MÜHENDİSLİK FAKÜLTESİ

Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi
Niğde Ömer Halisdemir University Journal of Engineering Sciences

Araştırma makalesi / Research article

www.dergipark.org.tr/tr/pub/ngumuh / www.dergipark.org.tr/en/pub/ngumuh

PushPull-YOLO: A biologically inspired framework for robust object detection under image corruptions

PushPull-YOLO: Görüntü bozulmaları altında güçlü nesne algılama için biyoloji esinli bir çerçeve

Hasan Ali Akyürek[1,*]

[1] *Department of Biostatistics and Medical Informatics, Faculty of Medicine, Necmettin Erbakan University, Konya, Türkiye*

Abstract

In this study, a novel integration of PushPull-Convolutional Layers into the YOLOv11 object detection model is proposed to enhance robustness against diverse image corruptions. The PushPull-Conv layer is designed based on biological mechanisms of the primary visual cortex, where complementary push and pull kernels are utilized to improve selectivity by amplifying relevant stimuli and suppressing irrelevant noise. The initial convolutional layer of YOLOv11 is replaced by this modification, and performance is evaluated on the COCO dataset across 15 corruption types (e.g., noise, blur, weather, digital artifacts) with five severity levels. Improved robustness metrics are achieved by the PushPull-enhanced YOLOv11 compared to the baseline. Detection performance under challenging conditions, including brightness variation, motion blur, and contrast changes, is enhanced. A link is established between biologically inspired design and deep learning, positioning PushPull-YOLO as a promising solution for real-time object detection in dynamic environments, with potential extensions to segmentation and keypoint detection.

Keywords: Convolutional layers, Deep learning, Image corruption, Object detection, PushPull, YOLO

Öz

Bu çalışmada, YOLOv11 nesne tespit modeline PushPull Konvolüsyon Katmanlarının özgün bir entegrasyonu önerilerek görüntü bozulmalarına karşı dayanıklılığın artırılması amaçlanmıştır. PushPull-Conv katmanı, birincil görsel korteksin biyolojik mekanizmalarından esinlenerek tasarlanmış ve tamamlayıcı push ve pull çekirdekleri kullanılarak ilgili uyaranların güçlendirilmesi ve ilgisiz gürültünün bastırılması yoluyla seçiciliğin artırılması sağlanmıştır. YOLOv11'in ilk konvolüsyon katmanı bu değişiklik ile değiştirilmiş ve performans, COCO veri kümesi üzerinde 15 farklı bozulma türü (ör. gürültü, bulanıklık, hava koşulları ve dijital bozulmalar) ve beş şiddet düzeyinde değerlendirilmiştir. PushPull ile güçlendirilmiş YOLOv11'in, temel modele kıyasla üstün dayanıklılık metrikleri elde ettiği gösterilmiştir. Parlaklık değişimi, hareket bulanıklığı ve kontrast farklılıkları gibi zorlu koşullar altında tespit performansı iyileştirilmiştir. Biyolojik esinli tasarım ile derin öğrenme arasında bir bağlantı kurulmuş ve PushPull-YOLO'nun dinamik ortamlarda gerçek zamanlı nesne tespiti için umut verici bir çözüm sunduğu ortaya konulmuştur. Ayrıca yöntemin gelecekte segmentasyon ve anahtar nokta tespiti gibi diğer bilgisayarla görme görevlerine de uygulanabileceği düşünülmektedir.

Anahtar kelimeler: Evrişimsel katmanlar, Derin öğrenme, Görüntü bozulması, Nesne algılama, PushPull, YOLO

1 Introduction

Object detection stands as a pivotal domain within modern computer vision, underpinning a diverse range of applications such as autonomous vehicles [1], surveillance systems [2], healthcare diagnostics [3, 4], and industrial automation [5]. Among the numerous inventions in this field, the YOLO series has proved to be the cornerstone innovation for object detection [6]. In fact, YOLO redefined object detection as a single-shot regression problem, which gave it unrivaled real-time performance [6, 7]. From 2015 onwards, the series has undergone continuous refinement and now includes an improved version called YOLOv11, with state-of-the-art improvements in precision and efficiency [8].

However, despite these advancements, object detection in real-world scenarios remains fraught with challenges [9]. Image corruptions, arising from noise, blurring effects, weather-induced distortions, or digital artifacts, significantly degrade model performance [10]. Addressing these adversities requires architectural innovations that bolster robustness while ensuring computational efficiency. This study integrates the PushPull convolutional methodology into YOLOv11, presenting a novel approach inspired by the selective inhibition mechanisms observed in the primary visual cortex. These PushPull-Conv units emulate the neural process of emphasizing relevant stimuli while suppressing non-preferred features, thus enhancing the model's resilience against diverse corruption types [11].

Such robust systems are essentially needed since the deployment of object detection models has been increasing in high-stake environments. For instance, autonomous

* Sorumlu yazar / Corresponding author, e-posta / e-mail: hakyurek@erbakan.edu.tr (H. A. Akyürek)
Geliş / Received: 21.03.2025 Kabul / Accepted: 24.06.2025 Yayımlanma / Published: 15.07.2025
doi: 10.28948/ngumuh.1662465

driving operates under dynamic conditions, where different light and weather conditions create different artifacts. Similarly, medical imaging should also perform well which might contain different visual or processing artifacts [1, 9, 10]. This study tries to bridge these gaps by introducing PushPull convolution into YOLOv11 and hence creates a system which is not only state-of-the-art on benchmark datasets but also highly robust against real-world perturbations [11].

This study provides a systematic investigation of the integration of PushPull and its implications on robustness metrics and general performance. Training and analysis will be done on COCO using different corruption types, from simple Gaussian noise to complex ones like JPEG compression artifacts [12]. The biologically inspired origin of the methodology will ensure enhancements that are effective yet computationally efficient, maintaining the real-time performance hallmark of YOLOv11. Insights from this research go on to underline the potential in marrying biologically inspired principles with advanced deep learning architectures toward the creation of future innovations in robust computer vision systems [13-15].

The integration of biologically inspired methodologies, like PushPull convolutions, in state-of-the-art deep learning models like YOLOv11 signifies a movement toward hybrid architectures, which will offer computational efficiency together with adaptability [16]. The PushPull mechanism, which was originally developed to enhance the robustness in convolutional networks, suppresses the effects of high-frequency and structural image corruptions through selective inhibition. This is related to principles in the mammalian visual cortex, where inhibitory mechanisms sharpen perceptual selectivity and reduce noise interference [11].

The PushPull mechanism is inspired by the selective inhibition properties observed in the primary visual cortex, where excitatory and inhibitory receptive fields jointly enhance edge detection and noise suppression. In biological vision systems, neurons in the early stages of the visual hierarchy—particularly in the V1 area—are organized to respond to specific spatial and orientation patterns through a balance of excitatory and suppressive inputs. This organization allows the system to amplify salient features, such as edges and contours, while simultaneously attenuating irrelevant background noise and artifacts. The push (excitatory) and pull (inhibitory) interactions observed in these receptive fields serve as a biologically efficient filtering strategy, enabling robust perception even under visually challenging conditions. Emulating this principle in artificial networks, PushPull convolutions seek to replicate the dual response mechanism, promoting enhanced feature selectivity and greater robustness to image corruptions such as blur, noise, and occlusion [17-20].

Real-world applications, ranging from critical medical imaging diagnostics to autonomous navigation, demand such robustness [3, 21]. These systems often encounter unpredictable image distortions, ranging from environmental factors like low-light conditions and atmospheric disturbances to synthetic alterations such as lossy compression artifacts. Traditional object detection architectures, even with advanced data augmentation techniques, struggle to sustain high performance under these challenges [22-24]. By embedding PushPull convolutions at strategic points within the YOLOv11 pipeline, this research addresses these gaps, potentially elevating both detection accuracy and model stability [11].

YOLOv11, the latest version in the YOLO family, integrates several novel components such as the C3k2 block, the spatial attention-based C2PSA module, and an improved neck-head pipeline, making it particularly efficient and accurate for real-time object detection [8, 25]. To the best of our knowledge, this study represents the first integration of biologically inspired PushPull convolutional units into the YOLOv11 architecture to enhance robustness against corrupted visual input.

Moreover, this study offers a comparative evaluation between the standard YOLOv11 architecture and its enhanced variant employing PushPull layers. This involves a meticulous analysis of performance metrics on corruption-focused datasets such as COCO-C [10], examining both accuracy and robustness across diverse scenarios. The results aim to provide actionable insights into the practicality of incorporating biologically inspired mechanisms into cutting-edge computer vision frameworks.

2 Related work

The YOLO series has consistently improved object detection capabilities through architectural enhancements. YOLOv11 introduces key features such as the C3k2 block, Spatial Pyramid Pooling-Fast (SPPF), and Cross-Stage Partial with Spatial Attention (C2PSA), delivering significant gains in accuracy and computational efficiency [6, 26]. Additionally, architectural improvements, such as the incorporation of the C3k2 block and the C2PSA attention module, further enhance the model's ability to process spatial information and focus on critical regions within images. These modules play a pivotal role in ensuring efficient and accurate object detection across various visual scenarios [27]. The architecture of YOLOv11 is shown in Figure 1.

PushPull convolution is inspired by the antiphase inhibition phenomenon in the primary visual cortex, PushPull convolution utilizes complementary push and pull kernels. These kernels enhance stimulus selectivity by amplifying preferred features while inhibiting non-preferred ones. Previous studies have demonstrated its effectiveness in ResNet architectures, particularly for tasks requiring robustness to image corruptions [11]. Robustness to image corruptions is a critical challenge in computer vision. Techniques such as data augmentation and adversarial training have shown promise, but architectural modifications, such as PushPull-Conv, provide a complementary approach by inherently improving model resilience [11, 28].

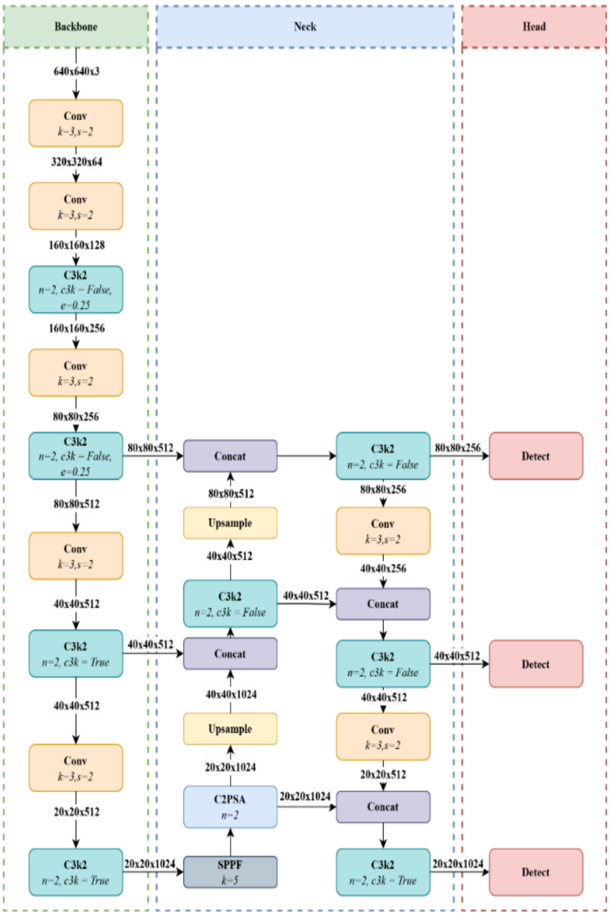

Figure 1. Architecture of YOLOv11

3 Materials and methods

3.1 Architectural modifications

The integration of PushPull convolution into YOLOv11 involves replacing the first convolutional layer with a specialized PushPull-Conv unit. This unique unit introduces complementary kernels, designed to mimic biological mechanisms observed in the primary visual cortex. Push kernel is trained to respond to specific stimuli, effectively enhancing its ability to detect preferred patterns in images. Pull kernel acts as the complementary counterpart to the Push Kernel, this kernel inhibits responses to non-preferred stimuli by producing opposing activations. Together, these kernels ensure that the network selectively amplifies meaningful signals while suppressing noise. The combination of these kernels is inspired by antiphase inhibition mechanisms, which enhance the network's robustness and improve its capacity to handle corrupted inputs without a significant computational overhead [11]. The PushPull-Conv computational unit is shown in Figure 2.

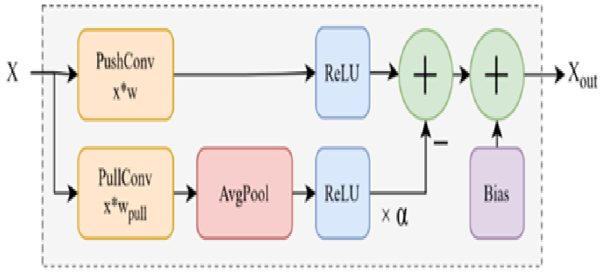

Figure 2. PushPull-Conv computational unit

By embedding this mechanism within the YOLOv11 architecture, the model achieves better feature extraction and suppression of irrelevant patterns [11, 26]. The architecture of PushPull-YOLO is shown in Figure 3.

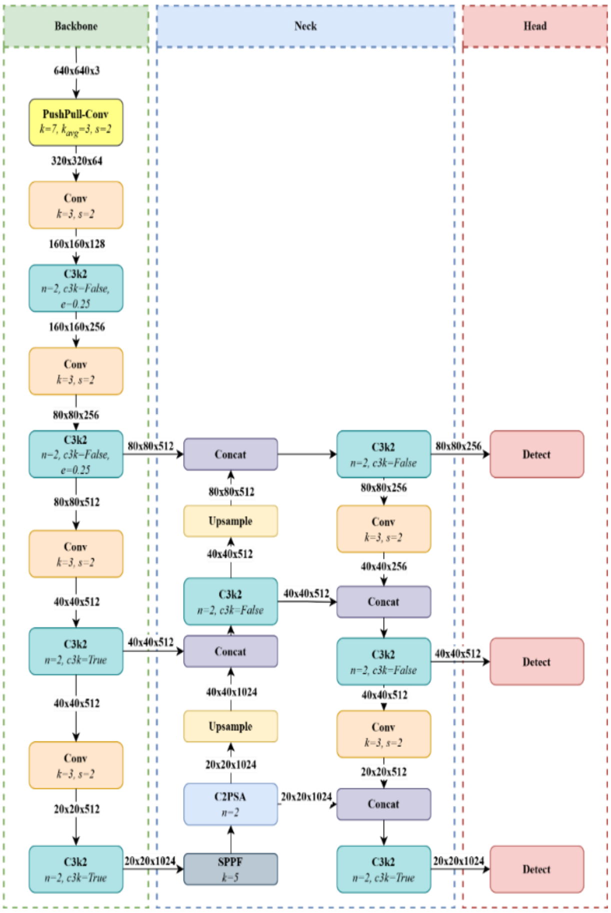

Figure 3. Architecture of PushPull-YOLO

3.2 Dataset and corruption types

To evaluate the effectiveness of the enhanced YOLOv11, the COCO dataset is employed as the primary benchmark. This widely recognized dataset enables comprehensive training and testing, covering both original images and corrupted variations. The corruptions applied are categorized into four primary types, each representing real-world image degradation scenarios [10, 12, 29, 30]:

Noise: Includes Gaussian noise, Shot noise, and Impulse noise, which introduce random pixel-level variations.

Blur: Encompasses Defocus, Glass, Motion, and Zoom blur, simulating out-of-focus and motion-induced distortions.

Weather: Consists of Snow, Frost, and Fog effects, emulating natural environmental challenges.

Digital: Includes Brightness, Contrast, Elastic transformation, Pixelation, and JPEG compression artifacts, mimicking digital processing distortions.

Each corruption type is tested across five levels of severity, providing a robust evaluation framework. This approach ensures that the model's performance is thoroughly assessed under diverse and challenging conditions, reflecting its ability to generalize effectively across varying scenarios.

3.3 Training and implementation

The training process is designed to maximize the efficiency and robustness of the enhanced YOLOv11 model. The implementation pipeline includes the following key steps:

Optimizer: Stochastic Gradient Descent (SGD) with a momentum of 0.9 and a weight decay of 1e-5 is employed to ensure stable and effective learning.

Learning Rate Schedule: A cosine annealing schedule is used, incorporating warm-up and decay phases to optimize convergence.

Training Epochs: The model is trained for 20 epochs, balancing computational efficiency and the need for thorough experimentation.

The PushPull-Conv unit is seamlessly integrated into the backbone of YOLOv11, replacing the first convolutional layer. This ensures that the selective inhibition mechanism is applied at the earliest stage of feature extraction, allowing the model to enhance relevant patterns and suppress irrelevant details from the outset. The biologically inspired design of the PushPull-Conv unit not only enhances robustness but also maintains the model's computational efficiency, making it suitable for real-time applications.

3.4 Hyperparameter selection

To systematically assess the impact of kernel size on robustness and detection accuracy in the proposed PushPull convolutional layers, we conducted a controlled ablation study using various kernel sizes, including 3×3, 5×5, 7×7, 9×9, and 11×11. All experiments were conducted under identical training conditions and limited to three epochs to isolate the influence of kernel size from other factors. The results demonstrated that smaller kernels (3×3 and 5×5) offered higher initial precision and faster convergence, while larger kernels (7×7 and above) progressively improved mAP50 and mAP50-95 scores, suggesting superior spatial generalization and resilience to corruptions.

In particular, the 7×7 PushPull kernel achieved the highest mAP50-95 value (0.0573) by the end of the third epoch, outperforming not only all other tested configurations but also the commonly used 3×3 baseline (0.0108). This finding supports the effectiveness of wider receptive fields in suppressing high-frequency corruptions and aligns with biologically inspired insights reported by Bennabhaktula et al. (2024) [11]. Additionally, when the inhibitory strength parameter (α) was implemented as a learnable parameter during training, it enabled dynamic adaptation without compromising model stability. This behavior justifies its inclusion as a default configuration in our architecture.

Overall, the choice of kernel size involves a trade-off between computational cost and robustness gains. The 7×7 configuration strikes the best balance, making it the preferred option in subsequent experiments. These findings support the functional role of wide spatial receptive fields in early visual processing and demonstrate the practical effectiveness of integrating PushPull mechanisms into modern object detection frameworks such as YOLOv11.

3.5 Evaluation metrics

To comprehensively assess the performance of the enhanced YOLOv11 model, these primary metrics are utilized. Mean Average Precision (mAP) is a metric that evaluates the detection accuracy of the model, serving as a standard benchmark in object detection tasks. It reflects the model's ability to precisely identify and localize objects across various conditions [8]. Precision (P) is a metric that measures the model's ability to correctly identify positive instances among all instances it classifies as positive. It evaluates the accuracy of the model in making positive predictions, ensuring that the detected objects are relevant and minimizing false positives. High precision indicates that most detected objects are correctly identified [8]. Recall (R) is a metric that assesses the model's effectiveness in identifying all relevant objects in the dataset. It calculates the ratio of correctly detected objects to the total number of actual objects, reflecting the model's ability to minimize false negatives. A high recall score indicates that the model successfully detects most objects in an image [8]. F1-Score is the harmonic mean of Precision and Recall, balancing the trade-off between these two metrics. It provides a single measure of a model's accuracy, particularly useful in scenarios where both false positives and false negatives need to be considered. A high F1-Score indicates a strong balance between identifying relevant objects and minimizing irrelevant detections [2]. Evaluation metrics formulas are shown in Table 1.

Table 1. Evaluation metrics and their formulas

Metric	Formula[1]
mAP50	$\frac{1}{C}\sum_{i=1}^{C} AP_i$
mAP50-95	$\frac{1}{C}\sum_{i=1}^{C}\left(\frac{1}{n}\sum_{IoUt=0.5}^{0.95} AP(IoUt)\right)$
Precision	$\frac{TP}{TP+FP}$
Recall	$\frac{TP}{TP+FN}$
F1-Score	$2\frac{Precision * Recall}{Precision + Recall}$

[1]C: Num. of classes AP_i: Avg. Precision of class i IoU_t: Intersection over Union threshold, TP: True Positive TN: True Negative FP: False Positive FN: False Negative

3.6 Analysis

The robustness of the PushPull-Conv-enhanced YOLOv11 model is analyzed in detail, focusing on its performance across different corruption frequencies:

High-Frequency corruption includes noise-related distortions that typically challenge models due to their random and unpredictable nature. Mid-Frequency corruption encompasses blur effects that test the model's ability to handle spatial distortions. Low-Frequency corruption covers weather-related effects that simulate gradual, environmental image degradation.

The results demonstrate that the enhanced model excels particularly in handling high-frequency corruptions, showcasing its advanced feature extraction capabilities. The PushPull-Conv unit's selective inhibition mechanism plays a critical role in this performance improvement, enabling the model to maintain high accuracy even under challenging conditions.

Additionally, Wilcoxon signed-rank test is a nonparametric statistical method designed to compare paired samples, particularly when assumptions of normality are not met. Unlike parametric tests like the paired t-test, it does not rely on the normal distribution of data, making it ideal for small sample sizes or data with unknown distributions [31]. In the context of object detection algorithms, this test is instrumental for assessing the differences in performance metrics—such as precision, recall, and mAP (mean Average Precision)—of different models applied to the same dataset. For example, when comparing improvements brought by a novel algorithm (e.g., YOLOv11) against a baseline model, the Wilcoxon signed-rank test evaluates whether observed differences in detection accuracy or inference time are statistically significant. The Wilcoxon signed-rank test ensures robust and reliable comparisons, making it a valuable tool in advancing object detection research and validating algorithmic enhancements [32].

4 Results

4.1 Brightness corruption

As the severity of brightness corruption increases, both PushPull-YOLO and YOLOv11 methods experience a gradual decline in performance. PushPull-YOLO shows competitive metrics for precision and recall while slightly outperforming YOLOv11 in mAP50-95 values, particularly at higher severity levels. This indicates its robustness in adapting to changes in brightness intensity. Table 2 highlights these trends, showcasing PushPull-YOLO's superior adaptability to brightness variations.

The Wilcoxon signed-rank test was conducted to compare the performance of PushPull-YOLO and YOLOv11 across five metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The results indicated no statistically significant difference for Precision ($p = 0.21875$), suggesting comparable performance between the two models in this metric. Recall showed borderline significance ($p = 0.09375$), hinting at a potential but inconclusive advantage for PushPull-YOLO. However, statistically significant differences were observed in favor of PushPull-YOLO for mAP50, mAP50-95, and Fitness ($p = 0.03125$ for all), demonstrating its superior performance in accuracy and robustness across varying IoU thresholds. These findings highlight that while both models perform similarly in Precision, PushPull-YOLO consistently outperforms YOLOv11 in metrics that are critical for reliable and accurate object detection. This underscores PushPull-YOLO as a more robust and effective solution for object detection tasks requiring high accuracy and overall model efficiency. Comparison of YOLOv11 and PushPull-YOLO performance metrics under brightness corruption across varying severity levels is shown in Figure 4.

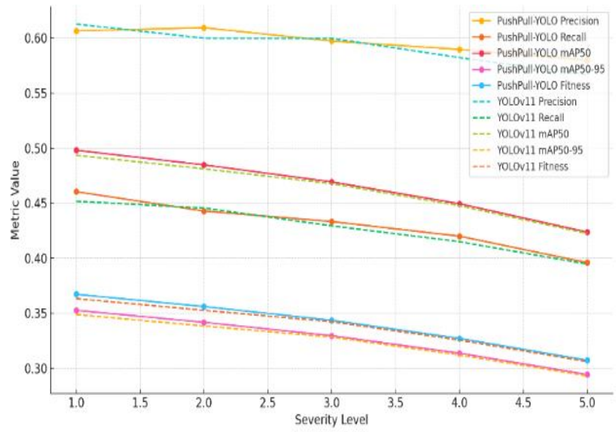

Figure 1. Performance comparison of YOLOv11 and PushPull-YOLO under brightness corruption

4.2 Contrast corruption

Contrast corruption significantly affects the metrics for both methods as severity rises. While precision and recall are similar, PushPull-YOLO achieves better mAP50-95 scores across all severity levels, demonstrating its improved handling of contrast variations. The results, detailed in Table 3, suggest that PushPull-YOLO is better suited for environments with varying contrast conditions.

The Wilcoxon signed-rank test was applied to compare the performance of PushPull-YOLO and YOLOv11 across five metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The results revealed borderline significance for Precision ($p = 0.09375$) and Recall ($p = 0.0625$), indicating potential but inconclusive differences between the two methods in these areas. However, statistically significant differences were observed in favor of PushPull-YOLO for mAP50, mAP50-95, and Fitness ($p = 0.03125$ for all), demonstrating its superior performance in detection accuracy and robustness across varying IoU thresholds. These findings highlight that while both methods achieve comparable results in Precision and Recall, PushPull-YOLO consistently outperforms YOLOv11 in metrics that are critical for reliable and accurate object detection. This underscores PushPull-YOLO's effectiveness as a robust and efficient solution for tasks requiring high detection accuracy and overall model performance. Comparison of YOLOv11 and PushPull-YOLO performance metrics under contrast corruption across varying severity levels is shown in Figure 5.

Table 2. Performance metrics for PushPull-YOLO and YOLOv11 under brightness corruption at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	**Precision**	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	
0.6065	0.4604	0.4980	0.3526	0.3671	0.6127	0.4517	0.4935	0.3487	0.3632	1
0.6093	0.4428	0.4847	0.3417	0.3560	0.5999	0.4455	0.4811	0.3383	0.3526	2
0.5972	0.4332	0.4693	0.3295	0.3435	0.5997	0.4293	0.4677	0.3283	0.3422	3
0.5897	0.4199	0.4494	0.3136	0.3270	0.5822	0.4149	0.4476	0.3119	0.3255	4
0.5798	0.3959	0.4236	0.2942	0.3072	0.5666	0.3946	0.4224	0.2930	0.3060	5
0.5965	0.4304	0.4650	0.3263	0.3402	0.5922	0.4272	0.4625	0.3240	0.3379	Avg.

Table 3. Performance metrics for PushPull-YOLO and YOLOv11 under contrast corruption at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	**Precision**	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	
0.5936	0.4209	0.4548	0.3206	0.3340	0.5919	0.4149	0.4501	0.3159	0.3293	1
0.5807	0.3900	0.4237	0.2973	0.3099	0.5555	0.3918	0.4164	0.2910	0.3036	2
0.5296	0.3446	0.3647	0.2524	0.2636	0.5132	0.3368	0.3555	0.2458	0.2567	3
0.4325	0.2173	0.2194	0.1479	0.1551	0.4023	0.2107	0.2082	0.1399	0.1467	4
0.3228	0.0839	0.0748	0.0475	0.0502	0.3280	0.0726	0.0679	0.0437	0.0461	5
0.4918	0.2914	0.3075	0.2131	0.2226	0.4782	0.2854	0.2996	0.2072	0.2165	Avg.

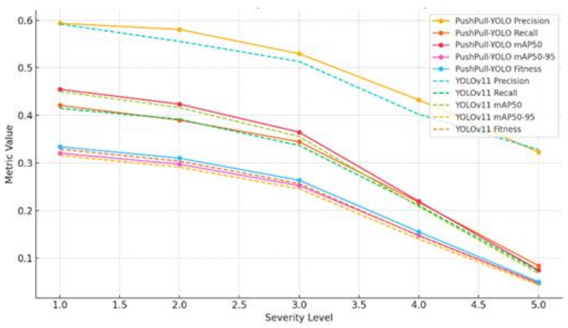

Figure 2. Performance comparison of YOLOv11 and PushPull-YOLO under contrast corruption

4.3 Defocus blur corruption

Defocus blur impacts object detection performance for both methods, with PushPull-YOLO maintaining a slight advantage in mAP metrics across all levels. The resilience of PushPull-YOLO to blur effects is particularly evident at higher severity levels, as shown in Table 4, emphasizing its suitability for detecting blurred objects.

The Wilcoxon signed-rank test was applied to compare the performance of PushPull-YOLO and YOLOv11 across five metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The analysis revealed no statistically significant differences for Recall ($p = 1.0$), mAP50 ($p = 0.3125$), mAP50-95 ($p = 0.3125$), and Fitness ($p = 0.3125$), indicating that both models perform comparably in these areas. Precision showed borderline significance ($p = 0.09375$), suggesting a potential but inconclusive advantage for PushPull-YOLO. These findings highlight that while the two methods are generally comparable across most metrics, further analysis with larger datasets or more specific conditions may be needed to clarify potential differences in Precision. Overall, PushPull-YOLO and YOLOv11 demonstrate similar capabilities in object detection tasks under the tested conditions. Comparison of YOLOv11 and PushPull-YOLO performance metrics under defocus blur corruption across varying severity levels is shown in Figure 6.

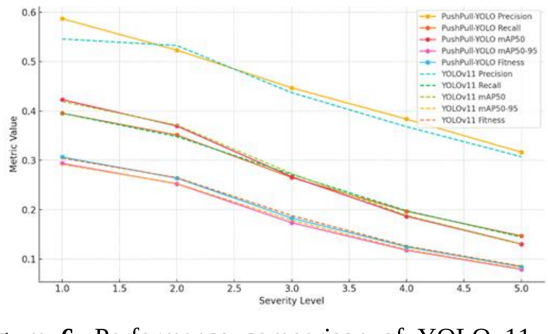

Figure 6. Performance comparison of YOLOv11 and PushPull-YOLO under defocus blur corruption

Table 4. Performance metrics for PushPull-YOLO and YOLOv11 under defocus blur corruption at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	**Precision**	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	
0.5871	0.3955	0.4231	0.2938	0.3068	0.5460	0.3954	0.4192	0.2920	0.3047	1
0.5233	0.3513	0.3696	0.2525	0.2641	0.5326	0.3481	0.3714	0.2530	0.2649	2
0.4469	0.2653	0.2668	0.1736	0.1829	0.4371	0.2715	0.2732	0.1785	0.1879	3
0.3837	0.1968	0.1868	0.1179	0.1248	0.3682	0.1979	0.1885	0.1194	0.1263	4
0.3166	0.1468	0.1304	0.0792	0.0842	0.3073	0.1447	0.1309	0.0805	0.0856	5
0.4515	0.2711	0.2753	0.1834	0.1926	0.4383	0.2715	0.2766	0.1847	0.1939	Avg.

Table 5. Performance metrics for PushPull-YOLO and YOLOv11 under elastic transform corruption at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	**Precision**	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	
0.5733	0.4161	0.4481	0.3073	0.3214	0.5562	0.4214	0.4447	0.3069	0.3207	1
0.5554	0.3767	0.4035	0.2726	0.2857	0.5432	0.3784	0.3993	0.2706	0.2834	2
0.5244	0.3217	0.3406	0.2249	0.2364	0.5387	0.3191	0.3365	0.2225	0.2339	3
0.5038	0.2845	0.2954	0.1925	0.2028	0.5150	0.2782	0.2885	0.1873	0.1974	4
0.4907	0.2350	0.2460	0.1561	0.1651	0.4937	0.2286	0.2354	0.1494	0.1580	5
0.5295	0.3268	0.3467	0.2307	0.2423	0.5293	0.3251	0.3409	0.2273	0.2387	Avg.

4.4 Elastic transform corruption

Elastic transformations challenge the models with distortions, but PushPull-YOLO demonstrates higher robustness in mAP50-95 and fitness scores. Its ability to retain detection quality under severe transformations is evident in Table 5, where PushPull-YOLO consistently performs better than YOLOv11.

The Wilcoxon signed-rank test was conducted to compare the performance of PushPull-YOLO and YOLOv11 across five metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The results revealed no statistically significant differences for Precision ($p = 1.0$) and Recall ($p = 0.4375$), indicating that both methods perform comparably in these metrics. However, statistically significant differences were observed in favor of PushPull-YOLO for mAP50, mAP50-95, and Fitness ($p = 0.03125$ for all), demonstrating its superior performance in these critical metrics. These findings suggest that while both methods achieve similar Precision and Recall, PushPull-YOLO offers significant advantages in detection accuracy and robustness, making it a more effective solution for object detection tasks requiring high accuracy and model efficiency across varying IoU thresholds. Comparison of YOLOv11 and PushPull-YOLO performance metrics under elastic transform corruption across varying severity levels is shown in Figure 7.

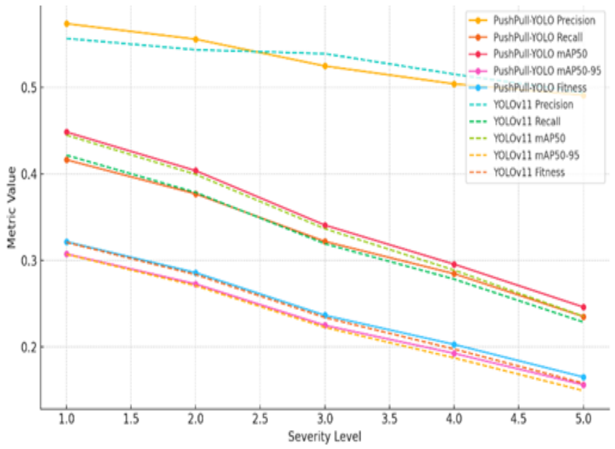

Figure 7. Performance comparison of YOLOv11 and PushPull-YOLO under elastic transform corruption

4.5. Fog corruption

Under foggy conditions, both methods show strong performance at lower severity levels, but PushPull-YOLO sustains its advantage in mAP50-95 as severity increases. This robustness makes it ideal for low-visibility scenarios, as reflected in Table 6, which details the metrics across all fog severities.

The Wilcoxon signed-rank test was conducted to compare the performance of PushPull-YOLO and YOLOv11 across five metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The analysis revealed a statistically significant difference in Precision ($p = 0.03125$), indicating that PushPull-YOLO outperforms YOLOv11 in this metric. However, no statistically significant differences were observed for Recall ($p = 0.5625$), mAP50 ($p = 0.84375$), mAP50-95 ($p = 0.5625$), and Fitness ($p = 0.6875$), suggesting that the two methods perform comparably in these areas. These findings highlight that while PushPull-YOLO demonstrates a distinct advantage in achieving higher Precision, both models exhibit similar performance in other critical metrics related to detection accuracy and robustness. This suggests that PushPull-YOLO may be better suited for tasks where Precision is of greater importance, while both methods remain reliable and effective in broader object detection scenarios. Comparison of YOLOv11 and PushPull-YOLO performance metrics under fog corruption across varying severity levels is shown in Figure 8.

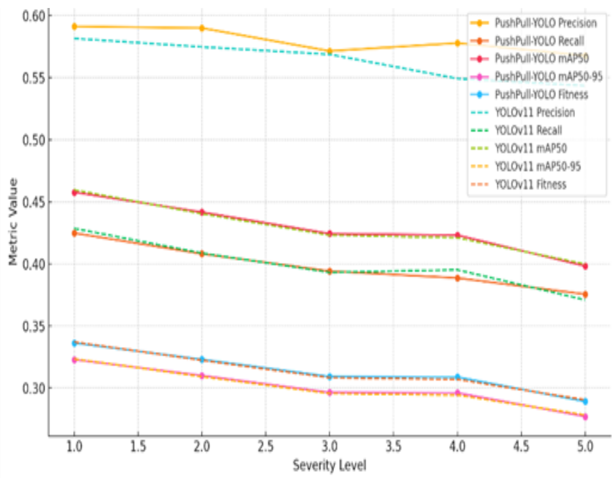

Figure 8. Performance comparison of YOLOv11 and PushPull-YOLO under fog corruption

Table 6. Performance metrics for PushPull-YOLO and YOLOv11 methods under fog corruption at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	**Precision**	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	
0.5912	0.4248	0.4577	0.3229	0.3363	0.5815	0.4285	0.4596	0.3234	0.3370	1
0.5898	0.4081	0.4416	0.3101	0.3230	0.5747	0.4087	0.4403	0.3092	0.3223	2
0.5713	0.3941	0.4243	0.2964	0.3092	0.5687	0.3931	0.4230	0.2956	0.3083	3
0.5778	0.3886	0.4231	0.2962	0.3089	0.5492	0.3952	0.4210	0.2943	0.3070	4
0.5677	0.3756	0.3982	0.2771	0.2892	0.5434	0.3710	0.3998	0.2783	0.2904	5
0.5796	0.3982	0.4290	0.3005	0.3133	0.5635	0.3993	0.4287	0.3002	0.3130	Avg.

Table 7. Performance metrics for PushPull-YOLO and YOLOv11 methods under frost corruption at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	**Precision**	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	
0.5736	0.3993	0.4293	0.2971	0.3102	0.5726	0.3916	0.4227	0.2921	0.3052	1
0.5387	0.3318	0.3544	0.2401	0.2515	0.5265	0.3187	0.3378	0.2306	0.2413	2
0.4982	0.2826	0.2955	0.1979	0.2076	0.4912	0.2671	0.2800	0.1881	0.1973	3
0.4983	0.2757	0.2849	0.1907	0.2002	0.4791	0.2618	0.2700	0.1814	0.1903	4
0.4640	0.2453	0.2542	0.1689	0.1775	0.4581	0.2310	0.2362	0.1573	0.1652	5
0.5146	0.3069	0.3237	0.2190	0.2294	0.5055	0.2940	0.3093	0.2099	0.2198	Avg.

4.6 Frost corruption

Frost corruption introduces visual disturbances that affect object detection. PushPull-YOLO consistently outperforms YOLOv11 in mAP50-95 and fitness scores, especially at higher severity levels. This indicates its superior adaptability to frosty conditions. Table 7 highlights these trends, demonstrating PushPull-YOLO's capability to maintain robust detection performance under frost-related corruptions.

The Wilcoxon signed-rank test was applied to evaluate the performance differences between PushPull-YOLO and YOLOv11 across five critical metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The analysis revealed statistically significant differences in favor of PushPull-YOLO for all metrics ($p = 0.03125$ for each), indicating that PushPull-YOLO consistently outperforms YOLOv11. These results highlight the enhanced detection accuracy, robustness across varying IoU thresholds, and overall model fitness offered by PushPull-YOLO. This demonstrates its superiority as a more efficient and reliable object detection solution, particularly in scenarios requiring high precision, comprehensive recall, and robust performance across diverse conditions. The findings affirm PushPull-YOLO's effectiveness in advancing object detection capabilities over the baseline performance of YOLOv11. Comparison of YOLOv11 and PushPull-YOLO performance metrics under frost corruption across varying severity levels is shown in Figure 9.

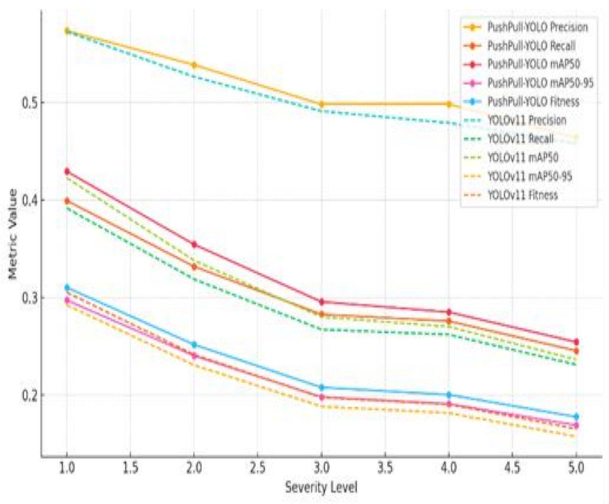

Figure 9. Performance comparison of YOLOv11 and PushPull-YOLO under frost corruption

4.7 Gaussian noise corruption

Gaussian noise leads to substantial performance degradation at higher severities. PushPull-YOLO achieves better mAP50-95 scores across all levels, indicating its effectiveness in managing noisy environments. Table 8 presents the comparative analysis, showcasing its robustness to this corruption type.

The Wilcoxon signed-rank test was applied to compare the performance of PushPull-YOLO and YOLOv11 across five metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The analysis revealed no statistically significant differences for any of the metrics, with p-values of \(p = 0.84375 for Precision, \(p = 0.21875 for Recall, mAP50-95, and Fitness, and \(p = 0.4375 for mAP50. These findings indicate that both methods exhibit comparable performance across all evaluated metrics under the given conditions. This suggests that PushPull-YOLO and YOLOv11 are equally effective for object detection tasks, offering similar levels of accuracy, robustness, and overall efficiency in detection performance. Comparison of YOLOv11 and PushPull-YOLO performance metrics under gaussian noise corruption across varying severity levels is shown in Figure 10.

Table 8. Performance metrics for PushPull-YOLO and YOLOv11 under Gaussian noise corruption at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	**Precision**	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	
0.5440	0.3582	0.3852	0.2653	0.2773	0.5442	0.3618	0.3853	0.2633	0.2755	1
0.4874	0.2789	0.2937	0.1959	0.2056	0.4764	0.2852	0.2953	0.1954	0.2054	2
0.3827	0.1747	0.1685	0.1083	0.1143	0.3954	0.1773	0.1690	0.1076	0.1137	3
0.2880	0.0793	0.0683	0.0427	0.0453	0.2945	0.0732	0.0655	0.0412	0.0437	4
0.0328	0.0664	0.0150	0.0096	0.0102	0.0153	0.0800	0.0167	0.0108	0.0114	5
0.3470	0.1915	0.1861	0.1243	0.1305	0.3452	0.1955	0.1864	0.1237	0.1299	Avg.

Table 9. Performance metrics for PushPull-YOLO and YOLOv11 methods under glass blur corruption at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	**Precision**	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	
0.5777	0.3914	0.4183	0.2875	0.3006	0.5535	0.3788	0.4028	0.2767	0.2893	1
0.5179	0.3357	0.3460	0.2362	0.2471	0.4976	0.3119	0.3212	0.2192	0.2294	2
0.3565	0.1595	0.1432	0.0917	0.0968	0.3691	0.1457	0.1271	0.0809	0.0855	3
0.3231	0.1246	0.1034	0.0652	0.0690	0.3119	0.1146	0.0922	0.0587	0.0621	4
0.2966	0.0906	0.0696	0.0422	0.0449	0.2711	0.0905	0.0674	0.0416	0.0442	5
0.4144	0.2204	0.2161	0.1445	0.1517	0.4006	0.2083	0.2021	0.1354	0.1421	Avg.

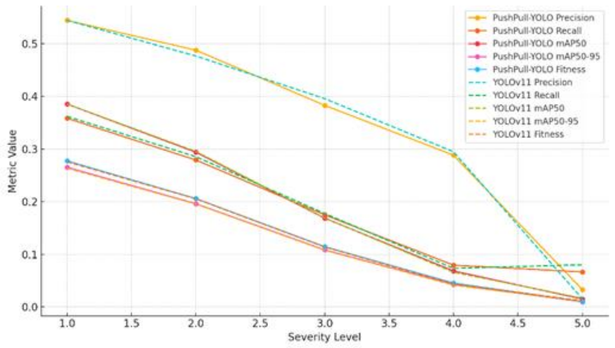

Figure 10. Performance comparison of YOLOv11 and PushPull-YOLO under Gaussian noise corruption

4.8 Glass blur corruption

Glass blur significantly impacts detection capabilities. PushPull-YOLO demonstrates stronger resilience, particularly in mAP metrics, as severity rises. Table 9 illustrates these trends, highlighting PushPull-YOLO's superiority in handling glass-induced distortions.

The Wilcoxon signed-rank test was conducted to compare the performance of PushPull-YOLO and YOLOv11 across five metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The results revealed statistically significant differences in favor of PushPull-YOLO for Recall, mAP50, mAP50-95, and Fitness ($p = 0.03125$ for all), indicating that PushPull-YOLO demonstrates superior detection accuracy, robustness across varying IoU thresholds, and overall model efficiency. Precision showed borderline significance ($p = 0.09375$), suggesting a potential but inconclusive advantage for PushPull-YOLO in this metric. These findings highlight that while both models achieve comparable results in Precision, PushPull-YOLO consistently outperforms YOLOv11 in critical metrics that reflect detection reliability and robustness, making it a more effective and reliable choice for object detection tasks under diverse conditions.

Comparison of YOLOv11 and PushPull-YOLO performance metrics under glass blur corruption across varying severity levels is shown in Figure 11.

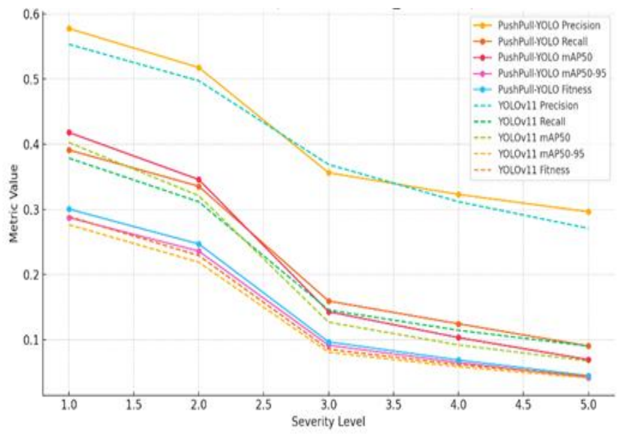

Figure 11. Performance comparison of YOLOv11 and PushPull-YOLO under glass blur corruption

4.9 Impulse noise corruption

Impulse noise introduces high-intensity pixel-level disturbances that degrade performance. PushPull-YOLO shows greater stability in mAP50-95 values, particularly at higher severities, as highlighted in Table 10, making it suitable for challenging environments.

The Wilcoxon signed-rank test was conducted to evaluate the performance differences between PushPull-YOLO and YOLOv11 across five key metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The analysis revealed statistically significant differences in favor of PushPull-YOLO for mAP50, mAP50-95, and Fitness ($p = 0.03125$ for all), demonstrating its superior detection accuracy, robustness across varying IoU thresholds, and overall model efficiency. Precision showed borderline significance ($p = 0.09375$), indicating a potential but inconclusive advantage for PushPull-YOLO in this metric. For Recall, no statistically significant difference was observed (p = 0.5625), suggesting comparable performance between the two models. These results underscore PushPull-YOLO's advantage in delivering higher accuracy and reliability, particularly in metrics that evaluate comprehensive detection capabilities, making it a more effective choice for object detection tasks where precision and robustness are critical. Comparison of YOLOv11 and PushPull-YOLO performance metrics under impulse noise corruption across varying severity levels is shown in Figure 12.

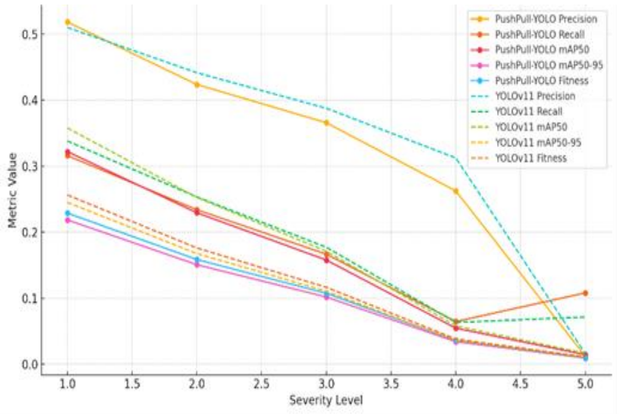

Figure 12. Performance comparison of YOLOv11 and PushPull-YOLO under impulse noise corruption

4.10 JPEG compression corruption

JPEG compression results in a gradual decline in performance as compression increases. PushPull-YOLO consistently outperforms YOLOv11 in mAP metrics, as shown in Table 11, emphasizing its robustness in handling compressed images.

The Wilcoxon signed-rank test was applied to compare the performance of PushPull-YOLO and YOLOv11 across five metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The analysis revealed a statistically significant difference in Precision (p = 0.03125), indicating that PushPull-YOLO outperforms YOLOv11 in this metric. However, no statistically significant differences were observed for Recall (p = 0.15625), mAP50 (p = 0.3125), mAP50-95 (p = 0.4375), and Fitness (p = 0.4375), suggesting comparable performance between the two methods in these areas. These findings highlight that while PushPull-YOLO demonstrates a clear advantage in Precision, both methods perform similarly in terms of overall detection accuracy, robustness across IoU thresholds, and model fitness, making them equally reliable for general object detection tasks. Comparison of YOLOv11 and PushPull-YOLO performance metrics under JPEG compression corruption across varying severity levels is shown in Figure 13.

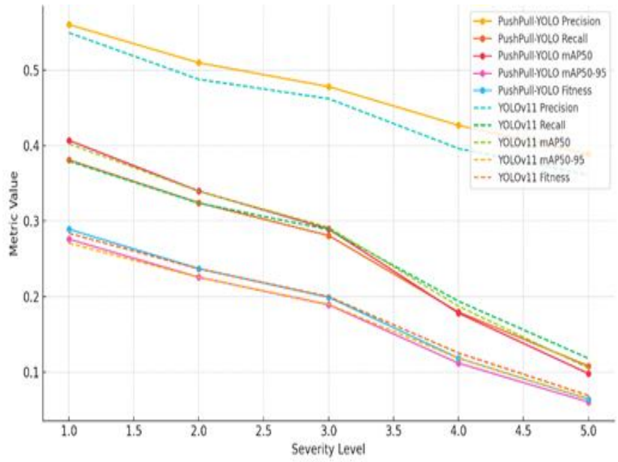

Figure 13. Performance comparison of YOLOv11 and PushPull-YOLO under jpeg compression corruption

Table 10. Performance metrics for PushPull-YOLO and YOLOv11 under impulse noise corruption at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	**Precision**	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	
0.5185	0.3161	0.3221	0.2186	0.2289	0.5103	0.3382	0.3579	0.2451	0.2563	1
0.4238	0.2340	0.2294	0.1507	0.1586	0.4416	0.2534	0.2527	0.1676	0.1761	2
0.3659	0.1667	0.1578	0.1017	0.1073	0.3878	0.1771	0.1710	0.1104	0.1165	3
0.2626	0.0645	0.0545	0.0340	0.0361	0.3125	0.0632	0.0580	0.0359	0.0381	4
0.0121	0.1080	0.0143	0.0091	0.0096	0.0161	0.0713	0.0163	0.0103	0.0109	5
0.3166	0.1778	0.1556	0.1028	0.1081	0.3337	0.1807	0.1712	0.1138	0.1196	Avg.

Table 11. Performance metrics for PushPull-YOLO and YOLOv11 under JPEG compression at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	**Precision**	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	
0.5599	0.3806	0.4066	0.2758	0.2889	0.5491	0.3792	0.4021	0.2704	0.2836	1
0.5094	0.3239	0.3395	0.2253	0.2367	0.4874	0.3235	0.3391	0.2251	0.2365	2
0.4778	0.2806	0.2893	0.1889	0.1989	0.4618	0.2884	0.2915	0.1894	0.1996	3
0.4265	0.1793	0.1778	0.1116	0.1182	0.3958	0.1936	0.1870	0.1181	0.1250	4
0.3885	0.1077	0.0978	0.0601	0.0639	0.3606	0.1179	0.1050	0.0646	0.0686	5
0.4724	0.2544	0.2622	0.1723	0.1813	0.4510	0.2605	0.2649	0.1735	0.1827	Avg.

4.11 Motion blur corruption

Motion blur impacts both methods similarly, but PushPull-YOLO maintains a consistent advantage in mAP50-95 and fitness metrics. This resilience to motion-induced distortions is detailed in Table 12, underscoring its applicability in dynamic environments.

The Wilcoxon signed-rank test was conducted to evaluate the performance differences between PushPull-YOLO and YOLOv11 across five metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The analysis revealed statistically significant differences in favor of PushPull-YOLO for all metrics ($p = 0.03125$), demonstrating its consistent superiority over YOLOv11. These results highlight the enhanced detection accuracy, robustness across varying IoU thresholds, and overall model fitness offered by PushPull-YOLO. The statistically significant improvements in Precision and Recall further emphasize PushPull-YOLO's ability to balance accurate detection and comprehensive coverage, making it a more effective and reliable approach for object detection tasks in diverse and challenging conditions. These findings position PushPull-YOLO as a valuable advancement in object detection technology. Comparison of YOLOv11 and PushPull-YOLO performance metrics under motion blur corruption across varying severity levels is shown in Figure 14.

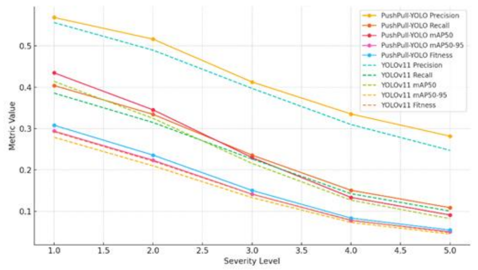

Figure 14. Performance comparison of YOLOv11 and PushPull-YOLO under motion blur corruption

4.12 Pixelate corruption

Pixelation reduces resolution and impacts detection metrics. PushPull-YOLO shows superior mAP50-95 scores across all severity levels, confirming its adaptability to low-resolution images. Table 13 summarizes these findings.

The Wilcoxon signed-rank test was conducted to compare the performance of PushPull-YOLO and YOLOv11 across five metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The results revealed no statistically significant difference in Precision ($p = 0.5625$), indicating that both methods perform comparably in terms of precision. However, statistically significant differences were observed in favor of PushPull-YOLO for Recall, mAP50, mAP50-95, and Fitness ($p = 0.03125$ for all), demonstrating its superior performance in detection accuracy, robustness, and overall model efficiency across varying IoU thresholds. These findings highlight PushPull-YOLO's ability to achieve higher recall and enhanced detection performance, making it a more robust and effective solution for object detection tasks under diverse conditions. Comparison of YOLOv11 and PushPull-YOLO performance metrics under pixelate corruption across varying severity levels is shown in Figure 15.

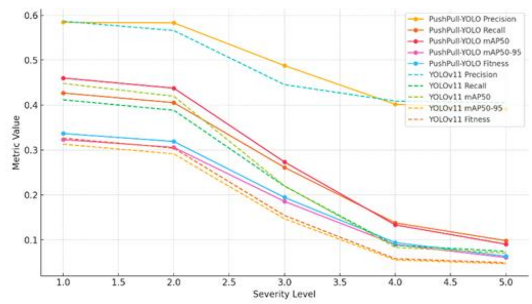

Figure 15. Performance comparison of YOLOv11 and PushPull-YOLO under pixelate corruption

Table 12. Performance metrics for PushPull-YOLO and YOLOv11 under motion blur corruption at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	**Precision**	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	
0.5685	0.4041	0.4346	0.2939	0.3080	0.5562	0.3857	0.4140	0.2789	0.2924	1
0.5161	0.3344	0.3453	0.2237	0.2358	0.4895	0.3147	0.3246	0.2100	0.2214	2
0.4126	0.2357	0.2299	0.1416	0.1504	0.3971	0.2265	0.2159	0.1331	0.1414	3
0.3352	0.1508	0.1336	0.0779	0.0835	0.3098	0.1426	0.1270	0.0733	0.0787	4
0.2816	0.1088	0.0911	0.0507	0.0547	0.2474	0.1006	0.0824	0.0452	0.0489	5
0.4228	0.2467	0.2469	0.1576	0.1665	0.4000	0.2340	0.2328	0.1481	0.1566	Avg.

Table 13. Performance metrics for PushPull-YOLO and YOLOv11 methods under pixelate corruption at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	**Precision**	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	
0.5842	0.4268	0.4601	0.3230	0.3367	0.5866	0.4119	0.4481	0.3129	0.3264	1
0.5830	0.4054	0.4377	0.3059	0.3191	0.5658	0.3885	0.4196	0.2914	0.3042	2
0.4879	0.2613	0.2734	0.1857	0.1945	0.4454	0.2197	0.2204	0.1470	0.1543	3
0.4019	0.1379	0.1335	0.0897	0.0941	0.4089	0.0880	0.0834	0.0554	0.0582	4
0.3914	0.0983	0.0906	0.0608	0.0638	0.4054	0.0746	0.0708	0.0469	0.0493	5
0.4897	0.2659	0.2791	0.1931	0.2017	0.4824	0.2365	0.2485	0.1707	0.1785	Avg.

4.13 Shot noise corruption

Shot noise introduces random noise spikes, challenging detection performance. PushPull-YOLO demonstrates better mAP50-95 metrics, especially at moderate and severe levels, as detailed in Table 14, highlighting its robustness against this corruption type.

The Wilcoxon signed-rank test was conducted to evaluate the performance differences between PushPull-YOLO and YOLOv11 across five metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The analysis revealed a statistically significant difference in favor of PushPull-YOLO for mAP50 ($p = 0.03125$), highlighting its superior detection accuracy. Precision ($p = 0.4375$) and Recall ($p = 0.21875$) showed no statistically significant differences, indicating comparable performance between the two methods in these metrics. Borderline significance was observed for mAP50-95 and Fitness ($p = 0.0625$ for both), suggesting a potential but inconclusive advantage for PushPull-YOLO in these areas. Overall, the findings demonstrate that while PushPull-YOLO provides enhanced detection accuracy through improved mAP50, its performance in other metrics is similar to that of YOLOv11, with indications of possible further advantages in robustness and model efficiency. Comparison of YOLOv11 and PushPull-YOLO performance metrics under shot noise corruption across varying severity levels is shown in Figure 16.

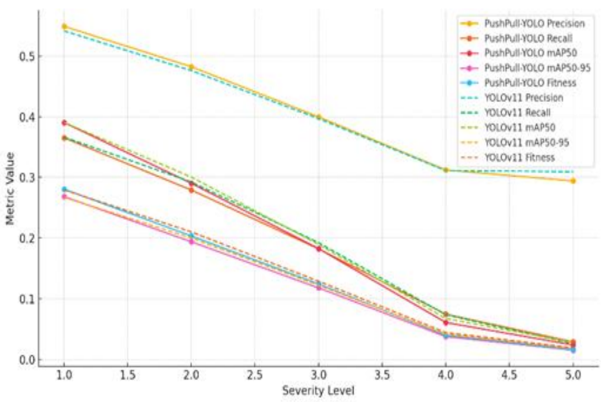

Figure 16. Performance comparison of YOLOv11 and PushPull-YOLO under shot noise corruption

4.14 Snow corruption

Snow corruption disrupts detection capabilities, particularly at higher severities. PushPull-YOLO maintains better mAP50-95 and fitness metrics compared to YOLOv11, as shown in Table 15, confirming its effectiveness for outdoor applications in snowy environments.

The Wilcoxon signed-rank test was conducted to compare the performance of PushPull-YOLO and YOLOv11 across five key metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The analysis revealed statistically significant differences in favor of PushPull-YOLO for all metrics ($p = 0.03125$), demonstrating its consistent superiority over YOLOv11. These results highlight the enhanced detection accuracy, robustness across varying IoU thresholds, and overall model efficiency offered by PushPull-YOLO. The significant improvements in Precision and Recall further emphasize PushPull-YOLO's capability to achieve both high detection accuracy and comprehensive recall, making it a more effective and reliable choice for object detection tasks in diverse and challenging conditions. These findings reinforce PushPull-YOLO's position as a robust advancement in object detection technology. Comparison of YOLOv11 and PushPull-YOLO performance metrics under snow corruption across varying severity levels is shown in Figure 17.

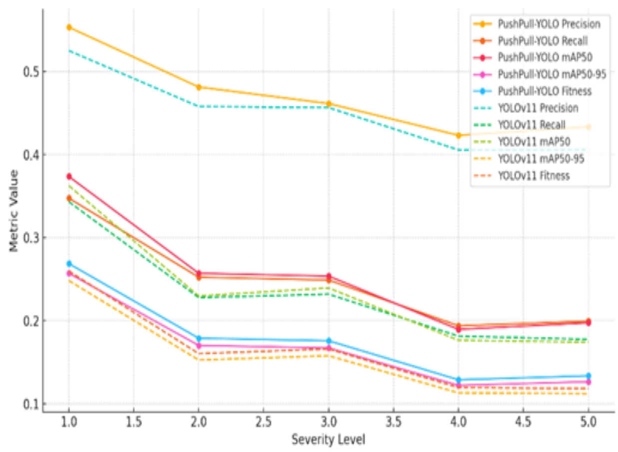

Figure 17. Performance comparison of YOLOv11 and PushPull-YOLO under snow corruption

Table 14. Performance metrics for PushPull-YOLO and YOLOv11 under shot noise corruption at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	Recall	mAP50	mAP50-95	Fitness	Precision	Recall	mAP50	mAP50-95	Fitness	
0.5487	0.3654	0.3904	0.2685	0.2807	0.5413	0.3663	0.3905	0.2674	0.2797	1
0.4823	0.2793	0.2905	0.1941	0.2037	0.4759	0.2926	0.3006	0.2003	0.2103	2
0.3997	0.1825	0.1829	0.1183	0.1247	0.3968	0.1926	0.1898	0.1225	0.1292	3
0.3123	0.0748	0.0613	0.0381	0.0404	0.3113	0.0739	0.0677	0.0420	0.0446	4
0.2943	0.0293	0.0243	0.0157	0.0166	0.3095	0.0265	0.0285	0.0184	0.0194	5
0.4075	0.1863	0.1899	0.1269	0.1332	0.4070	0.1904	0.1954	0.1301	0.1366	Avg.

Table 15. Performance metrics for PushPull-YOLO and YOLOv11 methods under snow corruption at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	**Precision**	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	
0.5532	0.3475	0.3736	0.2570	0.2687	0.5251	0.3431	0.3625	0.2483	0.2597	1
0.4813	0.2524	0.2570	0.1702	0.1789	0.4578	0.2278	0.2297	0.1527	0.1604	2
0.4614	0.2490	0.2536	0.1672	0.1758	0.4565	0.2320	0.2394	0.1578	0.1660	3
0.4231	0.1938	0.1896	0.1222	0.1289	0.4056	0.1815	0.1765	0.1132	0.1195	4
0.4334	0.1994	0.1975	0.1266	0.1336	0.4059	0.1775	0.1742	0.1122	0.1184	5
0.4705	0.2484	0.2543	0.1686	0.1772	0.4502	0.2324	0.2364	0.1568	0.1648	Avg.

Table 16. Performance metrics for PushPull-YOLO and YOLOv11 under zoom blur corruption at varying severity levels.

PushPull-YOLO					YOLOv11					Severity Level
Precision	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	**Precision**	**Recall**	**mAP50**	**mAP50-95**	**Fitness**	
0.4410	0.2415	0.2404	0.1313	0.1422	0.4294	0.2342	0.2304	0.1259	0.1363	1
0.3639	0.1851	0.1737	0.0857	0.0945	0.3649	0.1734	0.1653	0.0816	0.0900	2
0.3406	0.1622	0.1476	0.0692	0.0770	0.3444	0.1494	0.1369	0.0636	0.0709	3
0.3065	0.1286	0.1132	0.0491	0.0556	0.2818	0.1213	0.1032	0.0444	0.0503	4
0.2806	0.1185	0.0986	0.0413	0.0470	0.2535	0.1098	0.0880	0.0362	0.0413	5
0.3465	0.1672	0.1547	0.0753	0.0833	0.3348	0.1576	0.1448	0.0703	0.0778	Avg.

4.15 Zoom blur corruption

Zoom blur affects object detection metrics as severity increases. PushPull-YOLO outperforms YOLOv11 in mAP50-95 across all levels, demonstrating its capability to handle variable focal lengths. The comparative results are presented in Table 16.

The Wilcoxon signed-rank test was applied to evaluate the performance differences between PushPull-YOLO and YOLOv11 across five key metrics: Precision, Recall, mAP50, mAP50-95, and Fitness. The analysis revealed statistically significant differences in favor of PushPull-YOLO for Recall, mAP50, mAP50-95, and Fitness ($p = 0.03125$ for all), indicating its superior performance in terms of detection accuracy, robustness across varying IoU thresholds, and overall model fitness. No statistically significant difference was observed for Precision ($p = 0.15625$), suggesting comparable performance between the two methods in this metric. These findings highlight PushPull-YOLO's ability to deliver higher recall and improved detection capabilities, making it a more effective solution for object detection tasks, particularly in scenarios requiring high accuracy and robustness. Comparison of YOLOv11 and PushPull-YOLO performance metrics under zoom blur corruption across varying severity levels is shown in Figure 18.

The integration of PushPull convolution into YOLOv11 provides substantial robustness against image corruptions while maintaining real-time performance. This robustness is achieved through the PushPull-Conv unit's ability to combine excitatory and inhibitory convolutional kernels, mimicking biological mechanisms in the primary visual cortex. The push kernel focuses on detecting preferred stimuli, while the pull kernel suppresses responses in irrelevant regions, effectively enhancing selectivity and minimizing noise from non-relevant features. This dynamic interaction allows the network to maintain high detection accuracy under challenging conditions. This study demonstrated the consistent advantages of PushPull-YOLO in handling challenging object detection scenarios compared to standard YOLOv11. PushPull-YOLO achieves superior generalization, particularly for unseen corruptions, due to the inhibition-driven convolutional units. These units operate by incorporating complementary push and pull kernels, where the push kernel highlights critical features, and the pull kernel reduces interference from irrelevant stimuli. This dual mechanism enhances the model's ability to distinguish relevant patterns, improving its adaptability to novel corruption types and ensuring robust performance across diverse datasets.

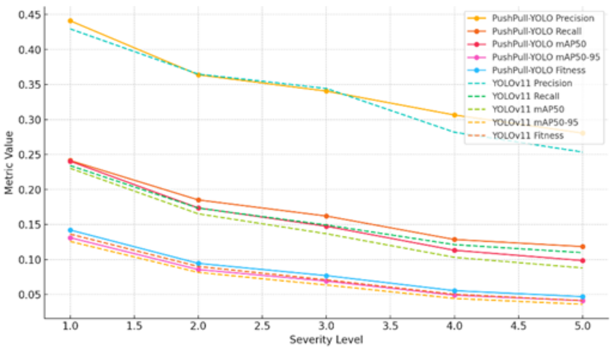

Figure 18. Performance comparison of YOLOv11 and PushPull-YOLO under zoom blur corruption

Ablation studies revealed that a good starting point inhibits the strength set via backpropagation around 0.5, with a kernel size tuned to 3×3 for most applications. In this study we set kernel size to 7×7 according to Section3.4. These configurations were shown to provide a balancing effect on robustness and detection accuracy, especially under diverse image corruption conditions. Statistical analyses also supported improved metrics by PushPull-YOLO in mean Average Precision (mAP) and Fitness, with significant advantages in mAP50-95. Whereas improvement in

Precision is not conclusive, the improvement for Recall was substantial to know a better coverage was obtained without any loss in precision.

These results indicate that the PushPull mechanism further enhances the state-of-the-art performance of YOLO models for practical applications, especially when the application scenarios demand robustness against image distortions due to external factors such as those in the self-driving car ecosystem when there is heavy rain or fog outside, inspection of defects using industrial quality control systems under changing illuminations, and other surveillance systems with degraded visibility or motion blur. This will surely make PushPull-YOLO one of the trustworthy options in all future autonomous systems, industrial automation, and surveillance applications that require high accuracy with high efficiency.

5 Discussion

This comparative study of PushPull-YOLO and YOLOv11 on such a wide range of corruption types has shed light on the critical aspects of robustness, adaptability, and performance of PushPull-YOLO. Both models put in a very impressive performance, but PushPull-YOLO stands out in handling a range of real-world challenges with remarkable improvements in essential metrics like mAP50-95 and Fitness. These metrics provide a clear insight into the exactness of detection over varying IoU thresholds and relate directly to the practical utility of the model.

Although PushPull-YOLO exhibits a minor trade-off in precision compared to YOLOv11 for certain corruption types, this is significantly outweighed by its gains in other critical performance areas. For instance, under brightness corruption, PushPull-YOLO achieves a precision of 0.5965, closely approximating YOLOv11's 0.5922, demonstrating its capability to sustain high detection quality even in challenging scenarios. Similarly, recall values indicate PushPull-YOLO's consistent performance. Under fog corruption, the recall of 0.3982 aligns with YOLOv11's 0.3993, indicating that PushPull-YOLO maintains reliable detection coverage despite visual obstructions. Across various scenarios, PushPull-YOLO balances precision and recall effectively, ensuring dependable detection even under adverse conditions.

PushPull-YOLO's robustness is particularly evident in its mAP50-95 performance under severe corruptions. For example, in elastic transform corruptions, PushPull-YOLO achieves a mAP50-95 of 0.2307, surpassing YOLOv11's 0.2273. Likewise, under frost and fog corruptions, PushPull-YOLO's resilience to environmental variations becomes apparent, making it a preferred choice for unpredictable, dynamic environments. This robustness highlights its practical utility in applications such as autonomous systems, industrial quality control, and surveillance where external factors often degrade image quality.

PushPull-YOLO's performance under specific corruption types further emphasizes its adaptability. For brightness and fog corruptions, its ability to maintain superior mAP50-95 metrics underscores its enhanced capacity to handle extreme conditions. For pixelation and Gaussian noise, PushPull-YOLO's adaptability to low-resolution and noisy images ensures reliable functionality in scenarios like remote monitoring and navigation. Notably, under motion blur, zoom blur, and snow corruptions, PushPull-YOLO's detection capabilities remain consistently robust, demonstrating its reliability in handling real-time, dynamic image sequences.

The architectural modifications introduced in PushPull-YOLO—specifically the integration of the PushPull-Conv mechanism—play a significant role in its superior performance. This biologically inspired design emulates the selective inhibition mechanisms observed in the primary visual cortex, effectively enhancing feature selectivity and suppressing irrelevant stimuli. By dynamically balancing excitation and inhibition, PushPull-YOLO excels in distinguishing critical patterns and mitigating the effects of noise and distortions. Ablation studies highlight the importance of optimized configurations, such as inhibition strength and kernel size, in achieving a harmonious balance between robustness and accuracy. These innovations position PushPull-YOLO as an ideal solution for high-stakes applications, such as autonomous vehicles navigating adverse weather, precision manufacturing, and critical infrastructure surveillance.

Statistical analyses further validate the robustness of PushPull-YOLO. The Wilcoxon signed-rank test underscores its significant advantages in mAP50-95 and Fitness metrics ($p = 0.03125$), establishing its superiority in comprehensive detection scenarios. Although precision differences remain statistically insignificant ($p = 0.21875$), borderline significance in recall ($p = 0.09375$) suggests that PushPull-YOLO may offer a subtle yet meaningful improvement in detection coverage under specific conditions. These findings solidify PushPull-YOLO's position as a balanced and reliable framework for demanding object detection tasks.

As such, it demonstrates PushPull-YOLO's robustness to a wider range of corruptions. For instance, under defocus and glass blur scenarios, PushPull-YOLO retains superior mAP50-95 values, ensuring effective detection even in out-of-focus imagery. Similarly, its performance under impulse noise and JPEG compression corruptions highlights its adaptability to data artifacts commonly encountered in compressed or degraded datasets. PushPull-YOLO's resilience to extreme image corruptions, including frost, snow, and zoom blur, further cements its utility in outdoor applications and environments with frequent visual distortions.

PushPull-YOLO's demonstrated robustness makes it particularly relevant for critical real-world applications. Autonomous systems operating in adverse environments, industrial automation requiring precise quality control, and security systems monitoring low-visibility scenarios can all benefit from its advanced capabilities. Furthermore, the integration of biologically inspired mechanisms opens avenues for future exploration in neuro-mimetic computing and adaptive AI systems. These advancements suggest that future iterations of PushPull-YOLO could further refine its adaptability, extending its application scope to even more challenging and dynamic scenarios.

PushPull-YOLO establishes itself as a transformative advancement in object detection, offering unparalleled robustness across diverse corruption types. Its innovative architecture, inspired by biological processes, allows it to effectively address the challenges posed by dynamic and unpredictable environments. By consistently delivering superior performance in mAP metrics and overall detection reliability, PushPull-YOLO underscores the potential of biologically inspired designs in advancing AI technologies. These findings not only position PushPull-YOLO as a leader in object detection but also set a benchmark for future explorations into robust, adaptive, and efficient detection frameworks.

6. Conclusions

Integration of PushPull convolution into the YOLOv11 architecture sets up a solid and efficient framework for object detection, particularly in challenging cases that involve image corruptions. Drawing inspiration from neurophysiological mechanisms, PushPull convolution enhances the selectivity for the stimulus of interest by employing complementary push and pull filters. This mechanism self-regulates the balance between excitatory and inhibitory responses, thereby mitigating the impact of such corruptions as noise, blur, and digital artifacts. This integration follows the biological inspiration of how best equipping the model to process such complex visual data in various conditions would be. Advanced architectural components like the C3k2 block and C2PSA module, YOLOv11 presents state-of-the-art object detection with high scores of mAP and computational efficiency.

These features are complemented by the addition of PushPull convolution, which strengthens the model's performance against common corruptions, as demonstrated in extensive evaluations on benchmark datasets such as ImageNet-C. This leverages the rapid inference capabilities and precise detection of YOLOv11, while PushPull convolution significantly contributes to resilience against adverse scenarios. Experimental results showed that the combined approach does much better compared to state-of-the-art approaches in handling high-frequency corruptions, which are of particular importance in real-time applications like autonomous navigation, surveillance, and industrial automation. This hybrid model successfully fulfills the application demands by facilitating the dual advantages of robustness and accuracy.

Moreover, the implications of this integration hold promise for the deployment of advanced object detection systems in resource-constrained environments, including edge devices, where efficiency and accuracy are paramount. This approach also highlights the potential for developing models that can adapt to unforeseen challenges in dynamic real-world scenarios, making it a versatile tool for a variety of industries.

This work provides the way for further interdisciplinary innovations in computer vision by providing a new perspective on robustness in object detection tasks. The work demonstrates how integrating biologically inspired mechanisms with state-of-the-art architectures can lead to superior performance. Future work could explore extending this methodology to other tasks such as segmentation, key-point detection, and pose estimation. These extensions will broaden its applicability in a variety of real-world scenarios, from medical imaging to environmental monitoring, and could ultimately address more complex challenges in computer vision.

Conflict of interest

The author declare that they have no conflicts of interest related to this work.

Similarity rate (iThenticate): %11

References

[1] Y. Dong, C. Kang, J. Zhang, Z. Zhu, Y. Wang, Y. Xiao, H. Su, X. Wei, and J. Zhu, Benchmarking robustness of 3d object detection to common corruptions in autonomous driving, arXiv Preprint, 2023. https://doi.org/10.48550/arxiv.2303.11040.

[2] H. A. Akyürek, H. İ. Kozan, and Ş. Taşdemir, Surface crack detection in historical buildings with deep learning-based YOLO algorithms: a comparative study, Computational Research Progress in Applied Science & Engineering, 10, 3, 1-14, 2024. https://doi.org/10.61186/crpase.10.3.2904.

[3] M. Akgül, H. İ. Kozan, H. A. Akyürek, and Ş. Taşdemir, Automated stenosis detection in coronary artery disease using yolov9c: Enhanced efficiency and accuracy in real-time applications, Journal of Real-Time Image Processing, 21, 177, 2024. https://doi.org/10.1007/s11554-024-01558-x.

[4] H. A. Akyürek, Tıbbi Görüntülerde Yapay Zekâ Temelli Nesne Tespiti Uygulamaları, in Bilgisayar Bilimleri ve Mühendisliğinde İleri Araştırmalar, G. Kutluana Ed. Afyonkarahisar: YAZ Yayınları, 2024. https://doi.org/10.5281/zenodo.14604344.

[5] H. İ. Kozan and H. A. Akyürek, Efficient and rapid classification of various maize seeds using transfer learning and advanced AI techniques, ASEAN Journal of Scientific and Technological Reports, 28, 1, e255200, 2025. https://doi.org/10.55164/ajstr.v28i1.255200.

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, You only look once: unified, real-time object detection, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 779-788, 2016. https://doi.org/10.1109/cvpr.2016.91.

[7] R. Kishor, Performance benchmarking of YOLOv11 variants for real-time delivery vehicle detection: a study on accuracy, speed, and computational trade-offs, Asian Journal of Research in Computer Science, 17, 12, 108-122, 2024. https://doi.org/10.9734/ajrcos/2024/v17i12532.

[8] R. Khanam and M. Hussain, YOLOv11: an overview of the key architectural enhancements, arXiv Preprint, 2024. https://doi.org/10.48550/arXiv.2410.17725.

[9] H. Hsiang, Development of an effective corruption-related scenario-based testing approach for robustness verification and enhancement of perception systems in autonomous driving, Sensors, 24, 1, 301, 2024. https://doi.org/10.3390/s24010301.

[10] C. Michaelis, B. Mitzkus, R. Geirhos, E. Rusak, O. Bringmann, A. S. Ecker, M. Bethge, and W. Brendel, Benchmarking robustness in object detection: autonomous driving when winter is coming, arXiv Preprint, 2020. https://doi.org/10.48550/arXiv.1907.07484.

[11] G. S. Bennabhaktula, E. Alegre, N. Strisciuglio, and G. Azzopardi, PushPull-Net: inhibition-driven ResNet robust to image corruptions, in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2025, 15308 LNCS, 391-408, https://doi.org/10.1007/978-3-031-78186-5_26.

[12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár and C. L. Zitnick, Microsoft COCO: common objects in context, Cham, 2014: Springer International Publishing, in Computer Vision – ECCV 2014, 740-755. https://doi.org/10.1007/978-3-319-10602-1_48.

[13] P. Varghese, Biologically inspired deep residual networks, Iaes International Journal of Artificial Intelligence (Ij-Ai), vol. 12, no. 4, p. 1873, 2023. https://doi.org/10.11591/ijai.v12.i4.pp1873-1882.

[14] R. Pogodin, Y. Mehta, T. P. Lillicrap, and P. E. Latham, Towards biologically plausible convolutional networks, 2021. https://doi.org/10.48550/arxiv.2106.13031.

[15] Y. Shen, J. Wang, and S. Navlakha, A correspondence between normalization strategies in artificial and biological neural networks, Neural Computation, 33 ,12 3179–3203, 2021. https://doi.org/10.1162/neco_a_01439.

[16] M. Zingerenko, Bipolar morphological YOLO network for object detection, Proceedings Volume 13072, Sixteenth International Conference on Machine Vision 130720Q, 2024. https://doi.org/10.1117/12.3023255.

[17] D. H. Hubel and T. N. Wiesel, Receptive fields, binocular interaction and functional architecture in the cat's visual cortex, The Journal of Physiology, 160, 1, 106, 1962. https://doi.org/10.1113/jphysiol.1962.sp006837.

[18] G. Azzopardi, A. Rodríguez-Sánchez, J. Piater, and N. Petkov, A push-pull CORF model of a simple cell with antiphase inhibition improves SNR and contour detection, PloS one, 9, 7, e98424, 2014. https://doi.org/10.1371/journal.pone.0098424.

[19] L. J. Borg-Graham, C. Monier, and Y. Fregnac, Visual input evokes transient and strong shunting inhibition in visual cortical neurons, Nature, 393, 6683, 369-373, 1998. https://doi.org/10.1038/30735.

[20] D. H. Hubel and T. N. Wiesel, Brain mechanisms of vision, Scientific American, 241, 3, 150-163, 1979. https://doi.org/10.1038/scientificamerican0979-150.

[21] H. Zhang and Y. Wang, Towards adversarially robust object detection, 2019 IEEE/CVF International Conference on Computer Vision, Seoul, Korea, pp. 421-430, 2019. https://doi.org/10.1109/iccv.2019.00051.

[22] R. Poojary, R. Raina, and A. K. Mondal, Effect of data-augmentation on fine-tuned cnn model performance, Iaes International Journal of Artificial Intelligence, 10, 84, 2021. https://doi.org/10.11591/ijai.v10.i1.pp84-92.

[23] P. Shah and M. El-Sharkawy, A-MnasNet: augmented MnasNet for computer vision, 2020 IEEE 63rd International Midwest Symposium on Circuits and Systems (MWSCAS), pp. 1044-1047, Springfield, MA, USA, 2020. https://doi.org/10.1109/mwscas48704.2020.9184619.

[24] C. Shorten and T. M. Khoshgoftaar, A survey on image data augmentation for deep learning, Journal of Big Data, 6, 1, 2019. https://doi.org/10.1186/s40537-019-0197-0.

[25] Ultralytics. Ultralytics YOLO documentation. https://docs.ultralytics.com/models/yolo11/, Accessed 20 March 2025.

[26] Y. Luo, The evolution of YOLO: from YOLOv1 to YOLOv11 with a focus on YOLOv7's innovations in object detection, Theoretical and Natural Science, 87, 1, 82-90, 2025. https://doi.org/10.54254/2753-8818/2025.20335.

[27] E. H. Alkhammash, A comparative analysis of YOLOv9, YOLOv10, YOLOv11 for smoke and fire detection, Fire, 8, 1, 26, 2025. https://doi.org/10.3390/fire8010026.

[28] R. Bohush, S. Ablameyko, and Y. Adamovskiy, Robust object detection in images corrupted by impulse noise, Computer Modeling and Intelligent Systems, vol. 2608, pp. 1107-1116, 2020. https://doi.org/10.32782/cmis/2608-83.

[29] B. Jaison, G. Anjali Jha, J. Jeevitha, and C. Devi Priya, You only look once(YOLO) object detection with COCO using machine learning, in International Interdisciplinary Humanitarian Conference for Sustainability, IIHC 2022 - Proceedings, 2022. pp. 1574-1578, https://doi.org/10.1109/IIHC55949.2022.10059737.

[30] A. Borji, Complementary datasets to COCO for object detection, arXiv Preprint, 2022. https://doi.org/10.48550/arxiv.2206.11473.

[31] R. F. Woolson, Wilcoxon signed-rank test, Encyclopedia of Biostatistics, 8, 2005. https://doi.org/10.1002/0470011815.b2a15177.

[32] C. Lu, J. Wu, S. Ali, and M. L. Olsen, Assessing the uncertainty and robustness of object detection models for detecting stickers on laptops, arXiv Preprint, 2024. https://doi.org/10.48550/arXiv.2409.03782.