

# Introducing Multiagent Systems to AV Visual Perception Sub-tasks: A proof-of-concept implementation for bounding-box improvement

Alaa Daoud<sup>1</sup>, Corentin Bunel<sup>1</sup> and Maxime Guériau<sup>1</sup>

<sup>1</sup>INSA Rouen Normandie, Univ Rouen Normandie, Univ Le Havre Normandie, Normandie Univ, LITIS UR 4108, F-76000 Rouen, France

## Abstract

Object detection is a pivotal task in computer vision, with applications spanning from autonomous driving to surveillance. Traditionally, methods like Non-Maximum Suppression (NMS) and its variants have been used to refine object detection outputs. Fusing predictions from different object detection models using confidence scores to average overlapping bounding boxes from multiple detection models has demonstrated superior performance over conventional methods. In this work, we employ multiple agents, each responsible for handling individual bounding boxes, to generate an improved fused prediction. This agent-based adaptation aims to leverage decentralized processing to potentially increase the system's efficiency and adaptability across various object detection scenarios, particularly in autonomous vehicle (AV) perception systems. We develop two distinct behaviors for the bounding box agents: one replicating the state-of-the-art Weighted Boxes Fusion (WBF) method in a decentralized manner, and the other introducing competitive behavior where agents interact based on Intersection over Union (IoU) and confidence values. We evaluate the performance of our approach using the COCO dataset, demonstrating the flexibility and potential of integrating MAS into object detection workflows including those for AV perception systems.

## Keywords

Autonomous driving, perception systems, bounding-box refinement, Multiagent Systems

## 1. Introduction

Autonomous vehicles and intelligent transport systems depend on advanced computer vision technologies, with object detection being a critical task. This enables vehicles to recognize and respond to surrounding objects effectively, with region proposal identifying potential object locations early in the detection process, crucial for timely responses in autonomous driving [1]. Traditional techniques like Non-Maximum Suppression (NMS) often struggle to balance precision and recall, especially in dynamic environments. Solovyev et al. [2] introduced Weighted Boxes Fusion (WBF), using confidence scores to average overlapping bounding boxes from multiple detection models, demonstrating superior performance over conventional methods.

The integration of Multiagent Systems (MAS) into object detection workflows offers new perspectives to address traditional challenges [3, 4]. MAS provide dynamic and adaptable decision-making capabilities, enhancing autonomous vehicles' ability to handle complex, unpredictable road conditions. MAS support distributed and adaptive processing [5], complementing modern GPU-based computer vision. By distributing tasks across agents, MAS enhances system flexibility and resilience, especially in dynamic environments like autonomous driving or video surveillance [6, 7, 8]. Each agent manages a subset of tasks, improving resilience to errors [9, 10].

MAS can adjust strategies based on scenarios [11, 12], adapting parameters for bounding box fusion based on context, scene complexity, or environmental changes [13]. Agents operate independently on different hardware, optimizing processing power and allowing system scalability [14]. Local decisions are combined through a global process, enhancing accuracy [15]. MAS can continually learn from

---

13th International Workshop on Agents in Traffic and Transportation (ATT 2024) held in conjunction with ECAI 2024

✉ alaa.daoud@insa-rouen.fr (A. Daoud); corentin.bunel@insa-rouen.fr (C. Bunel); maxime.gueriau@insa-rouen.fr (M. Guériau)

ORCID 0000-0002-3640-327X (A. Daoud); 0000-0001-6637-9795 (C. Bunel); 0000-0002-8742-6623 (M. Guériau)



© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

their environment and from the interactions between agents [16]. This potential for adaptive learning motivates the agentification approach, as it opens the possibility for future enhancements. By achieving an agentified method, we can later integrate learning capabilities to further improve adaptability and performance in evolving object detection scenarios. Agent-based approaches are well-suited for integrating diverse models and data sources [17], which is essential for the ensemble approaches used in WBF where predictions from different models are combined.

Agentifying output refinement methods such as NMS or WBF involves assigning individual agents to handle specific bounding boxes, enabling dynamic adjustment based on individual box characteristics. This approach addresses real-time processing requirements and improves scalability and fault tolerance by decentralizing the decision-making process [18, 19, 20]. In this work, we aim to design and implement a proof-of-concept system integrating MASs into the process of improving bounding boxes in object detection. We will develop two behaviors for the bounding box agents: one replicating the state-of-the-art Weighted Boxes Fusion (WBF) method in a decentralized manner, and the other introducing competitive behavior where agents interact based on Intersection over Union (IoU) and confidence values. Finally, we will deploy the system and assess its performance using the COCO dataset, testing various levels of competition and cooperation between agents. The remainder of this paper is structured as follows: Section 2 presents the related work in object detection, multiagent systems, and their integration. Section 3 details the system architecture and design principles of the AWBF method. Section 4 describes the implementation of the proof-of-concept system and the development of agent behaviors. Section 5 discusses the experimental setup and evaluation using the COCO dataset. Section 6 presents the results and analysis of the experimental evaluation. Section 7 concludes the paper with a summary of findings and future work directions.

## 2. Related Work

Object detection is a fundamental task in computer vision, critical for intelligent transportation systems (ITS) applications such as autonomous driving, traffic monitoring, and surveillance. The integration of MASs into object detection workflows offers significant potential to enhance system efficiency, robustness, and adaptability. This section reviews recent advancements in object detection techniques relevant to the ITS, with a focus on bounding box fusion and the role of MASs.

### 2.1. Bounding Box Improvement Techniques

Wang et al. (2019) presented the Multi-Stage Complementary Fusion (MCF3D) network, an end-to-end architecture for 3D object detection that integrates LiDAR and RGB data. This network employs attention mechanisms and prior knowledge to achieve state-of-the-art results, enhancing the detection accuracy necessary for autonomous driving applications [21].

Qian et al. (2020) proposed an improved object detection method for remote sensing images, incorporating a novel bounding box regression loss and a multi-level features fusion module. This method enhances the precision of object localization, which is crucial for applications such as traffic monitoring and vehicle detection [22].

Solovyev et al. (2021) introduced the Weighted Boxes Fusion (WBF) method, which averages overlapping bounding boxes from multiple detection models using confidence scores. This approach demonstrated superior performance over traditional techniques, highlighting the effectiveness of fusion methods in improving object detection accuracy [2]. This method is particularly relevant for ITS applications where robust and accurate object detection is paramount for safety and efficiency.

Zhang and Wu (2022) proposed a multi-view feature adaptive fusion framework that enhances 3D object detection by optimizing depth feature fusion and loss function design. This approach improves the regression accuracy of bounding boxes, which is essential for ITS applications where precise object localization is critical [23].

Liu et al. (2023) developed the Fusion network by Box Matching (FBMNet) for multi-modal 3D detection. This method aligns features at the bounding box level, providing stability in challenging

scenarios such as asynchronous sensors and misaligned sensor placements, common issues in ITS applications [24].

## 2.2. Multiagent Systems in Object Detection

Introducing MAS to object detection and computer vision systems is not a new idea. For example Choksuriwong et al. (2005) developed a MAS for image understanding that localizes and recognizes objects using a distributed system implemented on a cluster computer. This approach leverages invariant features and supervised classification to improve object recognition accuracy, which is vital for traffic monitoring systems [25]. However, the application of MAS in these areas has decreased recently with the advancements in machine learning techniques and their improved performance in handling object detection tasks. Despite this shift, some researchers have continued to explore the potential of MAS in object detection through various approaches.

Jiang et al. (2019) proposed a multi-agent deep reinforcement learning (MADRL) approach for multi-object tracking, using YOLO V3 for object detection and Independent Q-Learners (IQL) for policy learning. This method achieves better performance in precision, accuracy, and robustness compared to other state-of-the-art methods, which is particularly beneficial for real-time traffic monitoring and surveillance [26].

Fekir and Benamrane (2015) introduced a MAS for boundary detection and object tracking using active contours and multi-resolution treatment. This system improves object boundary detection and tracking through cooperative agent strategies, enhancing the accuracy and efficiency of ITS applications such as vehicle and pedestrian tracking [27].

Vincent et al. (2022) described a MAS using stereovision for perception, enabling agents to collaborate and enhance scene understanding through graph matching algorithms. This approach addresses challenges in correspondence identification and non-covisibility, critical for ITS applications such as multi-vehicle coordination and traffic management [28].

Mahmoudi et al. (2013) utilized a MAS for object recognition in complex urban areas, leveraging WorldView-2 satellite imagery and digital surface models. This system improves object recognition accuracy through knowledge-based reasoning and cooperative agent capabilities, essential for urban traffic monitoring and smart city applications [29].

## 2.3. Positioning Our Proposal

In light of the existing work, our proposal aims to integrate the strengths of both bounding box fusion techniques and MASs to develop a more robust and efficient object detection framework tailored for ITS applications. Our approach leverages the distributed processing capabilities of MASs to enhance the accuracy and scalability of bounding box fusion methods. By incorporating advanced fusion techniques and adaptive agent strategies, our system aims to address the limitations of existing methods, such as handling dynamic environments and improving detection precision. Our contributions include:

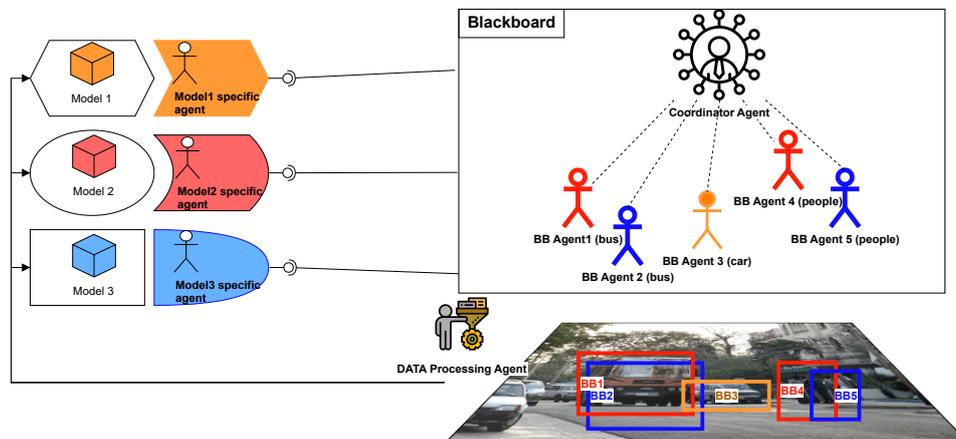
1. A multi-agent based framework for bounding box improvement that dynamically assigns agents to handle specific bounding boxes.
2. Integration of advanced fusion techniques, such as Weighted Box Fusion (WBF) and Non-Maximum Suppression (NMS), to enhance detection accuracy in various ITS scenarios.
3. Implementation of adaptive agent strategies / behaviors that allow the switch between cooperation and competition dynamically, ensuring robust performance in real-world ITS applications.

To the best of our knowledge, we are among the first to propose integrating MAS into specific computer vision sub-tasks such as bounding box filtering and fusion. This approach aims to exploit the advantages of MAS to enhance the accuracy, efficiency, and adaptability of object detection systems in ITS applications.

### 3. System Architecture for AWBF

The agentified Weighted Boxes Fusion (WBF) system integrates multiple agents, each handling individual bounding boxes from various detection models. This Multiagent System (MAS) enhances the efficiency and accuracy of bounding box fusion through distributed processing and specialized agent roles. A central blackboard mechanism facilitates information sharing and coordination.

MAS offers decentralized decision-making and dynamic adaptability, enhancing resilience and flexibility in handling varied scenarios [30].



**Figure 1:** AWBF Agents, colors correspond to the models who generate initial bounding boxes

The blackboard acts as a global communication hub, simplifying data interactions and providing a robust framework for synchronized information exchange among agents [31]. Specific agent roles, from bounding box processing to model-specific adaptations, optimize performance and accuracy by leveraging domain-specific knowledge and algorithms [32]. Feedback mechanisms enable dynamic adaptation, allowing agents to adjust strategies based on performance and data input changes, maintaining high accuracy in dynamic environments [33].

#### 3.1. Overview of Agent Roles

The system includes various agents with specific responsibilities:

- **Bounding-Box Agents:** Handle individual bounding boxes, analyze, and propose fusions with overlapping boxes.
- **Model-specific Agents:** Manage bounding boxes from specific detection models. Can be seen as interfaces between the MAS and CV models. Each agent extracts bounding box proposals from its respective model to ensure compatibility and apply model-specific behaviors and adjustments.
- **Coordinator Agents:** Oversee the fusion process, resolve conflicts between bounding-box agents, and make final decisions on merged bounding boxes.
- **Data Processing Agents:** Optionally handle image preprocessing and result postprocessing.

#### 3.2. Blackboard Information Sharing System

The blackboard serves as a shared information space for communication and data exchange:

- **Data Repository:** Central storage for bounding box data, including coordinates, confidence scores, and model origins.
- **Communication Medium:** Allows agents to read and write data, maintaining system modularity and scalability.
- **Coordination Facilitator:** Coordinates actions among agents, especially in resolving fusion conflicts.

### 3.3. Processing Workflow

The workflow involves:

1. **Data Input and Distribution:** Model-specific Agents extract bounding boxes from different models and transfer them to Data processing agents who prepare and distribute bounding boxes to bounding box agents.
2. **Bounding Box Analysis and Posting:** Bounding box agents analyze and post findings to the blackboard, proposing fusions.
3. **Review and Fusion:** Coordinator agents review and finalize fusion decisions, consulting model-specific agents as needed.
4. **Final Processing and Output:** Data processing agents optimize the fused bounding boxes for downstream applications.
5. **Feedback and Adaptation:** The system adapts to changes by updating agent strategies or parameters based on performance metrics.

## 4. Implementation and Development of Agent Behaviors

Our implementation is developed in Python, utilizing the existing WBF codebase to maintain consistency in data processing. By forking the original WBF repository, we leverage developed libraries, utilities, and functions, ensuring the use of the exact same logic in data processing. This allowed us to focus on integrating MAS features without reinventing the core bounding box fusion logic. We built an ad-hoc MAS framework tailored to our requirements. The agents interact via a shared blackboard for communication, and the system supports both centralized and decentralized processing. Following the system architecture described in the previous section, one can implement diverse behaviors and a variety of solution method logics by only changing the decision logic of the bounding box agent and adjusting the coordination mechanism.

Model-specific Agents interact with existing object detection models (e.g., YOLO, Faster R-CNN) to receive and process bounding boxes. Model-specific agents convert detection outputs into a standard format used by the system.

Main implementation challenges included managing computation time, communication overhead, and integrating the MAS with existing computer vision models. Future improvements will focus on developing variety of agent behaviors with optimized parameters for computational and accuracy performance, enhancing the system’s scalability, robustness, and adaptability, exploring further integration with advanced machine learning models and real-world deployment scenarios.

### 4.1. Agent Behaviors

We developed two distinct agent behaviors to demonstrate the versatility and potential of MAS in object detection. The first behavior replicates the Weighted Boxes Fusion (WBF) in a decentralized manner, while the second introduces a competitive interaction among agents.

#### 4.1.1. Behavior 1: Decentralized Weighted Boxes Fusion (WBF)

This behavior replicates the state-of-the-art WBF method in a decentralized manner. Each agent processes bounding boxes independently and posts results to a shared blackboard (see Algorithm 1), improving system resilience. The agent determines overlapping boxes as candidates for fusion by calculating the Intersection over Union (IoU). Boxes are considered for fusion if their IoU exceeds a certain threshold. The IoU calculation is given by:

$$IoU(Box_1, Box_2) = \frac{area(Box_1 \cap Box_2)}{area(Box_1 \cup Box_2)}$$

---

**Algorithm 1** Decentralized WBF Algorithm (AWBF) - BoundingBox Agent behavior

---

- 1: **Input:** Bounding boxes  $B$ , confidence scores  $S$ , labels  $L$
  - 2: Read  $B, S, L$  from the blackboard
  - 3: Determine overlapping boxes as candidates for fusion
  - 4: Filter candidates using IoU metric
  - 5: Apply WBF on the final set of candidates
  - 6: Post fused boxes to the blackboard
  - 7: **Output:** Fused bounding boxes
- 

---

**Algorithm 2** Competitive Interaction Algorithm - BoundingBox Agent  $A_i$  behavior

---

- 1: **Input:** Bounding boxes  $B$ , confidence scores  $S$ , labels  $L$
  - 2:  $A_i$  reads overlapping boxes from the blackboard
  - 3: **for** each overlapping box  $B_j$  **do**
  - 4:     Calculate  $IoU$  and  $IoB$  between  $A_i$  and  $B_j$
  - 5:     Calculate attack strength  $S_{\text{attack}} = \text{confidence}_{A_i} \times IoB_{B_j}$
  - 6:     Calculate defense strength  $S_{\text{defense}} = \text{confidence}_{B_j} \times IoB_{A_i}$
  - 7:     Calculate result  $R = S_{\text{attack}} - S_{\text{defense}}$
  - 8:     **if**  $R > T$  **then**
  - 9:          $A_i$  wins and  $B_j$  is removed
  - 10:     **else if**  $R < -T$  **then**
  - 11:          $B_j$  wins and  $A_i$  is removed
  - 12:     **else**
  - 13:         Fuse  $A_i$  and  $B_j$  using weighted average
  - 14:     **end if**
  - 15: **end for**
  - 16:  $A_i$  posts result to the blackboard
  - 17: **Output:** Final bounding boxes
- 

#### 4.1.2. Behavior 2: Competitive Interaction

In this behavior, agents compete based on a new metric that we introduce as Intersection over Box area (IoB). The IoBs for two boxes,  $A$  and  $B$ , are calculated separately as:

$$IoB_{A|B} = \frac{\text{area}(A \cap B)}{\text{area}(A)}, \quad IoB_{B|A} = \frac{\text{area}(A \cap B)}{\text{area}(B)}$$

Attacking or cooperating with other agents depending on calculated strengths. The strength of an attack of  $A$  on  $B$  and defense of  $B$  against  $A$  are defined by:

$$S_{\text{attack}}(A, B) = \text{confidence}_A \times IoB_{B|A}, \quad S_{\text{defense}}(B, A) = \text{confidence}_B \times IoB_{A|B}$$

The decision rule is based on the difference between attack and defense strengths and a decision threshold ( $T$ ):

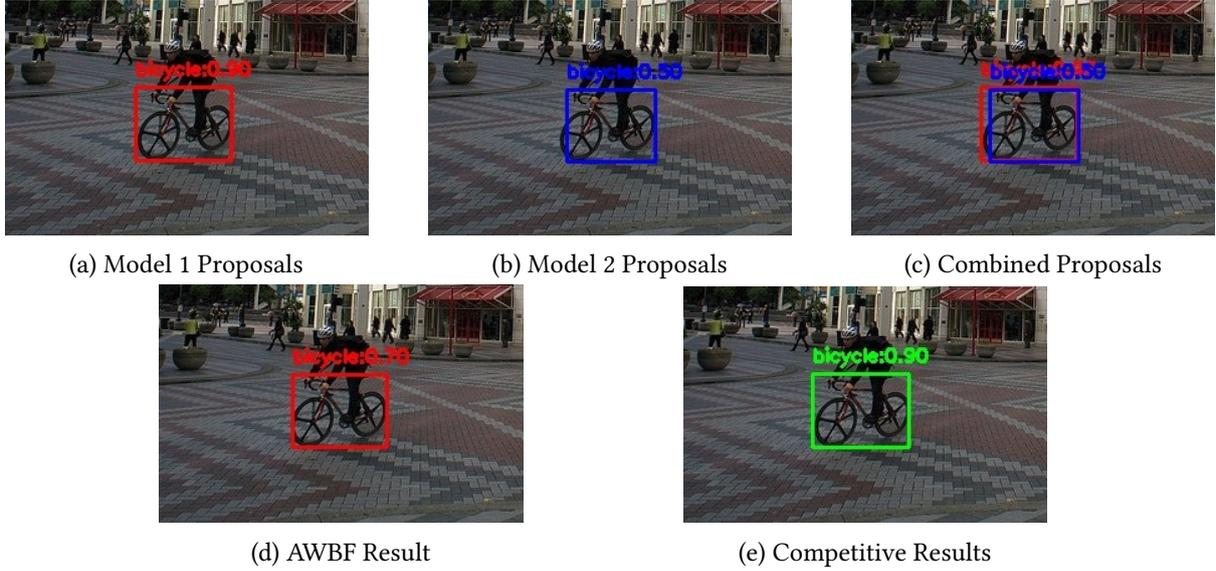
$$\text{Result}(A, B) = S_{\text{attack}}(A, B) - S_{\text{defense}}(B, A)$$

$$\text{winner determination} \begin{cases} \text{Result}(A, B) > T : & A \text{ wins and } B \text{ is removed} \\ \text{Result}(A, B) < -T : & B \text{ wins and } A \text{ is removed} \\ \text{otherwise} : & A \text{ and } B \text{ fuse using WBF} \end{cases}$$

The least case represent the area where agents can cooperate as their strengths are close. Threshold  $T$  can determine the level of cooperativeness, and thus the value  $(1 - T)$  refers to the competitiveness level. ( $T = 1$ ) indicates full cooperativeness settings, reverting to AWBF. Contrarily, ( $T = 0$ ) indicates full competitiveness unless attack and defense strengths are equal.

## Illustrative Example: Bounding Box Fusion for Bicycle Detection

To illustrate the AWBF and competitive behavior in action, we consider the detection of bicycles in image "138639" from COCO dataset using two ad-hoc models. (see Figure 2)



**Figure 2:** Visualization of Bounding Box Proposals and Fusion Results on COCO#138639 (images are cropped to emphasize the area of interest)

The bounding boxes from the two models are as follows:

- **Model 1:** {'box': [0.192, 0.752, 0.312, 0.873], 'score': 0.9, 'label': 2}
- **Model 2:** {'box': [0.203, 0.756, 0.314, 0.875], 'score': 0.5, 'label': 2}

The Intersection over Union for the bounding boxes is:  $\text{IoU} = \frac{\text{area of overlap}}{\text{area of union}} \approx 0.85$

If we apply the WBF method:

$$\text{WBF}_{\text{box}} = \frac{0.9 \cdot [0.192, 0.752, 0.312, 0.873] + 0.5 \cdot [0.203, 0.756, 0.314, 0.875]}{0.9 + 0.5}$$

$$\text{WBF}_{\text{box}} \approx [0.196, 0.754, 0.313, 0.874], \text{WBF}_{\text{score}} = \frac{0.9 + 0.5}{2} = 0.7$$

For competitive behavior, the attack and defence strengths rely on the calculation of the Intersection over Box values:

$$\text{IoB}_{B1|B2} = \frac{\text{area}(B1 \cap B2)}{\text{area}(B1)} \approx 0.89, \text{IoB}_{B2|B1} = \frac{\text{area}(B1 \cap B2)}{\text{area}(B2)} \approx 0.91$$

$$\text{Attack}_{B1|B2} = 0.9 \cdot 0.91 \approx 0.819, \text{Defense}_{B2|B1} = 0.5 \cdot 0.89 \approx 0.445$$

$$\text{Result} = \text{Attack}_{B1|B2} - \text{Defense}_{B2|B1} = 0.374$$

Given ( $T = 0.3$ ), the agent  $B1$  wins and  $B2$  is removed as ( $\text{Result} > T$ ). Increasing the  $T$  value to 4, the conflict result will fall into the cooperation range and thus we return back to AWBF.

## 5. Experimental Evaluation

To evaluate our methods, we conducted extensive tests using the COCO dataset. Our primary objective was to demonstrate the proof of concept without optimizing parameters or model weights beyond the default settings provided by the WBF code. Therefore, our results focus on comparing performance metrics rather than optimizing for maximum accuracy.

**Evaluation Metrics** The evaluation metrics used were those recommended and specified by COCO dataset. Namely, Average Precision and Average Recall:

- **Average Precision (AP)** Reveals the model’s ability to make accurate positive predictions. It is calculated at different Intersection over Union (IoU) thresholds.

- AP@[IoU=0.50:0.95]: This is the average AP over ten IoU thresholds (0.50 to 0.95 with a step size of 0.05).
- AP@0.50: This is the AP at an IoU threshold of 0.5.
- AP@0.75: This is the AP at an IoU threshold of 0.75.
- AP[small]: AP for small objects (area < 32 pixels).
- AP[medium]: AP for medium sized objects ( $32 \leq \text{area} \leq 96$  pixels).
- AP[large]: AP for large objects ( area  $\geq 96$  pixels).

- **Average Recall (AR)** Measuring the sensitivity by focusing on the model’s ability to correctly identify positive samples from the entire pool of positive instances.

- AR@[IoU=0.50:0.95]: This is the average recall over ten IoU thresholds (0.50 to 0.95 with a step size of 0.05).
- AR@0.50: This is the average recall at an IoU threshold of 0.5.
- AR@0.75: This is the average recall at an IoU threshold of 0.75.
- AR[small]: AR for small objects.
- AR[medium]: AR for medium objects.
- AR[large]: AR for large objects.

The results from test runs over the entire dataset are shown in the Table 1. Notably, the results demonstrate that AWBF outperforms individual models whose outputs were used in the fusion process. Although our results did not surpass those of the centralized WBF, they were mostly comparable. Specifically, our approach performed better than WBF on AP-small and AR@10 at an IoU of 0.5.

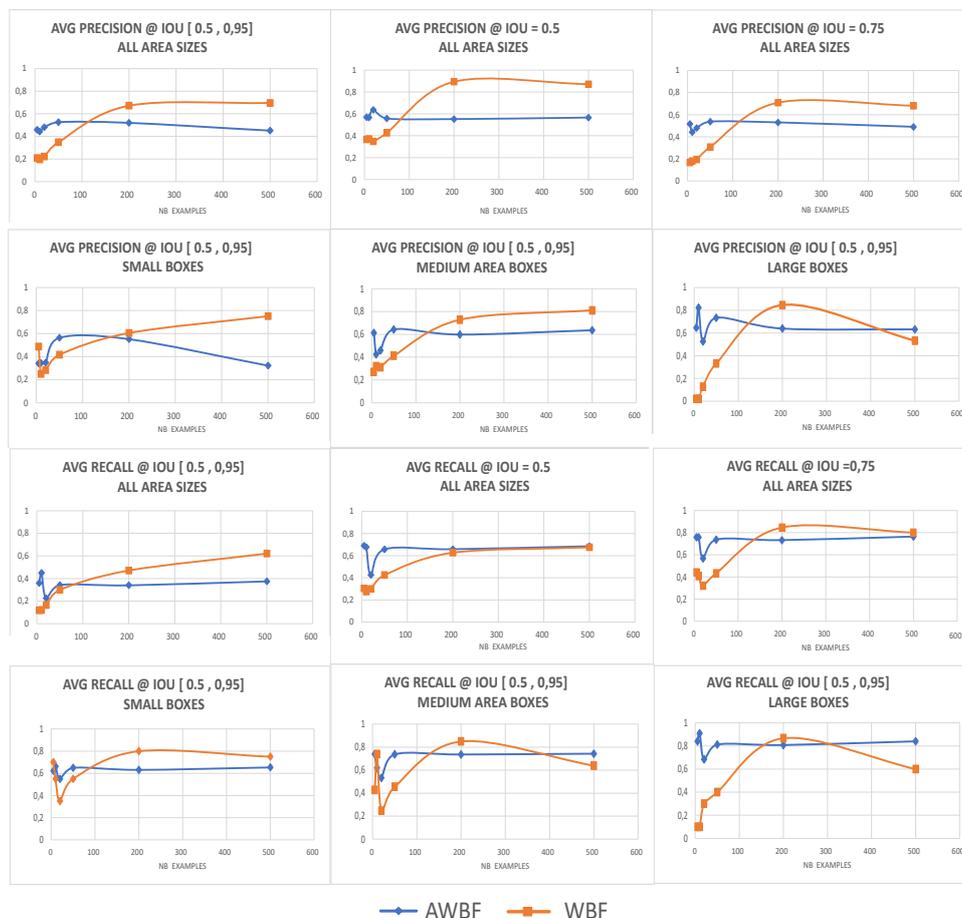
	Avg. Precision						Avg. Recall					
	@[0.5 , 0.95]	@[0.5]	@[0.75]	Small	Medium	Large	@[0.5 , 0.95]	@[0.5]	@[0.75]	Small	Medium	Large
EffNetB0	0.336	0.515	0.354	0.125	0.388	0.528	0.288	0.44	0.467	0.193	0.55	0.688
EffNetB0-m	0.335	0.516	0.351	0.129	0.389	0.524	0.288	0.441	0.467	0.198	0.55	0.687
EffNetB1	0.392	0.581	0.418	0.186	0.447	0.571	0.322	0.501	0.532	0.294	0.599	0.735
EffNetB1-m	0.392	0.581	0.417	0.184	0.447	0.571	0.323	0.502	0.531	0.279	0.602	0.735
EffNetB2	0.425	0.617	0.453	0.238	0.479	0.591	0.34	0.537	0.569	0.347	0.632	0.75
EffNetB2-m	0.426	0.617	0.454	0.24	0.481	0.593	0.341	0.537	0.569	0.358	0.634	0.748
EffNetB3	0.459	0.65	0.491	0.28	0.503	0.616	0.359	0.569	0.604	0.404	0.654	0.77
EffNetB3-m	0.455	0.646	0.487	0.282	0.494	0.618	0.357	0.566	0.6	0.412	0.65	0.766
EffNetB4	0.49	0.685	0.529	0.334	0.538	0.64	0.375	0.598	0.634	0.464	0.682	0.782
EffNetB4-m	0.488	0.684	0.524	0.33	0.533	0.642	0.373	0.596	0.633	0.468	0.68	0.783
EffNetB5	0.505	0.7	0.544	0.343	0.549	0.646	0.383	0.619	0.656	0.5	0.698	0.791
EffNetB5-m	0.502	0.696	0.539	0.335	0.546	0.645	0.379	0.614	0.651	0.484	0.692	0.789
EffNetB6	0.513	0.705	0.555	0.352	0.556	0.652	0.387	0.626	0.664	0.505	0.703	0.795
EffNetB6-m	0.511	0.701	0.551	0.341	0.555	0.654	0.384	0.623	0.66	0.489	0.704	0.805
EffNetB7	0.521	0.71	0.562	0.37	0.562	0.66	0.39	0.633	0.671	0.517	0.711	0.801
EffNetB7-m	0.519	0.71	0.558	0.364	0.562	0.659	0.388	0.63	0.668	0.509	0.71	0.803
DetRS	0.515	0.71	0.654	0.318	0.565	0.676	0.384	0.628	0.671	0.479	0.723	0.828
DetRS-m	0.515	0.707	0.564	0.316	0.563	0.677	0.384	0.629	0.673	0.486	0.721	0.834
resnet50	0.496	0.697	0.538	0.299	0.543	0.656	0.378	0.607	0.64	0.457	0.686	0.8
resnet50-m	0.496	0.694	0.535	0.296	0.545	0.657	0.379	0.61	0.642	0.464	0.689	0.799
yolo	0.5	0.678	0.546	0.336	0.544	0.644	0.381	0.628	0.688	0.533	0.734	0.826
WBF	<b>0.673</b>	<b>0.894</b>	<b>0.709</b>	<b>0.605</b>	<b>0.731</b>	<b>0.846</b>	<b>0.471</b>	<b>0.627</b>	<b>0.846</b>	<b>0.8</b>	<b>0.85</b>	<b>0.867</b>
AWBF	0.61	0.66	0.625	0.61	0.766	0.675	0.395	0.676	0.745	0.664	0.706	0.819

**Table 1**  
Benchmarking on COCO dataset

## 5.1. WBF Performance on Different Dataset Sizes:

When running experiments on subsets of the COCO dataset with different sizes, we observed that the centralized WBF method performs better with larger datasets but shows reduced efficiency on smaller datasets (see Figure 3). This can be explained by several factors:

- **Law of Large Numbers:** As the dataset size increases, the averaging process tends to smooth out random errors and fluctuations, leading to improved performance for the centralized WBF method.
- **Error Compensation:** With more data points, errors in individual detections can compensate for each other, leading to more accurate fusion results.
- **Increased Data Redundancy:** Larger datasets contain more redundant information, reinforcing correct detections and diluting the impact of incorrect ones.



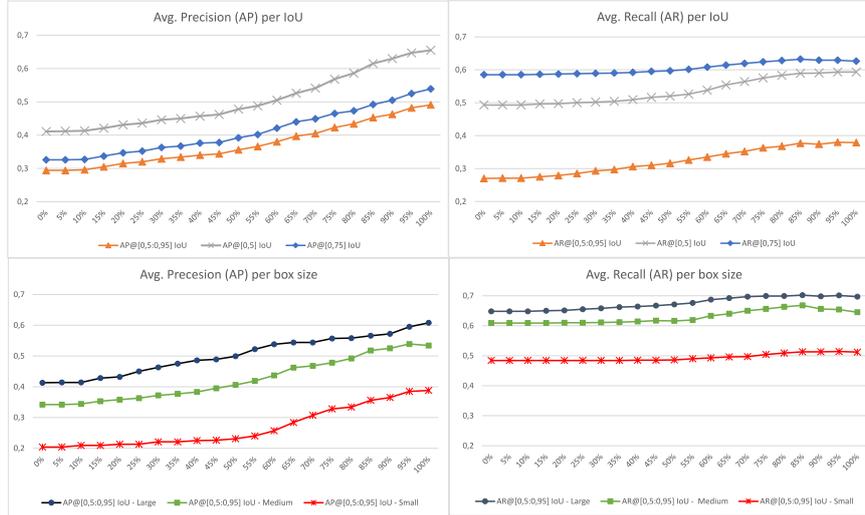
**Figure 3:** COCO evaluation metrics evolution as function of the size of test set - Evaluating the AWBF method performance on subsets of COCO dataset

## 5.2. AWBF Performance on Different Dataset Sizes:

The AWBF method exhibited more robust and stable performance across varying dataset sizes as Figure 3 shows, which can be attributed to the distributed processing and the redundancy: Each agent processes bounding boxes independently and in parallel, reducing the impact of individual errors and improving overall robustness. Also, Each agent's localized decision-making can lead to better performance, especially in smaller datasets where individual detections have a higher impact.

### 5.3. Competitive Behavior Experiments:

We also evaluated the competitive behavior using the default parameters. While the initial results did not match the quality of WBF, they demonstrated the potential for diverse agent behaviors. By adjusting the value of  $T$ , which controls the level of cooperativeness ( $1 - \text{competitiveness}$ ). We conducted multiple tests on a subset of 500 COCO images, varying the competitiveness level. We observed an interesting trend, as the results showed that increasing competitiveness improved precision (see Figure 4).



**Figure 4:** Precision and recall values evolution with the increase of competition level (decrease in cooperation threshold  $T$ ) in the agent behavior

AP increased with higher competitiveness, likely because competition removed lower-scoring boxes, reducing false positives and improving precision. Recall remained stable as even with fewer boxes, sufficient accurate boxes were retained.

To summarize, these evaluations demonstrated the flexibility and potential of integrating MAS into object detection workflows. While the competitive agent behavior requires further optimization, the initial results validate our approach and open avenues for more sophisticated multi-agent behaviors in future work.

## 6. Conclusion

In this work, we presented a proof-of-concept implementation integrating MAS into object detection workflows, specifically focusing on improving bounding box predictions, an essential component of autonomous vehicle perception systems. By leveraging the decentralized processing capabilities of MAS, we demonstrated two distinct agent behaviors: Decentralized (Agentified) Weighted Boxes Fusion and Competitive Interaction. Our experimental evaluation using the COCO dataset showed that while the decentralized WBF approach performed comparably to the centralized WBF, the competitive behavior illustrated the potential for further optimization and innovation in agent-based object detection systems. The results indicate that MAS can offer robust and adaptable solutions for object detection tasks, particularly in dynamic and complex environments like AV perception and intelligent transportation systems. Future work will focus on refining agent behaviors, enhancing system scalability, and integrating more advanced machine learning models to further improve performance and adaptability for AV applications.

## Acknowledgments

This work is funded by the French National Research Agency as part of the MultiTrans project under reference ANR-21-CE23-0032.

## References

- [1] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39 (2017) 1137–1149. doi:10.1109/TPAMI.2016.2577031.
- [2] R. Solovyev, W. Wang, T. Gabruseva, Weighted boxes fusion: Ensembling boxes from different object detection models, *Image and Vision Computing* 107 (2021) 104117. URL: <https://www.sciencedirect.com/science/article/pii/S0262885621000226>. doi:<https://doi.org/10.1016/j.imavis.2021.104117>.
- [3] O. Vinyals, I. Babuschkin, J. Chung, M. Mathieu, M. Jaderberg, W. M. Czarnecki, A. Dudzik, A. Huang, P. Georgiev, R. Powell, et al., Alphastar: Mastering the real-time strategy game starcraft ii, *DeepMind blog 2* (2019) 20.
- [4] S.-Y. Yang, H.-Y. Cheng, C.-C. Yu, Real-time object detection and tracking for unmanned aerial vehicles based on convolutional neural networks, *Electronics* 12 (2023) 4928. doi:10.3390/electronics12244928.
- [5] G. Weiß, A multiagent perspective of parallel and distributed machine learning, in: *Proceedings of the Second International Conference on Autonomous Agents, AGENTS '98*, Association for Computing Machinery, New York, NY, USA, 1998, p. 226–230. URL: <https://doi.org/10.1145/280765.280806>. doi:10.1145/280765.280806.
- [6] J. E. D. Narinx, A Real-Time Multi-Camera Depth Estimation ASIC with Custom On-Chip Embedded DRAM, Ph.D. thesis, École Polytechnique Fédérale de Lausanne, Lausanne, 2019. URL: <http://infoscience.epfl.ch/record/273168>. doi:<https://doi.org/10.5075/epfl-thesis-7163>.
- [7] S. Liu, C. Lyu, H. Gong, Vehicle video surveillance system based on image fusion and parallel computing, *International Journal of Circuit Theory and Applications* 49 (2020) 1532 – 1547. doi:10.1002/cta.2907.
- [8] M. Akil, L. Perroton, Special issue on parallel computing for real-time image processing, *Journal of Real-Time Image Processing* 6 (2011) 1–2. doi:10.1007/s11554-011-0192-y.
- [9] L. Ma, K. Xue, P. Wang, Distributed multiagent control approach for multitarget tracking, *Mathematical Problems in Engineering* 2015 (2015) 1–10. doi:10.1155/2015/903682.
- [10] R. Caballero-Águila, A. Hermoso-Carazo, J. Linares-Pérez, Networked distributed fusion estimation under uncertain outputs with random transmission delays, packet losses and multi-packet processing, *Signal Processing* 156 (2019) 71–83.
- [11] Y. Liang, K. Zhou, C. Wu, Dynamic task allocation method for heterogenous multiagent system in uncertain scenarios of agricultural field operation, *Journal of Physics: Conference Series* 2356 (2022). doi:10.1088/1742-6596/2356/1/012049.
- [12] Q. Shen, P. Shi, J. Zhu, S. Wang, Y. Shi, Neural networks-based distributed adaptive control of nonlinear multiagent systems, *IEEE Transactions on Neural Networks and Learning Systems* 31 (2020) 1010–1021. doi:10.1109/TNNLS.2019.2915376.
- [13] L. Wang, R. Li, J. Sun, X. Liu, L. Zhao, S. H. Soon, C. K. Quah, B. Tandianus, Multi-view fusion-based 3d object detection for robot indoor scene perception, *Sensors (Basel, Switzerland)* 19 (2019). doi:10.3390/s19194092.
- [14] R. de Azevedo, M. Cintuglu, T. Ma, O. Mohammed, Multiagent-based optimal microgrid control using fully distributed diffusion strategy, *IEEE Transactions on Smart Grid* 8 (2017) 1997–2008. doi:10.1109/TSG.2016.2587741.
- [15] S. Zhong, M. Wei, S. Gong, K. Xia, Y. Fu, Q. Fu, H. Yin, Behavior prediction for unmanned driving based on dual fusions of feature and decision, *IEEE Transactions on Intelligent Transportation Systems* 22 (2021) 3687–3696. doi:10.1109/TITS.2020.3037926.
- [16] N. Kaabouch, W.-C. Hu, T. Niemi, J. Kommeri, A.-P. Hameri, Improving energy-efficiency of scientific computing clusters, in: *Energy-Aware Systems and Networking for Sustainable Initiatives*, IGI Global, 2012, pp. 1–19. doi:10.4018/978-1-4666-1842-8.ch001.
- [17] P. Korus, J. Huang, Multi-scale analysis strategies in prnu-based tampering localization, *IEEE Transactions on Information Forensics and Security* 12 (2017) 809–824. doi:10.1109/TIFS.2016.

2636089.

- [18] P. Stone, M. Veloso, Multiagent systems: A survey from a machine learning perspective, *Autonomous Robots* 8 (2000) 345–383.
- [19] C. Gao, X. He, H. Dong, H. Liu, G. Lyu, A survey on fault-tolerant consensus control of multi-agent systems: trends, methodologies and prospects, *International Journal of Systems Science* 53 (2022) 2800–2813. URL: <https://doi.org/10.1080/00207721.2022.2056772>. doi:10.1080/00207721.2022.2056772. arXiv:<https://doi.org/10.1080/00207721.2022.2056772>.
- [20] M. Lujak, S. Giordani, A. Omicini, S. Ossowski, Scalable distributed decision-making and coordination in large and complex systems: Methods, techniques, and models, *Complexity* 2020 (2020) 1–3.
- [21] J. Wang, M. Zhu, D. Sun, B. Wang, W. Gao, H. Wei, Mcf3d: Multi-stage complementary fusion for multi-sensor 3d object detection, *IEEE Access* 7 (2019) 90801–90814. doi:10.1109/ACCESS.2019.2927012.
- [22] X. Qian, S. Lin, G. Cheng, X. Yao, H. Ren, W. Wang, Object detection in remote sensing images based on improved bounding box regression and multi-level features fusion, *Remote Sensing* 12 (2020). URL: <https://www.mdpi.com/2072-4292/12/1/143>. doi:10.3390/rs12010143.
- [23] Y. Zhang, H. Wu, 3d object detection based on multi-view adaptive fusion, in: *2022 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC), 2022*, pp. 743–748. doi:10.1109/IPEC54454.2022.9777488.
- [24] Z. Liu, X. Ye, Z. Zou, X. He, X. Tan, E. Ding, J. Wang, X. Bai, Multi-modal 3d object detection by box matching, 2023. arXiv:2305.07713.
- [25] A. Choksuriwong, C. Rosenberger, W. Smari, Multi-agents system for image understanding, in: *International Conference on Integration of Knowledge Intensive Multi-Agent Systems, 2005.*, 2005, pp. 149–154. doi:10.1109/KIMAS.2005.1427070.
- [26] M. Jiang, T. Hai, Z. Pan, H. Wang, Y. Jia, C. Deng, Multi-agent deep reinforcement learning for multi-object tracker, *IEEE Access* 7 (2019) 32400–32407. doi:10.1109/ACCESS.2019.2901300.
- [27] A. Fekir, N. Benamrane, Multi agent system for boundary detection and object tracking in image sequence based on active contours, *Multiagent Grid Syst.* 11 (2015) 81–93. URL: <https://doi.org/10.3233/MGS-150230>. doi:10.3233/MGS-150230.
- [28] G. Vincent, E. Patten, G. Ohmes, N. Couch, Multi-agent system perception with stereovision, in: *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 2, SIGCSE 2023, Association for Computing Machinery, New York, NY, USA, 2023*, p. 1235. URL: <https://doi.org/10.1145/3545947.3573289>. doi:10.1145/3545947.3573289.
- [29] F. Tabib Mahmoudi, F. Samadzadegan, P. Reinartz, Object oriented image analysis based on multi-agent recognition system, *Computers & Geosciences* 54 (2013) 219–230. URL: <https://www.sciencedirect.com/science/article/pii/S0098300412004141>. doi:<https://doi.org/10.1016/j.cageo.2012.12.007>.
- [30] G. Coulouris, J. Dollimore, T. Kindberg, *Distributed Systems: Concepts and Design*, 5 ed., Addison-Wesley, 2011.
- [31] H. P. Nii, The blackboard model of problem solving and the evolution of blackboard architectures, *AI Magazine* 7 (1986) 38. URL: <https://ojs.aaai.org/aimagazine/index.php/aimagazine/article/view/537>. doi:10.1609/aimag.v7i2.537.
- [32] M. Wooldridge, *An Introduction to MultiAgent Systems*, John Wiley & Sons, 2009.
- [33] S. Russell, P. Norvig, *Artificial Intelligence: A Modern Approach*, 3 ed., Prentice Hall, 2010.