance of YOLO models, evaluated using mAP and $mAP_{0.5}$ across ten distortion levels. Blurring distortion adversely affects object detection performance, posing challenges for all seven algorithms in detecting targets. At low distortion levels, mAP and $mAP_{0.5}$ exhibit a slow degradation. However, from levels 3 to 10, the decline becomes more pronounced. This reflects the increasing difficulty in detection caused by severe blurring.

The modified YOLOv5 achieves the highest $mAP_{0.5}$ scores from levels 1 to 5. At higher distortion levels (6–10), YOLOv9m and YOLOv11m outperform the other algorithms. where YOLOv11m shows a superior performance, indicating the highest resistance to blurring distortion.

For mAP, YOLOv9m outperforms the other algorithms from levels 1 to 5, while at higher levels of blurring (6–10), it is surpassed by YOLOv11m, which becomes the dominant algorithm. In contrast, YOLOv7 exhibits the lowest detection performance and experiences the most severe decline as distortion levels increase, highlighting its sensitivity to blurring distortion.

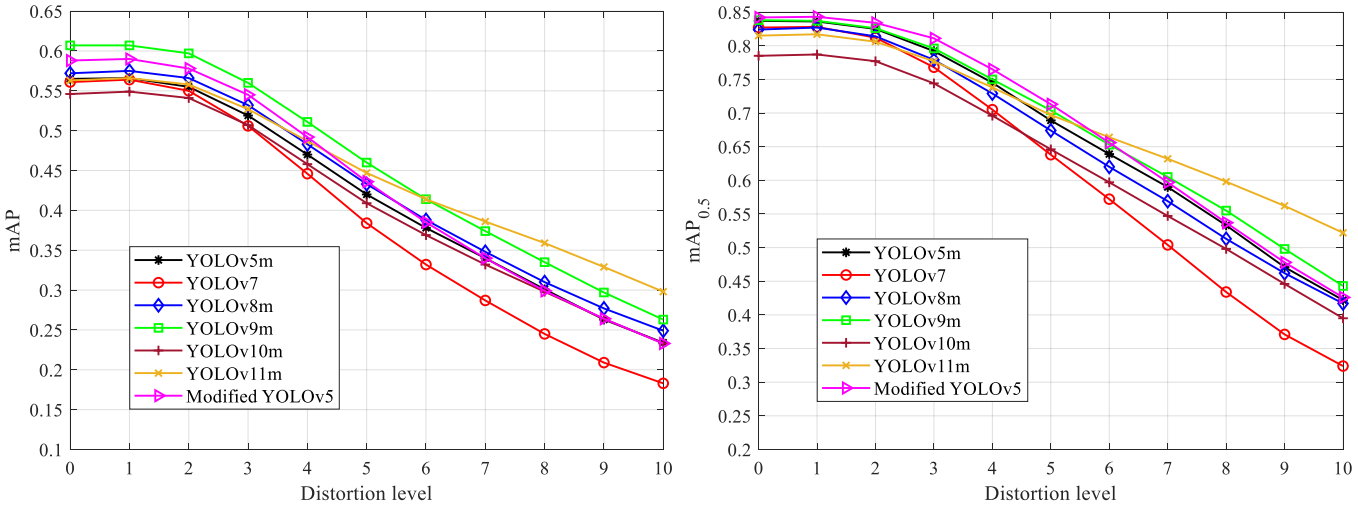



**Fig. 5.** Impact of blurring distortion on object detection performance for YOLOv5m, YOLOv7, YOLOv8m, YOLOv9m, YOLOv10m, YOLOv11m, and the modified YOLOv5, (a) mAP and (b) $mAP_{0.5}$

### *4.3 Detection performance per each class*

Figure 6 presents the mAP variation for the fifteen classes of the DOTA-v1.0 dataset across ten distortion levels of JPEG and JPEG2000 compressions, AWGN and blurring. The upper value of each bar represents the mAP for the original (undistorted) images, while the lower value represents the minimum mAP obtained for the higher distortion levels. A smaller variation in mAP indicates higher resilience to the distortion. There is a noticeable difference in mAP variation across classes and between algorithms. This can be attributed to differences among classes, including shape and size, as well as the impact of background.

For JPEG compression, object classes such as bridge, soccer ball field, basketball court, ground track field, baseball diamond, tennis court and roundabout show a higher mAP variability cross all algorithms compared to other classes. This is due to their high pixel representation, which is more affected by high compression, reducing their detectability. The variability in mAP under JPEG2000 compression is higher than under JPEG compression for object classes such as large vehicle, ship and small vehicle due the complex background and densely distributed targets within image. In contrast, certain object classes such as basketball court, ground track field, and tennis court show lower mAP variation compared to JPEG compression. Furthermore, object classes including helicopter and plane exhibit resilience to JPEG2000 compression, maintaining relatively high and stable mAP values even at high distortion levels.

AWGN significantly affects all object classes, leading to a high variability in mAP across all algorithms. This effect is particularly noticeable in YOLOv10m and YOLOv11m, while mAP values approach to zero at higher distortion levels, particularly for object classes such as bridge, soccer ball field, basketball court, ground track field, baseball diamond, tennis court, and roundabout. The important mAP degradation for these classes is attributed to the large target size in the images, making them more susceptible to noise that obscures critical details and renders the targets unidentifiable. In contrast, YOLOv9m and the modified YOLOv5 algorithm exhibits smaller mAP variations for targets such as large vehicle, helicopter, plane, ship, and small vehicle compared to other algorithms, emphasizing its enhanced detection ability in noisy environment.

Similar to AWGN, blurring is also an influential distortion type, exhibiting high variation in mAP across all algorithms. The effect varies by object class; for instance, the detection of helicopters, ships, and small vehicles is more significantly impacted due to reduced edge clarity, diminished texture detail, and lower discrimination between the target and background (particularly for ship and helicopter). These factors make them more challenging for detection algorithms to accurately differentiate objects from their surroundings.

The analysis of distortions on detection performance highlights the significant impact of image quality on object detection across all classes in the DOTA-v1.0 dataset. The analysis reveals that the object classes with small pixel representations such as large vehicle, helicopter, plane, ship, and small vehicle exhibit a high variability of mAP under blurring distortion and minor mAP variability under JPEG compression. In contrast, the object classes with large pixel representation such as basketball court, ground track field, baseball diamond, and roundabout exhibit lower mAP variability under blurring distortion compared to mAP variability under AWGN distortion and JPEG compression.

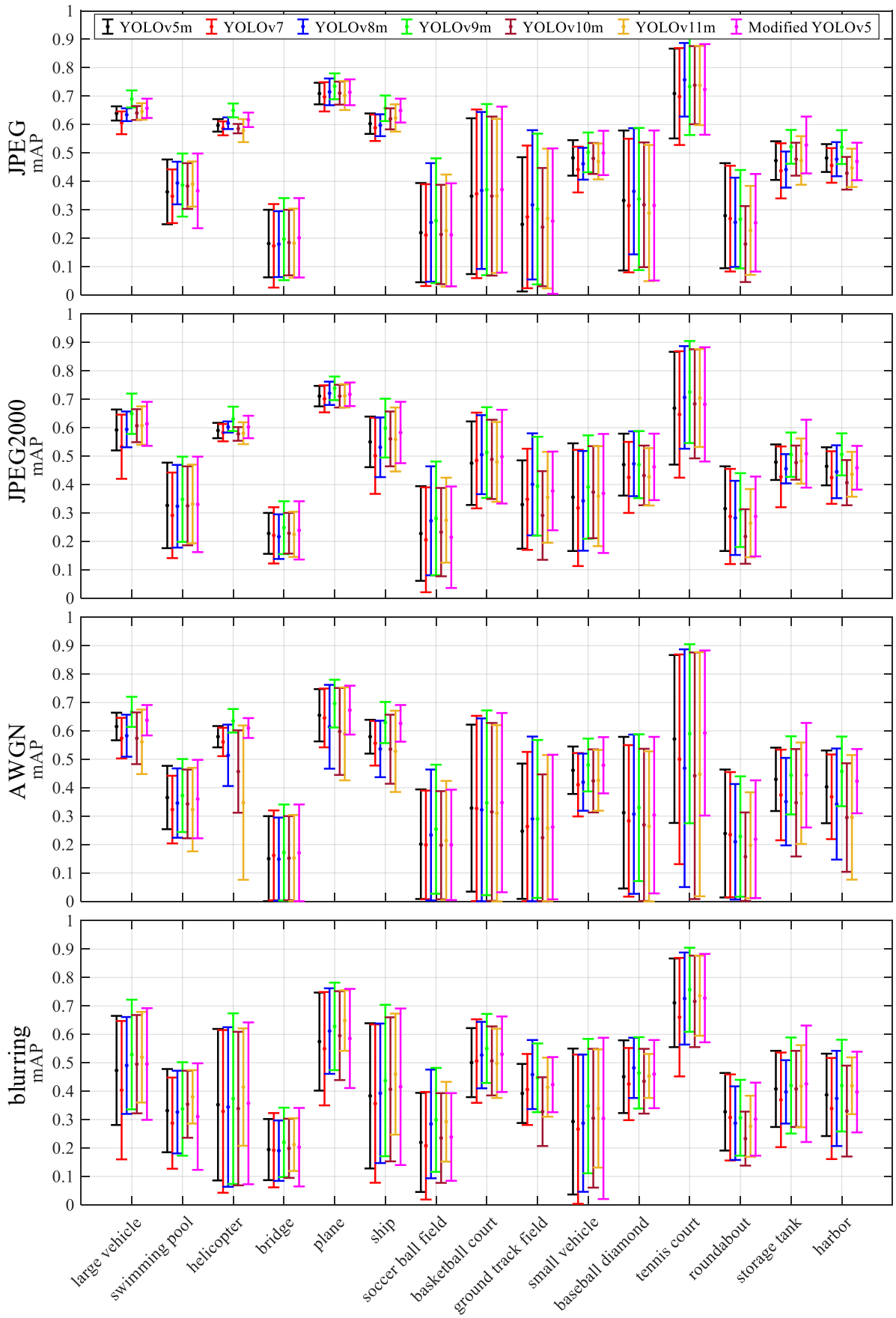



**Fig. 6.** Comparison of mAP across 10 distortion levels for DOTA-v1.0 object classes using YOLOv5m, YOLOv7, YOLOv8m, YOLOv9m, YOLOv10m, YOLOv11m, and the modified YOLOv5

### *4.4 Aggregate performance across distortions*

Table 2 summarizes the detection performance of YOLOv5m, YOLOv7, YOLOv8m, YOLOv9m, YOLOv10m, YOLOv11m, and the modified YOLOv5 in case of four distortion types. The evaluation inclu-des four key metrics: precision (P), recall (R), $mAP_{0.5}$, and mAP, providing a comprehensive comparison of each model's robustness affected by each distortion (**bold** font indicates the best results, while underlined represents the second-best results)..

**Table 2.** Detection performance metrics for YOLOv5m, YOLOv7, YOLOv8m, YOLOv9m, YOLOv10m, YOLOv11m, and the modified YOLOv5, averaged across all distortion levels for each distortion type

| Distortion type | Algorithm | P | R | $mAP_{0.5}$ | mAP |
|---|---|---|---|---|---|
| JPEG | **YOLOv5m** | **0.8275** | 0.7178 | **0.7733** | 0.5134 |
| | **YOLOv7** | 0.7951 | 0.6873 | 0.7473 | 0.4978 |
| | **YOLOv8m** | 0.8102 | 0.7045 | 0.7608 | 0.5204 |
| | **YOLOv9m** | 0.8121 | **0.7253** | 0.7723 | **0.5482** |
| | **YOLOv10m** | 0.7672 | 0.6815 | 0.7254 | 0.4968 |
| | **YOLOv11m** | 0.7898 | 0.6970 | 0.7483 | 0.5097 |
| | **modified YOLOv5** | 0.7958 | 0.7217 | 0.7698 | 0.5275 |
| JPEG2000 | **YOLOv5m** | 0.7655 | **0.6346** | 0.6913 | 0.4420 |
| | **YOLOv7** | 0.7091 | 0.5874 | 0.6316 | 0.4047 |
| | **YOLOv8m** | **0.7707** | 0.6006 | 0.6741 | 0.4461 |
| | **YOLOv9m** | 0.7654 | 0.6269 | **0.6929** | **0.4751** |
| | **YOLOv10m** | 0.7316 | 0.5815 | 0.6425 | 0.4286 |
| | **YOLOv11m** | 0.7529 | 0.6033 | 0.6711 | 0.4433 |
| | **modified YOLOv5** | 0.7352 | 0.6302 | 0.6830 | 0.4481 |
| AWGN | **YOLOv5m** | **0.8243** | 0.5610 | **0.6240** | 0.4093 |
| | **YOLOv7** | 0.7827 | 0.5252 | 0.5787 | 0.3787 |
| | **YOLOv8m** | 0.7515 | 0.5029 | 0.5607 | 0.3741 |
| | **YOLOv9m** | 0.8060 | **0.5654** | 0.6232 | **0.4367** |
| | **YOLOv10m** | 0.6901 | 0.4770 | 0.5298 | 0.3541 |
| | **YOLOv11m** | 0.6566 | 0.4643 | 0.5157 | 0.3462 |
| | **modified YOLOv5** | 0.7965 | 0.5600 | 0.6145 | 0.4184 |
| Blurring | **YOLOv5m** | **0.7562** | 0.5970 | 0.6542 | 0.4046 |
| | **YOLOv7** | 0.6963 | 0.5544 | 0.5956 | 0.3706 |
| | **YOLOv8m** | 0.7349 | 0.5662 | 0.6404 | 0.4161 |
| | **YOLOv9m** | 0.7485 | 0.6017 | 0.6667 | **0.4418** |
| | **YOLOv10m** | 0.7067 | 0.5397 | 0.6133 | 0.3960 |
| | **YOLOv11m** | 0.7524 | 0.6150 | **0.6813** | 0.4371 |
| | **modified YOLOv5** | 0.7194 | **0.6196** | 0.6660 | 0.4162 |

For JPEG compression, YOLOv9 achieves the highest mAP (0.5482) and recall (0.7253), while the modified YOLOv5 reaches the second-best result of mAP (0.5275) and recall of (0.7217). YOLOv5 provides the highest precision (0.8275) and $mAP_{0.5}$ (0.7730). In contrast, YOLOv10m demonstrates the lowest performance across all metrics.

Under JPEG2000 compression, YOLOv8 exhibits the best precision (0.7707), whereas the highest recall (0.6346) and the second-best results of precision (0.7655) and $mAP_{0.5}$ (0.6913) are reached by YOLOv5. YOLOv9m surpasses the other models in both $mAP_{0.5}$ (0.6929) and mAP (0.4751), indicating its robustness under JPEG2000 compression, while the lowest analyzed precision, $mAP_{0.5}$, and mAP are achieved by YOLOv7.

For AWGN distortion, the YOLOv5 outperforms all other models, achieving the highest precision (0.8243) and $mAP_{0.5}$ (0.6240), and the second-best result of recall (0.5610), while YOLOv9m provides the highest mAP (0.4367) and recall (0.5654) values, and reaches the second-best precision (0.8060) and $mAP_{0.5}$ (0.6232). In contrast, YOLOv11m demonstrates the lowest performance across all metrics.

In the case of blurring, YOLOv9m achieves the highest mAP (0.4418), and the second best $mAP_{0.5}$ (0.6667). while YOLOv11m attains the best $mAP_{0.5}$ (0.6813), and second-best result for the rest metrics. These results demonstrate the robustness of YOLOv9 and YOLOv11 against blurring distortion. Furthermore, YOLOv7, and YOLOv10m consistently records the lowest performance, highlighting their sensitivity to blurring distortion.

Based on this analysis, it is observed that YOLOv9m achieves a superior detection performance in terms of mAP regardless of the distortion type. It also demonstrates strong performance results for precision, recall and $mAP_{0.5}$. In addition, the modified YOLOv5 shows the second-best results in term of mAP. In contrast, YOLOv7 and YOLOv10 exhibit weak performance across all distortion types, indicating lower robustness compared to other models. The weak robustness of YOLOv7 can be explained by the deep aggregation of multi-scale features in E-ELAN that may amplify noise and compression artifacts in deeper layers. Additionally, its higher model complexity may increase sensitivity to pixel-level distortions, disrupting the consistency of learned features.

## 5 Conclusion

This study provides a comprehensive evaluation of the robustness of state-of-the-art YOLO models, including YOLOv5, YOLOv7, YOLOv8, YOLOv9, YOLOv10, YOLOv11, and a modified YOLOv5, in the presence of common image distortions in remote sensing images: AWGN, JPEG and JPEG2000 compressions, and Gaussian blurring. A total of 40 test sets (10 levels per distortion type) were generated from the DOTA-v1.0 dataset. Key findings indicate that the distortion causes the performance declining. YOLOv9 demonstrates superior robustness and resilience to distortions, outperforming others YOLO models in terms of mAP, while YOLOv7 and YOLOv10 exhibit the weakest resilience. The study also reveals that distortion resilience varies significantly across object classes. Generally, small objects, such as helicopters and ship show significant mAP reduction under blurring and lower reduction under JPEG compression. In contrast, large objects such as soccer ball field, basketball court, ground track field, baseball diamond, and roundabout exhibit a higher mAP dropping under AWGN and JPEG compression compared to blurring.

Future work should focus on improving YOLO architectures to mitigate distortion impacts, bridging the gap between controlled environments and real-world conditions. Additionally, expanding training datasets to include real-world complex remote sensing scenarios with different or combined distortions could further enhance the robustness of object detection models.

## Acknowledgments

This research has been a part of Project No. VA/TT/1/25-27 supported by the Ministry of Defence, Republic of Serbia.

## References

[1] J. Ding *et al.*, “Object detection in aerial images: A large-scale benchmark and challenges,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp. 7778–7796, 2022, doi: 10.1109/TPAMI.2021.3117983

[2] K. Li, G. Wan, G. Cheng, L. Meng, and J. Han, “Object detection in optical remote sensing images: A survey and a new benchmark,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 159, pp. 296–307, 2020, doi: 10.1016/j.isprsjprs.2019.11.023

[3] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017, doi: 10.1109/TPAMI.2016.2577031

[4] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980–2988. doi: 10.1109/ICCV.2017.322

[5] H. Bakir and R. Bakir, "Evaluating the robustness of yolo object detection algorithm in terms of detecting objects in noisy environment," *Journal of Scientific Reports-A*, no. 054, pp. 1–25, 2023, doi: 10.59313/jsr-a.1257361

[6] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2999–3007. doi: 10.1109/ICCV.2017.324

[7] G. Jocher, "YOLOv5 by ultralytics,", 2020. [Online]. Available: https://github.com/ultralytics/yolov5

[8] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 7464–7475. doi: 10.1109/CVPR52729.2023.00721

[9] G. Jocher, C. Ayush, and Q. Jing, "Ultralytics YOLOv8," 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[10] C.-Y. Wang, I.-H. Yeh, and H.-Y. Mark Liao, "YOLOv9: Learning what you want to learn using programmable gradient information," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024, pp. 1–21. doi: 10.1007/978-3-031-72751-1_1

[11] A. Wang *et al.*, "YOLOv10: Real-Time end-to-end object detection," in *Advances in Neural Information Processing Systems*, pp. 107984–108011, 2024, [Online]. Available: https://openreview.net/pdf?id=tz83Nyb711

[12] J. Glenn and Q. Jing, "Ultralytics YOLOv11,". 2024. [Online]. Available: https://github.com/ultralytics/ultralytics

[13] S. Cygert and A. Czyzewski, "Toward robust pedestrian detection with data augmentation," *IEEE Access*, vol. 8, pp. 136674–136683, 2020, doi: 10.1109/ACCESS.2020.3011356

[14] C. Michaelis *et al.*, "Benchmarking robustness in object detection: autonomous driving when winter is coming," *arXiv preprint arXiv:1907.07484*, Jul. 2019, [Online]. Available: https://arxiv.org/abs/1907.07484

[15] D. Li, J. Zhang, and K. Huang, "Universal adversarial perturbations against object detection," *Pattern Recognition*, vol. 110, art. no. 107584, 2021, doi: 10.1016/j.patcog.2020.107584

[16] S. Dodge and L. Karam, "Understanding how image quality affects deep neural networks," in *2016 Eighth International Conference on Quality of Multimedia Experience (QoMEX)*, 2016, pp. 1–6. doi: 10.1109/QoMEX.2016.7498955

[17] J. A. Rodríguez-Rodríguez, E. López-Rubio, J. A. Ángel-Ruiz, and M. A. Molina-Cabello, "The impact of noise and brightness on object detection methods," *Sensors*, vol. 24, no. 3, art. no. 821, 2024, doi: 10.3390/s24030821

[18] M. Aqqa, P. Mantini, and S. K. Shah, "Understanding how video quality affects object detection algorithms," in *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*, 2019, pp. 96–104. doi: 10.5220/0007401600960104

[19] A. Beghdadi, M. Mallem, and L. Beji, "Benchmarking performance of object detection under image distortions in an uncontrolled environment," in *IEEE International Conference on Image Processing (ICIP)*, 2022, pp. 2071–2075. doi: 10.1109/ICIP46576.2022.9897643

[20] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," *arXiv preprint*. 2020, [Online]. Available: https://arxiv.org/abs/2004.10934

[21] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10778–10787. doi: 10.1109/CVPR42600.2020.01079

[22] 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

[23] T. Gandor and J. Nalepa, "First gradually, then suddenly: understanding the impact of image compression on object detection using deep learning," *Sensors*, vol. 22, no. 3, art. no. 1104, 2022, doi: 10.3390/s22031104

[24] J. A. Rodríguez-Rodríguez, M. A. Molina-Cabello, R. Benítez-Rochel, and E. López-Rubio, "The effect of noise and brightness on convolutional deep neural networks," in *Pattern Recognition. ICPR International Workshops and Challenges*, 2021, pp. 639–654. doi: 10.1007/978-3-030-68780-9_49

[25] H. Liu, Z. Li, S. Lin, and L. Cheng, "Remote sensing image denoising based on deformable convolution and attention-guided filtering in progressive framework," *Signal, Image and Video Processing*, vol. 18, no. 11, pp. 8195–8205, 2024, doi: 10.1007/s11760-024-03461-1

[26] A. Bayerl, M. Keglevic, M. G. Wödlinger, and R. Sablatnig, "Impact of learned domain specific compression on satellite image object classification," in *Proceedings of the 26th Computer Vision Winter Workshop (CVWW)*, pp. 1–8, 2023. doi: 10.34726/5331

[27] Z. Chen, Y. Hu, and Y. Zhang, "Effects of compression on remote sensing image classification based on fractal analysis," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 7, pp. 4577–4590, 2019, doi: 10.1109/TGRS.2019.2891679

[28] C. Lee, S. Youn, J. Y. Baek, and J. Serra Sagristà, "Effects of compression on classification performance and discriminant information preservation in remotely sensed data," in *Proc. SPIE 9501, Satellite Data Compression, Communications, and Processing XI*. 2015, pp. 12–20, art. no. 950103. doi: 10.1117/12.2180223

[29] J. Wei, L. Mi, Y. Hu, J. Ling, Y. Li, and Z. Chen, "Effects of lossy compression on remote sensing image classification based on convolutional sparse coding," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022, doi: 10.1109/LGRS.2020.3047789

[30] Q. Shi *et al.*, "Research on spaceborne target detection based on YOLOv5 and image compression," *Future Internet*, vol. 15, no. 3, art. no. 114, 2023, doi: 10.3390/fi15030114

[31] G.-S. Xia *et al.*, "DOTA: A large-scale dataset for object detection in aerial images," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3974–3983. doi: 10.1109/CVPR.2018.00418

[32] L. A. Varga, S. Koch, and A. Zell, "Comprehensive analysis of the object detection pipeline on UAVs," *Remote Sensing*, vol. 14, no. 21, 2022, doi: 10.3390/rs14215508

[33] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 6568–6577. doi: 10.1109/ICCV.2019.00667

[34] R. Tsekhmystro, O. Rubel, O. Prysiazhniuk, and V. Lukin, "Impact of distortions in UAV images on quality and accuracy of object localization," *Radioelectronic and Computer Systems*, vol. 2024, no. 4, pp. 59–67, 2024, doi: 10.32620/reks.2024.4.05

[35] T. Adli, D. Bujaković, B. Bondžulić, M. Z. Laidouni, and M. Andrić, "A modified YOLOv5 architecture for aircraft detection in remote sensing images," *Journal of the Indian Society of Remote Sensing*, vol. 53, no. 3, pp. 933–948, 2025, doi: 10.1007/s12524-024-02033-7

[36] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The JPEG 2000 still image compression standard," *IEEE Signal Processing Magazine*, vol. 18, no. 5, pp. 36–58, 2001, doi: 10.1109/79.952804

**Touati Adli** received the Undergraduate Scientific degree at National Preparatory School for Engineering Studies, Algiers, Algeria in 2016 and M.Sc. degree in computer science engineering from Ecole Military Polytechnic, Algiers, Algeria in 2019. He is a doctoral student at Military Academy, University of Defence in Belgrade. His research interests include machine/deep learning and computer vision. He has published 10 papers in national and international conferences and journals.

**Dimitrije M. Bujaković** received the B.Sc. degree in electrical engineering from Military Technical Academy, Serbia in 2004, M.Sc. degree at the School of Electrical Engineering, University of Belgrade, Serbia in 2008 and Ph.D. degree from the School of Electrical Engineering, University of Belgrade, Serbia in 2016. He is an Associate Professor at the Department of Military Electrical Engineering, Military Academy, University of Defence in Belgrade. His research interests include pattern recognition and methods for signals analysis and digital signal processing. He has published more than 50 papers in national and international conferences and journals.

**Boban P. Bondžulić** received the B.Sc. degree in electrical engineering from Military Technical Academy, Serbia in 2000, M.Sc. degree at the School of Electrical Engineering, University of Belgrade, Serbia in 2005 and Ph.D. degree from Faculty of Technical Sciences, University of Novi Sad, Serbia in 2016. He is an Associate Professor at the Department of Telecommunications and Informatics, Military Academy, University of Defence in Belgrade. His research interests include information fusion, image and video quality evaluation, detection and tracking of moving objects, pattern recognition. He has published more than 130 papers in national and international conferences and journals.

**Mohammed Zouaoui Laidouni** received the Undergraduate Scientific degree at National Preparatory School for Engineering Studies, Algiers, Algeria in 2014 and M.Sc. degree in telecommunication engineering from Ecole Military Polytechnic, Algiers, Algeria in 2017. He is a doctoral student at Military Academy, University of Defence in Belgrade. His research interests image fusion and image quality assessment. He has published 9 papers in national and international conferences and journals.

**Milenko S. Andrić** received the B.Sc. degree in electrical engineering from Military Technical Academy, Serbia in 1995, M.Sc. degree at the School of Electrical Engineering, University of Belgrade, Serbia in 2001 and Ph.D. degree from the Military Academy, Serbia in 2006. He is a full-time professor at the Department of Military Electrical Engineering, Military Academy, University of Defence in Belgrade. His main research interests are in the fields of stochastically process in telecommunication and radar engineering, pattern recognition, methods for signals analysis and digital signal processing. He has published more than 110 papers in national and international conferences and journals.

Received 29 June 2025