Multiple Large-Language-Models Consensus for Object Detection—A Survey

Marcin Iwanowski * and Marcin Gahbler

Posted Date: 13 November 2025

doi: 10.20944/preprints202511.0879.v1

Keywords: object detection; data fusion, consensus learning; large language models; vision–language models; ensemble methods

 doi:10.20944/preprints202511.0879.v1

Article

Multiple Large-Language-Models Consensus for Object Detection—A Survey

Marcin Iwanowski [1,2,*], **Marcin Gahbler** [1]

[1] Institute of Engineering and Technology, Faculty of Physics and Astronomy, ul. Wileńska 7, 87-100 Toruń, POLAND
[2] Institute of Control and Industrial Electronics, Faculty of Electrical Engineering, Warsaw University of Technology; ul. Koszykowa 75, 00-662 Warsaw, POLAND
* Correspondence: iwanowski@fizyka.umk.pl

Abstract

The rapid development of large language models (LLMs) and vision–language models (VLMs) has enabled instruction-driven visual understanding, where a single foundation model can recognize and localize arbitrary objects from natural-language prompts. However, predictions from individual models remain inconsistent – LLMs hallucinate nonexistent entities, while VLMs exhibit limited recall and unstable calibration compared to purpose-trained detectors. To address these limitations, a new paradigm termed Multiple Large–Language–Model Consensus (Multi-LLM Consensus) has emerged. In this approach, multiple heterogeneous LLMs or VLMs process a shared visual–textual instruction, generate independent structured outputs (bounding boxes and categories). Next, their results are merged through consensus mechanisms. This cooperative inference improves spatial accuracy and semantic correctness, making it particularly suitable for generating high-quality training datasets for fast real-time object detectors. This survey provides a comprehensive overview of Multi-LLM Consensus for object detection. We formalize the concept, review related literature on ensemble reasoning and multimodal perception, and categorize existing methods into four frameworks: prompt-level, reasoning-to-detection, box-level, and hybrid consensus. We further analyze fusion algorithms, evaluation metrics, and benchmark datasets, highlighting their strengths and limitations. Finally, we discuss open challenges—vocabulary alignment, uncertainty calibration, computational efficiency, and bias propagation—and identify emerging trends such as consensus-aware training, structured reasoning, and collaborative perception ecosystems.

Keywords: object detection; data fusion, consensus learning; large language models; vision–language models; ensemble methods

1. Introduction

Over the past decade, object detection has undergone a remarkable transformation – from hand-crafted feature pipelines to deep convolutional networks, passing through region-based two-stage and one-stage models, anchor-free detectors, and, most recently, transformer-based architectures and open-vocabulary models. In parallel, the *semantic scope* of detectors evolved, initially limited to close-vocabulary models; recently, they no longer operate over fixed class lists but can respond to arbitrary natural-language descriptions. The emergence of large language models (LLMs) and vision–language models (VLMs) has further blurred the line between perception and reasoning, introducing a new era of *instruction-driven visual understanding* [1–4].

Models such as GPT-4V, LLaVA-Next, Qwen-VL, and InternVL demonstrate that a single foundation model can jointly interpret text and images, recognize novel objects, and even produce structured outputs such as bounding boxes or segmentation masks from natural-language prompts. However, despite their versatility, individual models remain imperfect. They are prone to hallucination – reporting non-existent objects [5], and to omissions, missing small or rare categories. Visual Language

 doi:10.20944/preprints202511.0879.v1

models also exhibit limited recall and unstable calibration compared with dedicated detectors [6]. This variability highlights a persistent challenge in foundation-model perception: the lack of consistency and reliability across diverse visual contexts.

A promising direction to overcome these limitations is to combine multiple large models into a cooperative system that reaches agreement through structured reasoning and data fusion. Instead of relying on a single LLM or VLM, several models process the same image and prompt in parallel, each producing an independent prediction in a standardized format—typically a list of objects, bounding boxes, and confidence scores. These structured outputs are then merged using consensus algorithms such as Weighted Boxes Fusion (WBF) [7], Agglomerative Late Fusion (ALFA) [8], or probabilistic ensembling (ProbEn) [9]. The result is a unified detection map that aggregates the complementary strengths of individual models while suppressing hallucinated or inconsistent results (see Figure 1 for the approach pipeline).

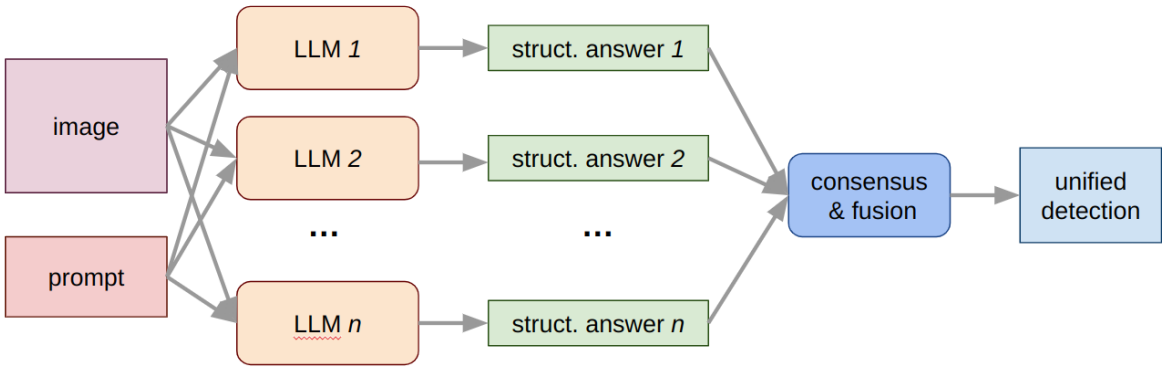

Figure 1. Multiple LLM consensus pipeline for object detection.

This general paradigm, referred to here as *Multi–Large–Language–Model Consensus* (Multi-LLM Consensus), extends classical ensemble learning into the image domain. It leverages model diversity, differences in architecture, pre-training data, and reasoning style to improve robustness, interpretability, and trustworthiness. Analogous to ensemble methods in traditional machine learning, consensus among independent LLMs can reduce variance, correct individual biases, and increase calibration reliability. Consensus in multimodal settings operates across both linguistic and spatial dimensions: models must agree not only on *what* objects are present but also on *where* they are located.

The motivation for this survey is twofold. First, it aims to systematically review the growing body of research that explores how multiple LLMs and VLMs can collaborate to achieve more accurate and semantically grounded object detection. Secondly, its goal is to establish a unified conceptual and methodological framework that bridges reasoning-level agreement and detection-level fusion.

This paper makes the following contributions:

- **Conceptual unification.** We formalize the notion of Multi-LLM Consensus for object detection and relate it to ensemble and consensus learning traditions in artificial intelligence.
- **Comprehensive taxonomy.** We categorize existing consensus paradigms into prompt-level, reasoning-to-detection, box-level, and hybrid designs, linking them with representative algorithms such as MoA [10], LLM-Blender [11], DetGPT [12], and ContextDET [13].
- **Survey of fusion algorithms.** We summarize data fusion techniques—including NMS, Soft-NMS, WBF, ALFA, and ProbEn – and discuss how they extend to multimodal, reasoning-guided detection.
- **Evaluation framework.** We outline appropriate datasets, metrics, and benchmarks for assessing consensus-based systems, emphasizing calibration, hallucination reduction, and inter-agent diversity.

 doi:10.20944/preprints202511.0879.v1

- **Challenges and outlook.** We identify key open problems: vocabulary alignment, calibration, efficiency, and bias—and highlight emerging research trends such as consensus-aware training and collaborative perception ecosystems.

The remainder of this article is organized as follows. Section 2 reviews the foundations of object detection, ensemble learning, and vision–language modeling. Section 3 examines LLM-driven detection frameworks such as DetGPT and LLMDet. Section 4 introduces the taxonomy of Multi-LLM Consensus approaches, followed by Section 5, which details data fusion algorithms and implementation strategies. Section 6 presents evaluation protocols and benchmarking practices, and Section 7 outlines challenges and future research directions. Finally, Section 8 concludes with a discussion on the broader implications of consensus-based perception for trustworthy multimodal AI.

2. Background of Object Detection

Object detection, the task of locating and classifying objects within an image, has been one of the central challenges of computer vision for over two decades. This field has evolved through several primary stages, each characterized by distinct model architectures and training paradigms. Modern object detection has evolved from region-proposal CNNs to anchor-free and transformer-based methods, and now includes open-vocabulary and language-guided systems. Each stage improved either speed, accuracy, or semantic flexibility. The recent arrival of multimodal foundation models introduces powerful reasoning capabilities but also new challenges in consistency and calibration. Figure 2 summarizes this historical evolution.

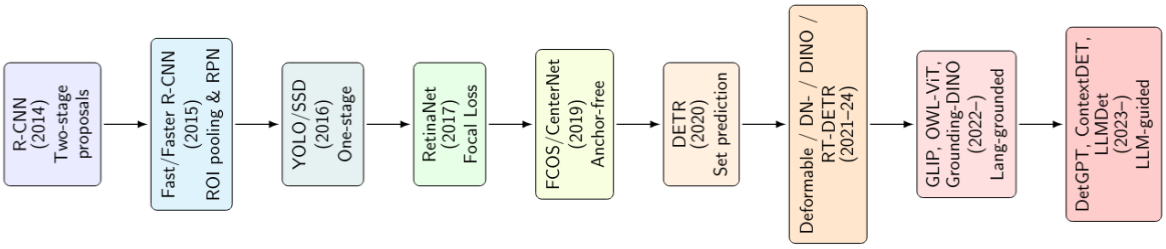

Figure 2. Evolution of deep learning–based object detection.

2.1. Closed Vocabulary Methods

The modern era of deep-learning-based detection began with the Region-based Convolutional Neural Network (**R-CNN**) [14]. The R-CNN decomposed detection into two steps: first, it generates region proposals using a hand-crafted algorithm such as Selective Search, and next, it classifies each proposed region with a CNN. Although accurate, this two-stage approach was computationally expensive, requiring thousands of CNN forward passes per image. Subsequent work focused on improving its efficiency. **Fast R-CNN** [15] introduced Region-of-Interest (ROI) pooling to extract features for all proposals from a shared feature map, dramatically reducing redundancy. **Faster R-CNN** [16] further unified the pipeline by adding a Region Proposal Network (RPN), which learned to generate candidate regions directly from convolutional features. This design made end-to-end training feasible and established the canonical two-stage architecture that dominated research for years. Later extensions, such as Feature Pyramid Networks (**FPN**) [17], improved multi-scale reasoning by building hierarchical feature maps, enabling robust detection of both small and large objects.

To further improve inference speed, researchers merged the pipeline into a single pass. **YOLO** ("You Only Look Once") [18] and **SSD** (Single Shot MultiBox Detector) [19] reframed detection as dense regression: a single network simultaneously predicted bounding boxes and class probabilities across a grid of pre-defined anchor boxes (one-stage detectors). While these models achieved real-time performance, their accuracy initially lagged behind two-stage methods. This gap was closed mainly by **RetinaNet** [20], which introduced the *Focal Loss* to mitigate the imbalance between foreground and background samples. Since then, one-stage detectors have become the preferred choice for applications that require high throughput, such as robotics or video analytics.

 doi:10.20944/preprints202511.0879.v1

The following conceptual step was to abandon anchor boxes. Methods like **CornerNet** [21] detected objects by locating paired keypoints (corners), while **CenterNet** [22] and **FCOS** [23] predicted object centers and corresponding box sizes directly. Anchor-free formulations simplified training, reduced the number of hyperparameters, and improved generalization to unseen scales or aspect ratios.

A significant paradigm shift came with **DETR** (DEtection TRansformer) [24], which redefined detection as a set-prediction problem using a transformer encoder–decoder architecture. Instead of passing through intermediary steps, DETR directly predicted a fixed-size set of object queries and matched them to ground truth. Although conceptually elegant, vanilla DETR suffered from slow convergence and poor recall of small objects. Several improved variants addressed these limitations: **Deformable DETR** [25] used sparse attention focused on relevant regions; **DN-DETR** [26] introduced denoising queries for stable training; **DINO** [27] refined the query design and optimization strategy; and **RT-DETR** [28,29] achieved real-time inference without sacrificing accuracy.

2.2. Open Vocabulary and Language-Guided Methods

Detectors, mentioned in the previous section, rely on a closed vocabulary of classes defined prior to actual training. They cannot recognize novel objects beyond that fixed label set. This limitation inspired the development of *open-vocabulary* detection, which allows models to generalize to unseen categories described by natural language.

The breakthrough enabling this transition was the introduction of contrastive vision–language pretraining in **CLIP** [30]. CLIP jointly trained image and text encoders to align visual and linguistic embeddings, creating a shared semantic space in which textual prompts could represent arbitrary object concepts. Subsequent detectors leveraged this principle by integrating textual conditioning into standard detection architectures.

Among the earliest were **GLIP** [31], which unified grounding and detection objectives by aligning region proposals with textual phrases, and **OWL-ViT** [32,33], which used a Vision Transformer backbone for zero-shot open-vocabulary detection. **Grounding-DINO** [34,35] extended these ideas, combining a strong objectness prior with text embeddings to achieve state-of-the-art grounding accuracy, while **YOLO-World** [36] adapted the approach for real-time performance.

Table 1 summarizes these systems and their characteristics along with classic approaches.

Table 1. Comparison of representative object detection models.

System	Type	Open-Voc.	Inputs	Outputs	Highlights
Classical detectors					
Faster R-CNN (2015) [16]	Two-stage CNN	×	Image	Boxes + classes	RPN + ROI pooling; strong accuracy
RetinaNet (2017)[20]	One-stage CNN	×	Image	Boxes + classes	Focal Loss for class imbalance
FCOS (2019) [23]	One-stage anchor-free	×	Image	Boxes + classes	Per-pixel center + distances
DETR (2020) [24]	Transformer set prediction	×	Image	Boxes + classes	Bipartite matching, simple pipeline
Deformable DETR (2021) [25]	Transformer	×	Image	Boxes + classes	Sparse attention, faster convergence
RT-DETR [28] / v2 (2023/24) [29]	Transformer (real-time)	×	Image	Boxes + classes	Real-time set-prediction detector
Open-vocabulary / language-grounded detectors					
CLIP (2021) [30]	Vision–language encoder	✓ (zero-shot cls.)	Image + text	Global logits (image–text)	Contrastive pretraining for OV features
GLIP (2022) [31]	OV detector (grounded pretrain)	✓	Image + text	Boxes + text-aligned labels	Unifies grounding + detection
OWL-ViT [32] / OWLv2 (2022/23) [33]	OV ViT detector	✓	Image + text	Boxes + labels	Zero-shot detection via text queries
Grounding-DINO (2023) [34]	OV transformer detector	✓	Image + text	Boxes + phrases	Strong grounding; open-set
Grounding-DINO 1.5 (2024) [35]	OV detector (improved)	✓	Image + text	Boxes + phrases	Better edge/open-set perf.
YOLO-World (2024) [36]	OV detector (real-time)	✓	Image + text	Boxes + labels	Real-time open-vocabulary
Multimodal LLMs (VLMs / MLLMs)					
LLaVA-Next (2024) [2]	MLLM (vision–language)	✓ (reasoning)	Image + text	Free-form text; coords via prompting	Strong VQA/analysis; instr. tuned
Qwen-VL (2024) [37]	MLLM (vision–language)	✓ (reasoning)	Image + text	Free-form; OCR; grounding-like	Versatile perception/localization
InternVL (2024) [3]	Vision–language foundation	✓ (reasoning)	Image + text	Free-form; grounding-style outputs	Scaled multimodal pretraining
LLM-guided detection / reasoning-to-detection					
DetGPT (2023) [12]	LLM-planned detection	✓ (via OV det.)	Image + instruction	Boxes + labels	LLM plans; OV det. localizes
ContextDET (2023) [13]	LLM context reasoning	✓	Image + context text	Boxes + labels	Context-aware cues for det.
LaMI-DETR (2024) [38]	Language-guided DETR	✓	Image + text	Boxes + labels	Instruction-level guidance
LLMDet (2025) [39]	LLM-supervised training	✓	Image + text (LLM labels)	Boxes + labels	Pseudo-labels from LLM for OV det.
Consensus / fusion (spatial stage)					
Soft-NMS (2017) [40]	Spatial fusion	—	Boxes & scores	Fused boxes	Score decay instead of suppression
WBF (2019/2021) [41]	Spatial fusion	—	Boxes & scores	Weighted fused boxes	Confidence-weighted averaging
ALFA (2019) [8]	Spatial fusion (clustering)	—	Boxes & scores	Cluster-based fusion	Agglomerative late fusion
ProbEn (2022) [9]	Probabilistic ensemble	—	Boxes & scores	Uncertainty-aware fusion	Bayesian modeling of boxes

The fusion of language and vision naturally led to the emergence of *Vision–Language Models (VLMs)* and later *Multimodal Large Language Models (MLLMs)*, such as GPT-4V [1], LLaVA-Next [2], Qwen-VL [37], and InternVL [3]. These models combine a visual encoder (often derived from CLIP or EVA) with an autoregressive language decoder, enabling them to process images and text jointly.

 doi:10.20944/preprints202511.0879.v1

They can describe scenes, answer visual questions, or even produce structured outputs such as JSON bounding boxes through appropriate prompting.

However, while MLLMs demonstrate impressive reasoning capabilities, their predictions can be inconsistent or incomplete [5,42]. They may hallucinate objects, omit subtle details, or vary across prompt phrasing. This inconsistency motivates the exploration of *consensus-based* approaches, in which multiple models contribute complementary perspectives that are later reconciled into a unified detection result—a topic further discussed in Sections 4–5.

3. LLM-Guided and Reasoning-Driven Object Detection

Open-vocabulary detectors such as GLIP, OWL-ViT, and Grounding-DINO (see Table 1) already link vision and language through textual prompts. However, their reasoning capability remains limited: they match visual regions to text embeddings but do not *understand* relationships, context, or complex instructions. For example, a user may ask, "*Find all objects that could be used for cooking, excluding plates*." Such a request requires logical reasoning and contextual interpretation beyond pure vision-language alignment.

This gap has led to the emergence of *LLM-guided object detection* – a family of approaches where a large language model (LLM) interprets or generates structured instructions that direct the detector. Instead of replacing visual encoders, the LLM operates as a high-level planner: it parses natural-language input, deduces what needs to be detected, and produces structured queries or semantic categories for a downstream vision module. This hybrid division of labor mirrors the way humans process scenes, where reasoning is followed by perception.

3.1. LLM-Guided Approaches

DetGPT [12] was among the first systems to formalize this paradigm. It couples an instruction-tuned LLM (such as Vicuna or GPT-4) with a visual detector like Grounding-DINO. The process unfolds in two stages:

1. The LLM receives a natural-language instruction and interprets it to produce a structured plan—a list of target object types or phrases, possibly with attributes (e.g., "detect red cars," "count people sitting at a table").
2. The plan is executed by an open-vocabulary detector, which performs localization for each text query and returns bounding boxes and confidence scores.

This design effectively transforms the LLM into a *semantic controller* that orchestrates the visual backend. Because the LLM can reason about context and task intent, DetGPT generalizes across diverse visual instructions: from generic object finding to compositional reasoning ("find the largest animal near the tree"). Furthermore, it allows flexible integration of multiple detectors, an idea that naturally extends to consensus-based pipelines discussed in Section 4.

ContextDET [13] advances the concept of LLM-driven guidance by injecting textual context directly into the visual decoding process. Instead of treating the LLM and the detector as strictly separate modules, ContextDET enables a two-way information exchange: the LLM generates contextual clues (e.g., "objects likely to appear in a kitchen") that modulate the attention maps of the visual encoder. This joint optimization improves both precision and recall, especially in cluttered scenes or when object boundaries are ambiguous.

ContextDET also illustrates a broader research direction—*contextual grounding*—in which linguistic priors constrain spatial predictions. By enriching visual tokens with semantic cues from language, the model learns to focus on relevant regions even in the absence of explicit annotations. The method can operate in zero-shot settings, bridging perception and reasoning in a more interpretable manner.

A related approach, **LaMI-DETR** (Language-Model-Integrated DETR) [38], explores tighter fusion between transformer-based detection and language understanding. Here, the LLM's output (for example, parsed query tokens or reasoning traces) is directly incorporated into the transformer decoder as conditioning information. Unlike DetGPT, which sequences reasoning and detection, LaMI-DETR

 doi:10.20944/preprints202511.0879.v1

blends them at the feature level: textual embeddings guide object queries through cross-attention. This design yields better performance in compositional reasoning tasks and supports *instruction-based* detection, where the LLM can modify how the detector prioritizes objects depending on task goals.

LLMDet [39] represents another key step: instead of using an LLM only at inference time, it leverages it during training. The LLM generates pseudo-labels, descriptions, or relational constraints for unlabeled images, effectively augmenting the training data with linguistic supervision. This approach turns LLMs into "data generators" that help detectors learn open-vocabulary associations without explicit human annotation. In practice, LLMDet can train competitive open-vocabulary detectors solely from LLM-synthesized captions and class hierarchies, drastically reducing labeling cost.

Table 1 lists key LLM-driven systems and their design differences. DetGPT and ContextDET rely on inference-time reasoning, whereas LLMDet introduces LLM-based supervision during training. LaMI-DETR, in turn, integrates LLM-derived semantics directly into the detector architecture.

- **Modularity:** DetGPT and ContextDET maintain modular design (LLM + detector), making them easy to pair with different open-vocabulary backbones.
- **Joint learning:** LaMI-DETR blurs the line between reasoning and perception, potentially improving performance but at a higher computational cost.
- **Supervision:** LLMDet demonstrates that LLMs can generate rich supervision signals, aligning vision models with linguistic concepts.

Together, these methods illustrate a spectrum of integration strategies—from loose coupling (LLM as planner) to tight multimodal fusion (shared latent space).

3.2. Evaluation and Challenges

Evaluating reasoning-guided detection involves both standard detection metrics and reasoning-aware benchmarks. Datasets such as RefCOCO, RefCOCOg, and Flickr30K Entities (see Table 2) are particularly suited because they test phrase-level grounding rather than fixed labels. Recent multimodal benchmarks—MMBench [43], MME [44], and MMBench-Plus [45]—also include instruction-based detection and compositional reasoning tasks, allowing unified evaluation of both reasoning accuracy and spatial localization.

Table 2. Overview of datasets and benchmarks relevant to object detection, open-vocabulary learning, multimodal reasoning, and hallucination evaluation.

Dataset / Benchmark	Domain / Focus	Type of Task	Scale
COCO [46]	General objects, everyday scenes	Detection, segmentation	118k images, 80 classes
LVIS [47]	Long-tailed object categories	Instance segmentation, detection	1.2M instances, 1200+ classes
ODinW / ELEVATER [48]	Open-world evaluation across domains	Open-vocabulary detection	20 domains, 180k images
RefCOCO / RefCOCO+ / RefCOCOg [49]	Referring expressions	Phrase grounding, referring detection	∼140k expressions, 50k images
Flickr30K Entities [50]	Image–sentence correspondences	Region-to-phrase grounding	30k images, 275k entities
Visual Genome [51]	Scene graphs, visual reasoning	Dense region annotations	108k images, 1.5M regions
POPE [52]	Hallucination diagnosis	Object hallucination evaluation	10k queries, 10 categories
H-POPE [53]	Holistic hallucination benchmark	Multi-object hallucination	20k visual-language pairs
THRONE [54]	Robustness to hallucination	Factual consistency, reasoning	7k compositional tasks
Hallucinogen [55]	Error taxonomy of hallucination	Diagnostic benchmark	12k visual queries
HalluBench [56]	Categorized hallucination types	Diagnostic evaluation for GPT-4V, MLLMs	5k image–text pairs
MMBench [43]	Multimodal reasoning	QA, captioning, detection	30k samples, 20 categories
MMBench-Plus [45]	Fine-grained multimodal evaluation	Compositional reasoning, vision–language QA	50k prompts, 25 tasks
MME [44]	Multimodal evaluation	Perception, comprehension, reasoning	30k visual–text tasks

For quantitative comparison, mAP is typically reported for detection, while reasoning quality is measured via textual agreement or question–answer accuracy. Qualitative visualization of LLM reasoning chains helps interpret model behavior, revealing whether bounding boxes correspond to the LLM's textual justifications.

Despite promising results, LLM-guided detection faces several obstacles:

- **Latency and cost.** Running large models like GPT-4V or InternVL in the detection loop is computationally expensive, especially for real-time applications.
- **Stability and determinism.** LLM outputs vary with temperature sampling and prompt phrasing; inconsistent reasoning leads to inconsistent detections.

 doi:10.20944/preprints202511.0879.v1

- **Grounding accuracy.** Many LLMs lack explicit spatial understanding and rely on external detectors for localization; this dependency may propagate detector biases.
- **Calibration and confidence.** Integrating probabilistic outputs from heterogeneous modules (LLM and detector) remains challenging.

Ongoing research explores techniques to mitigate these limitations: structured prompting, reasoning templates, and cross-model consensus (discussed in Section 4) can improve both robustness and interpretability.

LLM-guided detection represents a crucial step toward reasoning-aware perception. By combining the contextual understanding of LLMs with the spatial precision of visual detectors, these hybrid systems enable complex, instruction-driven object detection. They also provide the conceptual foundation for *Multi-LLM Consensus*, where multiple reasoning agents cooperate to produce more reliable and semantically consistent detections—the topic of the next section.

4. Multi-LLM Consensus and Ensemble Reasoning

While individual large language or vision–language models can interpret images and produce structured detections, their outputs are often inconsistent. Differences in training data, tokenization, or reasoning style may lead one model to hallucinate an object that another correctly omits. This observation parallels early findings in classical machine learning: combining multiple imperfect models can yield a system more accurate and robust than any single one.

The *Multi–Large–Language–Model (Multi-LLM) Consensus* paradigm extends this principle to reasoning-based vision systems. Here, several LLMs or VLMs receive the same image and prompt, independently generate structured predictions (e.g., lists of objects and bounding boxes), and a consensus mechanism fuses these outputs into a unified result. The process integrates linguistic reasoning, visual perception, and statistical aggregation—effectively merging ensemble learning with multimodal understanding.

Conceptually, Multi-LLM Consensus can occur at different levels of the detection pipeline:

- **Prompt-level consensus:** models agree on semantic understanding before detection (shared class or phrase lists).
- **Reasoning-to-detection consensus:** models produce reasoning chains that inform separate detectors.
- **Box-level consensus:** final spatial outputs (bounding boxes) are merged geometrically or probabilistically.
- **Hybrid consensus:** combinations of the above stages in a unified, hierarchical pipeline.

These mechanisms are summarized in Table 3, which compares representative frameworks and their properties.

Table 3. LLM-level consensus methods and their suitability for object-detection pipelines.

Method	Mechanism	Judge/Ranker	Call Cost	Aggregation Type	Fits Detection Stage	Pros / Cons
Self-Consistency [57]	Sample multiple CoT traces majority on final answer	No	k samples (1 model)	Voting (selection)	Prompt-level (classes/queries)	+ Simple, robust – needs deterministic schema
Mixture-of-Agents (MoA) [10]	Multiple models answer meta-consensus	Optional	N models	Voting / weighted	Prompt-level; can steer detectors	+ Heterogeneity gains – coordination cost
LLM-Blender [11]	Pairwise ranking generative fusion	Yes (PairRanker)	N + ranker + fuser	Rank-and-fuse (gen.)	Prompt-level (schema merging)	+ Synthesizes best of candidates – ranker overhead
LLM-as-a-Judge [58]	External LLM scores candidates	Yes	N + judge	Selection (scoring)	Prompt-level (JSON selection)	+ Close to human prefs – judge bias/cost
Multi-Agent Debate (MAD) [59]	Iterative critique/defense until convergence	No	Iterative rounds $\times N$	Dialogue-based consensus	Prompt-level, pre-detector (class list)	+ Improves factuality – latency
Free-MAD [60]	Debate without explicit controller decentralized debate	No	Iterative rounds $\times N$	Decentralized debate	Prompt-level	+ Lower orchestration – stability varies
LLM-TOPLA [61]	Diversity-maximizing ensemble pruning	No	N (pruned)	Diversity-aware voting	Prompt-level (class candidates)	+ Better diversity/efficiency – needs metrics
ICE (Iterative Consensus Ensemble) [62]	Iterative re-asking cross-checking	No	Iterative N	Iterative voting/refinement	Prompt-level, label vetting (queries)	+ Large gains in reliability – more calls
SkillAggregation [63]	Reference-free model-dependent weighting	No	N	Soft weighting (skills)	Prompt-level	+ No labels needed – weighting estimation
Voting Ensembles (Trustworthy) [64]	Strict/abstention voting for trust	No	N	Conservative vote	Prompt-level	+ Higher trust – lower coverage
FusionFactory [65]	Fuse multi-LLM logs (query/thought/model levels)	Optional	Offline + N	Multi-level fusion	Prompt-level (playbooks)	+ Systematic logs reuse – infra req.
Compound Inference Scaling [66]	Analysis: vote/filter-vote scaling laws	No	N/A (guidance)	Design guidance	Design of consensus depth	+ Predicts non-monotonic gains – not a method
Survey (Text/Code Ensembles) [67]	Taxonomy of output-level ensembling	No	N/A	Survey	Method selection	+ Broad guidance – not executable

 doi:10.20944/preprints202511.0879.v1

4.1. Prompt-Level Consensus

Prompt-level consensus is the most intuitive form: multiple LLMs or VLMs receive the same task instruction (e.g., "List and localize all visible objects in the image as JSON bounding boxes") and generate independent textual outputs. The system then reconciles these responses through linguistic aggregation, ensuring semantic consistency before any spatial localization.

A simple but effective method, **Self-Consistency** [57], generates multiple reasoning paths from the same model and selects the final answer appearing most frequently. Originally designed for textual reasoning, this approach translates naturally to detection: if several reasoning paths predict the same object label or bounding box, that consensus becomes the output. Self-Consistency marginalizes over stochastic reasoning errors, improving stability with minimal overhead.

The **Mixture-of-Agents (MoA)** framework [10] generalizes this idea by involving multiple heterogeneous LLMs ("agents") instead of multiple samples from one model. Each agent provides a candidate output, and a meta-consensus (often another LLM) integrates them via voting or weighted averaging. This architecture exploits the diversity of models—for instance, pairing GPT-4V (strong reasoning) with InternVL (strong perception) yields richer, complementary predictions.

LLM-Blender [11] takes a generative fusion approach: it first ranks candidate outputs using a separate "Ranker" model and then synthesizes a new answer that combines their strengths. Similarly, the **LLM-as-a-Judge** paradigm [58] employs a powerful LLM (e.g., GPT-4) to evaluate and score outputs from other models, selecting the most coherent one. Both approaches approximate human-like arbitration, improving factual correctness and readability, though at increased computational cost.

In contrast to static voting, **Multi-Agent Debate (MAD)** [59,68] allows LLMs to iteratively critique and refine each other's outputs. Agents engage in dialogue, pointing out inconsistencies or omissions until convergence. This process reduces hallucinations and often produces more complete object lists, though it increases latency and requires careful orchestration.

Recent work such as **Free-MAD** [60] removes explicit coordination, relying on self-organized dialogue among agents. **LLM-TOPLA** [61] instead maximizes diversity among candidate answers before voting, preventing redundant reasoning. Together, these methods demonstrate that reasoning diversity—analogous to ensemble variance—directly correlates with performance gains in consensus systems.

4.2. Reasoning-to-Detection Consensus

Whereas prompt-level consensus operates in the textual domain, *reasoning-to-detection consensus* couples linguistic agreement with visual grounding. Each agent produces structured reasoning traces—e.g., a list of object hypotheses or attributes—which are executed by open-vocabulary detectors such as Grounding-DINO or OWL-ViT.

For example, one model might reason, "objects include `[car, person, bicycle]`," while another proposes `[car, truck, pedestrian]`. Consensus mechanisms reconcile these hypotheses into a unified list of detection categories, which then guide the detector to perform localization. This process mirrors the architectures of DetGPT [12] and ContextDET [13] (see Section 3), extended to multi-agent reasoning.

Reasoning-to-detection consensus can be achieved through:

- **Weighted voting** on predicted object categories or attributes.
- **Union-based fusion** of all unique object classes to maximize recall.
- **Confidence calibration**—assigning trust weights to each model based on historical accuracy or semantic agreement.

By merging independent reasoning traces, the system can recover missed detections, normalize label synonyms, and reduce hallucinations—achieving better semantic coverage than any single agent.

 doi:10.20944/preprints202511.0879.v1

4.3. Box-Level Fusion Consensus

At the final stage of detection, each model (or reasoning-to-detection pipeline) outputs bounding boxes with class labels and confidence scores. *Box-level consensus* merges these spatial results into a unified detection map, applying well-established ensemble techniques from computer vision.

The simplest method, **Non-Maximum Suppression (NMS)**, removes overlapping boxes exceeding a given Intersection-over-Union (IoU) threshold, but it discards valuable information from weaker detections. More advanced algorithms preserve these details:

- **Weighted Boxes Fusion (WBF)** [41] computes a confidence-weighted average of overlapping boxes, producing smoother localization.
- **Agglomerative Late Fusion (ALFA)** [8] clusters spatially close boxes and merges them using geometric and semantic similarity.
- **Probabilistic Ensembling (ProbEn)** [9] models each detection as a spatial probability distribution, merging them through Bayesian inference to estimate uncertainty.
- **Soft-NMS** [40] decays the confidence of overlapping boxes rather than suppressing them entirely.

These techniques, originally designed for ensembling CNN-based detectors, can be directly reused in Multi-LLM Consensus pipelines: once multiple agents produce structured bounding boxes, spatial fusion ensures geometric coherence. This step is especially valuable when different models vary in localization accuracy or calibration.

4.4. Hybrid and Hierarchical Consensus

Real-world consensus frameworks often combine several of the above stages in a hierarchical manner:

1. Prompt-level consensus aligns the semantic vocabulary and filters hallucinated classes.
2. Reasoning-to-detection consensus generates consolidated detection queries.
3. Box-level fusion merges final spatial predictions from multiple detectors.

Such multi-stage designs have been successfully employed in hybrid architectures that integrate both reasoning diversity and geometric robustness. Recent frameworks like **Iterative Consensus Ensemble (ICE)** [62] and **SkillAggregation** [63] extend this idea further: agents iteratively re-ask or cross-validate each other, refining consensus until stable agreement is reached. ICE demonstrates that iterative re-evaluation significantly improves factuality and consistency, while SkillAggregation introduces weighting based on model "skill vectors" learned from prior performance, enabling reference-free trust estimation.

4.5. Scaling and Efficiency

Running multiple large models concurrently introduces substantial cost. Recent studies such as **Compound Inference Scaling** [69] analyze how performance gains scale with the number of models and calls. They find that improvements follow a diminishing-return pattern—typically saturating beyond 3–5 diverse agents—consistent with classical ensemble theory. Hence, practical systems often limit ensemble size and use parallel inference or adaptive selection strategies to balance accuracy and latency.

Furthermore, the **FusionFactory** framework [65] proposes unifying multiple LLM capabilities (e.g., reasoning, summarization, planning) by merging their intermediate logs rather than raw outputs. This reduces the need for repeated inference, offering a path toward efficient large-scale consensus.

4.6. Challenges and Research Directions

Multi-LLM Consensus represents a natural evolution of ensemble learning, extended into the multimodal domain. By aggregating the reasoning and perception outputs of multiple agents, consensus frameworks achieve greater robustness, semantic coverage, and interpretability. They form the conceptual bridge between LLM-guided detection (Section 3) and data-fusion algorithms (Sec-

 doi:10.20944/preprints202511.0879.v1

tion 5), establishing the methodological core of collaborative multimodal perception. While Multi-LLM Consensus improves robustness, it introduces new challenges:

- **Vocabulary alignment:** merging outputs with differing lexical forms ("bike" vs. "bicycle") requires semantic normalization.
- **Confidence calibration:** combining heterogeneous confidence scores demands robust scaling (e.g., temperature calibration, isotonic regression).
- **Conflict resolution:** deciding between mutually exclusive predictions ("cat" vs. "dog") is nontrivial when confidences are uncalibrated.
- **Efficiency:** high computational cost and synchronization overhead limit deployment in real-time settings.

Addressing these issues may involve consensus-aware training, where models are optimized jointly for agreement rather than isolated accuracy—a direction explored further in Section 7.

5. Data Fusion Algorithms for Detection Consensus

Data fusion algorithms form the computational backbone of Multi-LLM Consensus frameworks. The success of Multi-LLM Consensus in object detection ultimately depends on how individual model outputs are combined. Regardless of whether predictions come from classical CNN detectors, transformer-based systems, or LLM-guided reasoning pipelines, all yield structured results—bounding boxes, class labels, and confidence scores—that must be reconciled into a single, coherent detection map. This step is known as *data fusion* or *ensemble fusion*. They translate the abstract idea of "agreement among agents" into precise geometric and probabilistic operations. Choosing an appropriate fusion method – balancing accuracy, interpretability, and efficiency – is therefore crucial for deploying reliable consensus-based object detectors.

Historically, the concept originates from **ensemble learning** in machine learning [70], where multiple predictors are combined to reduce variance and improve generalization. In detection, fusion serves similar goals: (1) increase recall by pooling complementary detections, (2) improve localization by averaging correlated boxes, and (3) enhance calibration by smoothing overconfidence from individual models. Table 3 summarizes major fusion algorithms used in both classical and LLM-based detection.

5.1. Classical Fusion Strategies

The simplest and most widely adopted method is **Non-Maximum Suppression (NMS)** [71]. It removes redundant detections by keeping the highest-confidence box in a cluster of overlapping predictions that exceed a given Intersection-over-Union (IoU) threshold. While computationally efficient, NMS discards valuable information from lower-scoring boxes and may underperform when multiple detectors disagree on precise box positions.

To address these limitations, **Soft-NMS** [40] replaces hard suppression with confidence decay. Instead of discarding overlapping boxes, it decreases their confidence scores proportionally to IoU overlap:

$$s_i' = s_i \cdot \exp\left(-\frac{\mathrm{IoU}(b_i, b_{max})^2}{\sigma}\right),$$

where s_i is the original confidence and σ controls decay smoothness. This continuous weighting maintains recall and avoids abrupt score discontinuities, making it especially effective in crowded scenes.

The most influential modern technique is **Weighted Boxes Fusion (WBF)** [7,72]. Rather than eliminating overlapping boxes, WBF computes their weighted average:

$$b_{\text{fused}} = \frac{\sum_i s_i \cdot b_i}{\sum_i s_i},$$

 doi:10.20944/preprints202511.0879.v1

where b_i are box coordinates and s_i the confidence scores. By averaging coordinates from all detections with significant overlap, WBF achieves more precise localization and higher mAP than traditional NMS. It became the de facto standard for ensembling heterogeneous detectors.

Agglomerative Late Fusion (ALFA) [8] generalizes WBF by clustering boxes according to geometric and semantic similarity. ALFA iteratively merges clusters with high IoU and class similarity, computing fused boxes via weighted averaging. This hierarchical approach maintains robustness when fusing outputs from models with varying confidence scales or class ontologies, which is typical in Multi-LLM settings.

Probabilistic Ensembling (ProbEn) [9] interprets each detection as a spatial probability density function—typically a 2D Gaussian over box coordinates—and performs Bayesian fusion to estimate the posterior distribution of the true object location. This yields not only a fused box but also an uncertainty estimate, useful for risk-aware applications such as autonomous driving or medical imaging.

5.2. Fusion of Heterogeneous Outputs

Multi-LLM Consensus often combines outputs from models with heterogeneous formats: some produce exact bounding boxes, others polygons, segmentation masks, or textual object lists converted to coordinates. Ensuring consistent fusion therefore requires three key preprocessing steps:

1. **Coordinate normalization:** mapping all spatial outputs to the same image reference frame and resolution.
2. **Label harmonization:** aligning class names using semantic embeddings (e.g., CLIP text space) to unify synonyms and resolve ambiguities (`"bike"` vs. `"bicycle"`).
3. **Confidence calibration:** normalizing scores across models through temperature scaling or isotonic regression to avoid bias toward overconfident agents.

Proper standardization at this stage determines how effective later geometric fusion can be.

5.3. Hierarchical and Multi-Stage Fusion

Modern consensus frameworks rarely rely on a single fusion step. Instead, they employ hierarchical pipelines that merge information at multiple semantic and spatial levels:

- **Semantic fusion:** unifying class vocabularies across reasoning agents before spatial processing.
- **Geometric fusion:** applying algorithms such as WBF or ALFA to merge bounding boxes per class.
- **Confidence fusion:** combining calibrated confidence scores through weighted averaging or Bayesian posterior computation.

This hierarchy ensures that reasoning agreement constrains spatial aggregation, resulting in more interpretable and stable detections. In practice, multi-stage fusion improves both recall and calibration compared with single-stage alternatives.

5.4. Adaptive and Trust-Weighted Fusion

When models differ in reliability, assigning uniform weights to all of them may be suboptimal. **Trust-weighted fusion** adjusts model contributions based on empirical or contextual reliability. Each model m_k is assigned a trust coefficient w_k proportional to its past accuracy or semantic agreement with peers:

$$b_{\text{final}} = \frac{\sum_k w_k \cdot s_k \cdot b_k}{\sum_k w_k \cdot s_k}.$$

Trust weights can be computed dynamically during inference—for example, by measuring cross-model agreement on overlapping regions. Recent frameworks such as **SkillAggregation** [63] learn these weights automatically through meta-optimization, resulting in adaptive consensus that improves over static averaging.

 doi:10.20944/preprints202511.0879.v1

5.5. Uncertainty-Aware and Probabilistic Fusion

A limitation of traditional fusion methods is their deterministic nature: they produce single boxes without uncertainty estimates. In contrast, probabilistic approaches explicitly model uncertainty, representing each detection as a probability distribution. Beyond ProbEn [9], recent works propose Gaussian mixture models or Monte-Carlo fusion, where uncertainty is propagated through all stages of consensus. This is particularly valuable when combining reasoning-based outputs from LLMs, which may have high epistemic uncertainty due to prompt variability.

Visualizing uncertainty, e.g., by overlaying variance maps, helps interpret ambiguous predictions and supports downstream decision-making, such as active learning or human-in-the-loop verification.

5.6. Efficiency Considerations

Data fusion introduces computational overhead, especially when combining thousands of detections per image from multiple agents. Efficient implementations vectorize IoU computation and parallelize clustering on GPUs. Recent toolkits, such as **FusionFactory** [65], automate the merging of heterogeneous outputs from LLMs and VLMs at scale. Additionally, lightweight approximations of WBF using sparse indexing or confidence pruning can reduce complexity without substantial accuracy loss.

In practice, fusion time typically accounts for less than 5–10% of total inference cost, making it a negligible bottleneck compared with running large foundation models.

5.7. Integration in Multi-LLM Consensus Pipelines

In full Multi-LLM systems (Figure 1), fusion serves as the final unification step after semantic and reasoning consensus. For example:

1. Multiple LLMs generate independent structured detections.
2. Semantic fusion merges their textual outputs into a canonical class list.
3. Box-level algorithms (WBF, ALFA, ProbEn) fuse spatial predictions.
4. Confidence calibration and uncertainty propagation yield the final detection map.

This modular design allows swapping fusion algorithms depending on application needs:

- WBF for fast, accurate ensembling of similar detectors,
- ALFA for robust cross-model integration,
- ProbEn for uncertainty-aware applications.

6. Evaluation and Benchmarking of Consensus-Based Detection

Evaluating consensus-based object detection is fundamentally more complex than evaluating a single detector. In traditional setups, one measures how well a model localizes predefined classes. In Multi-LLM Consensus systems, however, predictions arise from multiple heterogeneous agents, often with open vocabularies and uncalibrated confidence scores. Consequently, performance must be assessed along several dimensions:

- spatial accuracy (bounding-box localization),
- semantic correctness (label or text alignment),
- calibration and uncertainty,
- inter-model agreement and robustness.

A rigorous evaluation framework is therefore essential for ensuring comparability across different consensus strategies.

6.1. Datasets and Benchmarks

Consensus systems are typically evaluated using a mix of closed- and open-vocabulary benchmarks. Traditional datasets such as **COCO** and **Pascal VOC** remain useful for quantitative baselines, but open-vocabulary and grounding datasets provide richer evaluation signals (see Table 2).

 doi:10.20944/preprints202511.0879.v1

Datasets **LVIS** [47] and **ODinW/ELEVATER** [48] offer large vocabularies and long-tailed distributions, enabling separate evaluation on *seen* and *unseen* categories. These datasets are ideal for measuring recall improvements and vocabulary generalization in consensus pipelines.

Datasets **RefCOCO**, **RefCOCO+**, and **RefCOCOg** [49] test whether models can correctly localize entities described by free-form language (e.g., "the person in a red shirt"). Consensus systems are evaluated by how well their fused detections align with textual descriptions across agents.

Recent datasets such as **MMBench** [43], **MME** [44], and **MMBench-Plus** [45] provide reasoning-intensive tasks that link detection with contextual understanding. They test whether multiple LLMs can collaboratively reason about spatial relations ("find the largest object left of the dog") and whether consensus improves factual consistency.

For specialized tasks (medical imaging, aerial imagery), datasets like **xView**, **DeepLesion**, or **BDD100K** assess how well consensus generalizes across domains. The key advantage is that fusion can mitigate single-model biases or overfitting to specific image styles.

6.2. Metrics for Spatial Accuracy

The primary measure of detection quality remains **mean Average Precision (mAP)** computed at multiple IoU thresholds (e.g., 0.5:0.95). mAP summarizes precision–recall trade-offs and directly quantifies localization accuracy. Additional metrics include:

- **Average Recall (AR)** – sensitivity to object coverage, useful for consensus ensembles that prioritize recall.
- **IoU consistency** – mean IoU across fused and individual boxes, showing geometric agreement between agents.
- **F1-score** – particularly relevant when detection decisions are binarized for task-specific evaluation.

Consensus systems typically show 2–5 pt mAP improvement over single models [7,12], especially for rare classes and ambiguous objects.

6.3. Metrics for Semantic and Reasoning Quality

Since consensus often involves linguistic reasoning, semantic evaluation is equally important.

When agents use different labels ("bicycle" vs. "bike"), cosine similarity in CLIP or Sentence-BERT embedding space can quantify semantic proximity. A consensus detection is deemed correct if its textual label lies within a similarity threshold to the ground-truth description.

Metrics from image captioning and grounding, BLEU, METEOR, or CIDEr, can measure correspondence between generated textual reasoning and localized regions. For referring expression datasets, *Referring Expression Accuracy (RefAcc)* measures the fraction of expressions localized correctly.

Some benchmarks (e.g., MMBench [43]) include multiple-choice reasoning questions. Here, consensus is evaluated by majority agreement across agents or by whether the fused reasoning chain leads to the correct answer.

6.4. Calibration and Uncertainty Metrics

When fusing predictions from multiple agents, **confidence calibration** becomes critical, as uncalibrated scores can distort fusion weights and give excessive influence to overconfident models. To quantify calibration quality, researchers commonly use **Expected Calibration Error (ECE)**, which measures the average absolute difference between predicted confidence and empirical accuracy across bins, the **Brier Score**, which captures overall probabilistic accuracy with lower values indicating better calibration, and **Negative Log-Likelihood (NLL)**, particularly suitable when probabilistic outputs (as in ProbEn [9]) are available. A well-calibrated consensus should produce confidence values that meaningfully reflect correctness probability, thereby improving interpretability for downstream decision-making.

 doi:10.20944/preprints202511.0879.v1

6.5. Agreement and Diversity Measures

A unique property of consensus frameworks is that multiple models can disagree even when all are "partly correct." Measuring the diversity among agents provides insight into ensemble effectiveness. One way to do this is by analyzing the **pairwise disagreement rate**, defined as the fraction of detections where two agents differ in label or localization beyond a threshold. Another indicator is the **inter-agent IoU**, which captures the geometric overlap between agents' detections—low values suggest complementary localization behavior. A further perspective is given by the **entropy of consensus**, reflecting the distributional uncertainty over class votes; lower entropy corresponds to stronger agreement.

Empirically, ensemble benefit correlates with diversity: systems with more varied reasoning traces (e.g., different prompt styles or model families) yield higher mAP gains [10,62].

6.6. Hallucination, Robustness, Efficiency and Cost Evaluation

A major advantage of consensus is its ability to reduce hallucination—false detections unsupported by the image. Recent analyses [5,6] show that hallucination frequency can be quantified as:

$$\text{Hallucination Rate} = \frac{\#\text{ false detections with no ground-truth match}}{\text{total detections}}.$$

Consensus typically decreases hallucination rate by suppressing outlier predictions that appear in only one model's output. Robustness can also be tested through perturbation benchmarks (e.g., image noise, occlusion) to measure how consistently consensus maintains detection quality.

Because Multi-LLM Consensus involves multiple heavy models, reporting only accuracy is insufficient. Researchers should also measure **inference latency** (seconds per image), **compute cost** (GPU-hours or energy usage), **consensus overhead** (time spent on fusion and calibration), as well as **scaling efficiency**, defined as the improvement in accuracy per additional model. Studies such as [69] show that performance improvements saturate after about five heterogeneous agents, beyond which cost begins to dominate benefit.

6.7. Best Practices for Evaluation

Based on current literature [5,10,12,41], one may formulate several best practices:

- **Standardize prompts and schemas.** Use identical instruction templates and JSON structures across agents to ensure comparability.
- **Report both spatial and semantic metrics.** mAP alone may obscure linguistic or reasoning improvements.
- **Visualize qualitative results.** Show consensus heatmaps and reasoning traces for interpretability.
- **Measure diversity explicitly.** Diversity metrics explain why certain ensembles succeed or fail.
- **Include efficiency analysis.** Present throughput and scaling trends alongside accuracy gains.

Evaluation of consensus-based detection requires a multidimensional perspective: localization accuracy, semantic understanding, calibration, diversity, and efficiency all play crucial roles. By adopting unified metrics and standardized datasets, the community can move toward fair comparison of consensus strategies and clearer insight into how reasoning diversity translates into perceptual reliability.

7. Challenges and Future Research Directions

While Multi-LLM Consensus has proven effective in improving robustness and semantic coverage, it also exposes several unresolved challenges. These limitations span computational, methodological, and ethical dimensions. Understanding them is crucial for transforming consensus-based detection from an experimental strategy into a practical, scalable paradigm. This section summarizes the main bottlenecks and emerging research directions that may shape the next generation of multimodal consensus systems.

 doi:10.20944/preprints202511.0879.v1

7.1. Current Challenges

Current Multi-LLM Consensus pipelines face several open challenges. First, **vocabulary alignment and label consistency** remains difficult, as different LLMs or VLMs may use incompatible tokenizers, naming conventions, or semantically equivalent but divergent labels (e.g., "bike" vs. "bicycle"), causing fusion mechanisms to interpret genuine agreement as conflict. Second, **confidence calibration and uncertainty estimation** is still unresolved: each model's confidence distribution reflects its own training dynamics, meaning that overconfident but inaccurate agents may overpower more reliable ones, and existing techniques such as temperature scaling or isotonic regression [73] only partially mitigate the issue. Third, the **computational cost and scalability** of running multiple foundation models is substantial, with latency scaling roughly linearly in the number of agents and diminishing returns observed beyond 3–5 heterogeneous models [69], motivating research into lightweight consensus, pruning, and adaptive routing. Fourth, **evaluation standardization** is still lacking: unlike classical detection, there is no widely accepted benchmark protocol or dataset configuration for fair comparison, which hinders reproducibility and consensus on best practices. Furthermore, **bias propagation and fairness** remain critical concerns, as ensembling similar models can amplify shared prejudices rather than cancel them, while accountability becomes diffused across agents. Finally, **interpretability and traceability** degrade with increasing pipeline complexity, making it difficult to reconstruct how a final decision was produced unless explicit reasoning graphs, provenance chains, or audit trails are exposed.

7.2. Emerging Research Trends

The trajectory of research indicates a shift from static ensemble fusion toward dynamic, collaborative intelligence. Consensus frameworks may evolve into foundational infrastructures that jointly maintain situational awareness through multiple reasoning and perception models. By combining interpretability, calibration, and cooperation, they could provide the backbone for trustworthy multimodal AI—capable of not only *seeing* but also *reasoning and verifying what it sees*. One may observe the following emerging trends in this field:

- **Consensus-aware training.** Most current systems perform consensus only at inference time. A natural next step is to incorporate consensus into training objectives.
- **Consensus-aware training** optimizes models for agreement or uncertainty reduction, enabling them to cooperate more effectively post hoc. Potential methods include: (1) multi-agent co-training with shared agreement losses, (2) reinforcement learning with group-level rewards, (3) distillation of consensus outputs into smaller student models. Early explorations, such as **SkillAggregation** [63], already move in this direction.
- **Structured and verifiable reasoning.** To make LLM-based detection auditable, reasoning should be expressed in structured, machine-verifiable formats rather than free text. Possible representations include symbolic scene graphs, pseudo-code, or executable reasoning traces that map directly to detection actions. Such explicit structure would enable formal verification, reduce hallucination, and make consensus derivations traceable.
- **Hierarchical and adaptive consensus.** Future systems will likely employ multi-layered architectures in which agents specialize and communicate. A lightweight model could handle generic detection, while larger models refine uncertain cases or reason about complex relations. Hierarchical consensus has already shown promise in hybrid designs (Section 4); integrating adaptive agent selection and trust-weighted fusion could further balance cost and accuracy.
- **Multimodal uncertainty modeling.** Robust perception requires quantifying both epistemic (model-based) and aleatoric (data-based) uncertainty. Combining probabilistic box fusion (Section 5) with reasoning uncertainty from LLMs could yield full-scene uncertainty maps. These estimates would support risk-aware decision-making and improve safety-critical applications such as autonomous driving.

 doi:10.20944/preprints202511.0879.v1

- **Collaborative perception ecosystems.** The long-term vision for Multi-LLM Consensus extends beyond individual detectors toward collaborative perception ecosystems. In such systems, multiple agents—text-based, vision-based, or multimodal—share information dynamically across tasks, scenes, and time. Each agent contributes complementary reasoning or sensory capabilities, and consensus serves as the coordination mechanism ensuring global coherence. Developing communication protocols, trust governance, and distributed training for such ecosystems is an open frontier.

8. Conclusion

The convergence of large-scale language and vision–language modeling with classical ensemble principles has opened a new frontier in visual perception research. This survey has examined the emerging paradigm of *Multi–Large–Language–Model Consensus* (Multi-LLM Consensus) for object detection – an approach that unites the reasoning diversity of multiple LLMs with the spatial precision of visual detectors to achieve more reliable, interpretable, and semantically rich perception.

We began by tracing the evolution of object detection from region-based CNNs through transformer architectures and open-vocabulary systems, highlighting how language supervision has progressively expanded the perceptual scope of detectors. Next, we reviewed the integration of LLMs as reasoning engines in hybrid frameworks such as DetGPT, ContextDET, and LLMDet, where language models guide or supervise the detection process. Building upon these foundations, we introduced a taxonomy of Multi-LLM Consensus strategies, ranging from prompt-level linguistic agreement to reasoning-to-detection fusion and spatial box-level ensembling. We analyzed representative algorithms, including Self-Consistency, Mixture-of-Agents, LLM-Blender, and Weighted Boxes Fusion.

We further discussed data fusion methodologies that operationalize consensus, covering techniques such as WBF, ALFA, ProbEn, and trust-weighted fusion. Evaluation practices were reviewed comprehensively, encompassing not only spatial accuracy (mAP, AR) but also semantic consistency, calibration, inter-model diversity, and hallucination reduction. Empirical evidence across multiple studies indicates that consensus ensembles consistently improve recall, calibration, and robustness compared with single-model baselines—particularly for rare, ambiguous, or open-vocabulary categories.

Nevertheless, challenges remain. Vocabulary alignment, score calibration, and computational scalability continue to limit practical deployment. The absence of standardized benchmarks impedes fair comparison, while bias propagation and lack of interpretability raise ethical and transparency concerns. Addressing these issues will require advances in *consensus-aware training*, structured reasoning representations, probabilistic fusion, and collaborative multimodal ecosystems where diverse agents cooperate dynamically.

Looking forward, Multi-LLM Consensus represents a conceptual shift toward **collective intelligence in artificial perception**. By integrating reasoning, grounding, and agreement among autonomous agents, consensus frameworks can transform object detection from a static pattern-recognition task into an adaptive process of shared understanding. As foundation models become increasingly multimodal and interconnected, consensus mechanisms are likely to play a central role in ensuring reliability, transparency, and trust in next-generation AI perception systems. Together, these insights outline a path toward more reliable, interpretable, and trustworthy multimodal AI systems that integrate reasoning and perception through consensus.

Author Contributions: Conceptualization, M.I.; methodology, M.I,; investigation, M.I.,M.G.; resources, M.I.,M.G.; writing—original draft preparation, M.I.,M.G.; writing—review and editing, M.I.; visualization, M.I.; supervision, M.I. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Informed Consent Statement: Not applicable

 doi:10.20944/preprints202511.0879.v1

Acknowledgments: During the preparation of this manuscript/study, the author(s) used ChatGPT5 for the purposes of preliminary paper search and summaries, partial descriptions, language corrections. The authors have reviewed and edited the output and take full responsibility for the content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ALFA	Agglomerative Late Fusion Algorithm
AP	Average Precision
AR	Average Recall
CLIP	Contrastive Language–Image Pre-training
COCO	Common Objects in Context
CVinW	Computer Vision in the Wild
DETR	DEtection TRansformer
ECE	Expected Calibration Error
ELEVATER	Evaluating Language-Augmented Visual Models (benchmark/toolkit)
FPS	Frames Per Second
GLIP	Grounded Language-Image Pre-training
GPT-4V	GPT-4 with Vision
ICinW	Image Classification in the Wild
IoU	Intersection over Union
LaMI-DETR	Language Model Instruction DETR
LLaVA	Large Language and Vision Assistant
LLM	Large Language Model
LLM-Blender	Ensembling Large Language Models with Pairwise Ranking and Generative Fusion
LLMDet	Learning Strong Open-Vocabulary Object Detectors under LLM Supervision
LVIS	Large Vocabulary Instance Segmentation
MAD	Multi-Agent Debate
MMBench	Multimodal Benchmark
MME	Multimodal Evaluation
MLLM	Multimodal Large Language Model
MoA	Mixture of Agents
NMS	Non-Maximum Suppression
ODinW	Object Detection in the Wild
OV	Open-Vocabulary
OVOD	Open-Vocabulary Object Detection
OWL-ViT	Open-Vocabulary detection with Vision Transformer
POPE	Hallucination evaluation benchmark for VLMs
ProbEn	Probabilistic Ensembling
RefCOCO	Referring Expressions COCO
RefCOCO+	Referring Expressions COCO+
RefCOCOg	Referring Expressions COCOg
VLM	Vision–Language Model
VOC	PASCAL Visual Object Classes
VQA	Visual Question Answering
WBF	Weighted Boxes Fusion
YOLO-World	You Only Look Once (YOLO) – World (open-vocabulary variant)
Qwen-VL	Qwen Vision–Language
InternVL	Intern Vision–Language

 doi:10.20944/preprints202511.0879.v1

References

1. OpenAI. GPT-4V(ision) System Card. https://openai.com/index/gpt-4v-system-card/, 2023. Accessed 2025-10-23.
2. Liu, H.; Li, C.; Hu, X.; Zhang, P. LLaVA-Next: Stronger Visual-Language Reasoning with Better Pretraining and Alignment. *arXiv preprint arXiv:2407.07895* **2024**.
3. Chen, W.; Fang, Y.; Wang, Z.; et al. InternVL: Scaling Up Vision-Language Representation Learning with Massive Multi-Modal Data. *arXiv preprint arXiv:2402.11459* **2024**.
4. Yin, S.; Fu, C.; et al. A Survey on Multimodal Large Language Models. *National Science Review* **2024**, *11*, nwae403. https://doi.org/10.1093/nsr/nwae403.
5. Yin, Z.; Xu, C.; Dai, W.; Liu, Z.; Li, X.; Chen, Y.; Chen, D.; He, L. Mitigating Visual Hallucination in Multimodal Large Language Models: A Survey. *arXiv preprint arXiv:2408.04683* **2024**.
6. Zhu, G.; Zhang, H.; Qiao, Y. A Comprehensive Survey of Open-Vocabulary Object Detection: Challenges, Methods, and Trends. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)* **2024**. In press, https://doi.org/10.1109/TPAMI.2024.3431682.
7. Solovyev, R.; Wang, W.; Gabruseva, T. Weighted Boxes Fusion: Ensembling Boxes from Different Object Detection Models. *Pattern Recognition Letters* **2021**, *152*, 52–60. https://doi.org/10.1016/j.patrec.2021.08.023.
8. Razinkov, I.; Volokitin, A.; Osokin, A. ALFA: Agglomerative Late Fusion for Object Detection Ensembles. In Proceedings of the Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), 2019, pp. 3106–3114. https://doi.org/10.1109/ICCVW.2019.00371.
9. Chen, K.; Song, M.; Sun, M.; Yan, H.; Li, K. ProbEn: Probabilistic Ensembling for Object Detection. In Proceedings of the Proceedings of the European Conference on Computer Vision (ECCV), 2022, pp. 137–153. https://doi.org/10.1007/978-3-031-19812-0_9.
10. Wang, Y.; Wang, Y.; Zhou, H.; Zhou, Z.H. Mixture-of-Agents Enhances Large Language Model Reasoning. In Proceedings of the International Conference on Learning Representations (ICLR), 2024.
11. Jiang, Y.; Xiao, T.; Zhang, C.; Chen, W.; Zhang, T.; Yang, Y. LLM-Blender: Ensembling Large Language Models with Pairwise Ranking and Generative Fusion. In Proceedings of the Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL), 2023, pp. 9343–9362. https://doi.org/10.18653/v1/2023.acl-long.789.
12. Pi, R.; Zang, S.; Luo, W.; Huang, J.; Wang, X.; Zhang, X.; Li, H. DetGPT: Detect What You Need via Reasoning. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2023, pp. 7303–7319. https://doi.org/10.18653/v1/2023.emnlp-main.456.
13. Zang, S.; Pi, R.; Luo, W.; Huang, J.; Wang, X.; Zhang, X.; Li, H. ContextDET: Context Reasoning for Open-Vocabulary Object Detection with Large Language Models. *arXiv preprint arXiv:2311.13024* **2023**.
14. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 580–587. https://doi.org/10.1109/CVPR.2014.81.
15. Girshick, R. Fast R-CNN. In Proceedings of the Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1440–1448. https://doi.org/10.1109/ICCV.2015.169.
16. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems 28 (NIPS 2015), 2015, pp. 91–99.
17. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 936–944. https://doi.org/10.1109/CVPR.2017.106.
18. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779–788. https://doi.org/10.1109/CVPR.2016.91.
19. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the Proceedings of the European Conference on Computer Vision (ECCV), 2016, pp. 21–37. https://doi.org/10.1007/978-3-319-46448-0_2.
20. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal Loss for Dense Object Detection. In Proceedings of the Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2999–3007. https://doi.org/10.1109/ICCV.2017.324.

 doi:10.20944/preprints202511.0879.v1

21. Law, H.; Deng, J. CornerNet: Detecting Objects as Paired Keypoints. In Proceedings of the Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 766–782. https://doi.org/10.1007/978-3-030-01234-2_47.
22. Zhou, X.; Wang, D.; Krähenbühl, P. Objects as Points. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 6529–6538.
23. Tian, Z.; Shen, C.; Chen, H.; He, T. FCOS: Fully Convolutional One-Stage Object Detection. In Proceedings of the Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 9627–9636. https://doi.org/10.1109/ICCV.2019.00972.
24. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. In Proceedings of the Proceedings of the European Conference on Computer Vision (ECCV), 2020, pp. 624–640. https://doi.org/10.1007/978-3-030-58452-8_38.
25. Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; Dai, J. Deformable DETR: Deformable Transformers for End-to-End Object Detection. In Proceedings of the International Conference on Learning Representations (ICLR), 2021.
26. Li, F.; Zhang, H.; Liu, S.; Guo, J.; Ni, L.; Liu, H.H.; Zhang, H. Dn-DETR: Accelerate DETR Training by Introducing Query DeNoising. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 13808–13817. https://doi.org/10.1109/CVPR52688.2022.01341.
27. Zhang, F.; Liu, X.; Zhang, H.; Qiao, Y.; Li, H. DINO: DETR with Improved DeNoising Anchor Boxes for End-to-End Object Detection. In Proceedings of the International Conference on Learning Representations (ICLR), 2023.
28. Lv, W.; Shang, T.; Xu, C.; Song, J.; Niu, X.; et al. RT-DETR: Real-Time DEtection TRansformer. *arXiv preprint arXiv:2304.08069* **2023**.
29. Lv, W.; et al. RT-DETRv2: Improved Real-Time Detection Transformer. *arXiv preprint arXiv:2403.12491* **2024**.
30. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning Transferable Visual Models From Natural Language Supervision. In Proceedings of the Proceedings of the 38th International Conference on Machine Learning (ICML). PMLR, 2021, pp. 8748–8763.
31. Li, X.; Li, X.; Li, X.; Zhang, W.; Sun, P.; Zhang, C.; Tong, Y.; Zhang, L.; Liu, H. Grounded Language-Image Pre-training. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 11102–11112. https://doi.org/10.1109/CVPR52688.2022.01083.
32. Minderer, M.; Gritsenko, A.; Albiero, V.; Drossos, K.; Anderson, P.; Pavlov, M.; He, X.; Basilico, J.; Sermanet, P. Simple Open-Vocabulary Object Detection with Vision Transformers. In Proceedings of the Proceedings of the European Conference on Computer Vision (ECCV), 2022, pp. 81–98. https://doi.org/10.1007/978-3-031-19842-7_6.
33. Minderer, M.; Drossos, K.; Anderson, P.; Basilico, J.; Pavlov, M.; He, X.; Sermanet, P. Scaling Open-Vocabulary Object Detection. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 3966–3976. https://doi.org/10.1109/CVPR52729.2023.00385.
34. Liu, S.; Li, F.; Zhang, H.; Wang, X.; Li, H.; Zhang, H.; Qiao, Y. Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. *arXiv preprint arXiv:2303.05499* **2023**.
35. Ren, Z.; Zhang, H.; Li, F.; Liu, S.; Wang, X.; Zhang, H.; Qiao, Y. Grounding DINO 1.5: Advanced Foundation for Open-World Detection and Captioning. In Proceedings of the Proceedings of the European Conference on Computer Vision (ECCV), 2024.
36. Cheng, S.; Zhang, H.; Zhang, H.; Li, F.; Liu, S.; Qiao, Y. YOLO-World: Real-Time Open-Vocabulary Object Detection. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024, pp. 11153–11163.
37. Bai, J.; et al. Qwen-VL: A Frontier Large Vision-Language Model for Advanced Perception and Understanding. *arXiv preprint arXiv:2404.10399* **2024**.
38. Du, Z.; Wang, X.; Zhang, H.; Qiao, Y.; Li, H. LaMI-DETR: Language-Guided Multi-Modal DETR for Open-Vocabulary Object Detection. In Proceedings of the Proceedings of the European Conference on Computer Vision (ECCV), 2024.
39. Fu, Y.; Zhang, H.; Qiao, Y.; Li, H. LLMDet: Training Open-Vocabulary Detectors with Large Language Models. *arXiv preprint arXiv:2403.11382* **2024**. Identified as 'Proc. CVPR 2025' in original list, but is a 2024 arXiv preprint.
40. Bodla, N.; Singh, B.; Chellappa, R.; Davis, L.S. Soft-NMS — Improving Object Detection With One Line of Code. In Proceedings of the Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 5561–5569. https://doi.org/10.1109/ICCV.2017.593.

 doi:10.20944/preprints202511.0879.v1

41. Solovyev, R.; Wang, W.; Gabruseva, T. Weighted Boxes Fusion: Ensembling Boxes for Object Detection Models. *Image and Vision Computing* **2021**, *111*, 104179. https://doi.org/10.1016/j.imavis.2021.104179.
42. Bai, Z.; Wang, P.; Xiao, T.; He, T.; Han, Z.; Zhang, Z.; Shou, M.Z. Hallucination of Multimodal Large Language Models: A Survey. *arXiv preprint arXiv:2404.18930* **2024**.
43. Liu, Y.; Li, X.; et al. MMBench: Is Your Multi-modal Model an All-around Player? *arXiv preprint arXiv:2307.06281* **2023**.
44. Fu, C.; Chen, P.; Shen, Y.; Qin, Y.; Lin, X.; Yang, J.; Zheng, X.; Li, K.; Sun, X.; Wu, Y.; et al. MME: A Comprehensive Evaluation Benchmark for Multimodal Large Language Models. *arXiv preprint arXiv:2306.13394* **2023**.
45. Liu, Y.; Li, X.; Zhao, R.; Zhang, T.; Wu, Z.; Liu, Q. MMBench-Plus: A Comprehensive and Fine-Grained Evaluation of Multimodal Large Language Models. *arXiv preprint arXiv:2408.03431* **2024**. Extends MMBench with more challenging compositional and reasoning tasks.
46. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In Proceedings of the Proceedings of the European Conference on Computer Vision (ECCV). Springer, 2014, pp. 740–755. https://doi.org/10.1007/978-3-319-10602-1_48.
47. Gupta, A.; Dollár, P.; Girshick, R. LVIS: A Dataset for Large Vocabulary Instance Segmentation. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 5356–5364. https://doi.org/10.1109/CVPR.2019.00550.
48. Li, C.; Liu, H.; Li, L.H.; Zhang, P.; Aneja, J.; Yang, J.; Jin, P.; Hu, H.; Liu, C.; Lee, Y.J.; et al. ELEVATER: A Benchmark and Toolkit for Evaluating Language-Augmented Visual Models. In Proceedings of the Advances in Neural Information Processing Systems 35 (NeurIPS 2022 Datasets and Benchmarks Track), 2022, [2204.08790]. Includes the ODinW suite for open-world object detection.
49. Yu, L.; Poirson, P.; Yang, S.; Berg, A.C.; Berg, T.L. Modeling Context in Referring Expressions. *arXiv preprint arXiv:1608.00272* **2016**. Defines RefCOCO, RefCOCO+, and RefCOCOg.
50. Plummer, B.A.; Wang, L.; Cervantes, C.M.; Caicedo, J.C.; Hockenmaier, J.; Lazebnik, S. Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models. In Proceedings of the Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2015, pp. 2641–2649. https://doi.org/10.1109/ICCV.2015.303.
51. Krishna, R.; Zhu, Y.; Groth, O.; Johnson, J.; Hata, K.; Kravitz, J.; Chen, S.; Kalantidis, Y.; Li, L.J.; Shamma, D.A.; et al. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *International Journal of Computer Vision (IJCV)* **2017**, *123*, 32–73. https://doi.org/10.1007/s11263-016-0981-7.
52. Li, Y.; Du, Y.; Zhou, K.; Wang, J.; Zhao, W.X.; Wen, J.R. Evaluating Object Hallucination in Large Vision-Language Models. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2023, pp. 201–216. Introduces POPE: Polling-based Object Probing Evaluation.
53. Li, Y.; Zhou, K.; Wang, J.; Zhao, W.X.; Wen, J.R. H-POPE: Holistic Evaluation of Object Hallucination in Vision-Language Models. *arXiv preprint arXiv:2402.11895* **2024**. Extension of POPE benchmark for evaluating multimodal hallucination and reasoning.
54. Jin, M.; Xu, C.; Zhang, W.; Zhou, M.; Zhang, Z. THRONE: A Comprehensive Benchmark for Measuring Hallucination in Large Multimodal Models. *arXiv preprint arXiv:2405.05785* **2024**. Evaluates factual grounding and hallucination robustness of MLLMs.
55. Chen, J.; Luo, T.; Liu, S.; Liu, Y.; Zhang, W.; Zhou, J. Hallucinogen: Benchmarking and Analyzing Hallucinations in Multimodal Large Language Models. *arXiv preprint arXiv:2406.18564* **2024**. Provides fine-grained categorization of hallucination errors in vision-language models.
56. Zhang, P.; Zhao, M.; Wang, Y.; Jiang, Y.; Yang, Y. HalluBench: Quantifying and Mitigating Visual Hallucinations in Multimodal Models. *arXiv preprint arXiv:2409.01721* **2024**. A diagnostic benchmark for hallucination categories in GPT-4V and open-source MLLMs.
57. Wang, X.; Wei, J.; Schuurmans, D.; Bosma, M.; Chi, E.H.; Le, Q.V.; Zhou, D. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In Proceedings of the International Conference on Learning Representations (ICLR), 2022.
58. Zheng, L.; Chiang, W.L.; Zhuang, S.; Zhuang, J.S.; Abbeel, P.; Lin, A. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. *arXiv preprint arXiv:2306.05685* **2023**.
59. Du, N.; Dai, A.M.; Zhou, D.; Le, Q.V.; et al.. Improving Factuality and Reasoning in LLMs through Multi-Agent Debate. In Proceedings of the International Conference on Learning Representations (ICLR), 2024. ICLR 2024, Multi-Agent Debate framework.

 doi:10.20944/preprints202511.0879.v1

60. Cui, Y.; Fu, H.; Zhang, H.; Wang, L.; Zuo, C. Free-MAD: Consensus-Free Multi-Agent Debate. *arXiv preprint arXiv:2509.11035* **2025**.
61. Tekin, S.F.; Ilhan, F.; Huang, T.; Hu, S.; Liu, L. LLM-TOPLA: Efficient LLM Ensemble by Maximising Diversity. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, 2024; pp. 11951–11966. https://doi.org/10.18653/v1/2024.findings-emnlp.698.
62. Omar, K.; Alazab, M.; Khan, L. ICE: Iterative Consensus Enhancement for Reliable Multi-Model Decision Making. *medRxiv preprint medRxiv:2405.11233* **2024**. Preprint.
63. Sun, Y.; Zhang, J.; Zhang, M.; Liu, H. Skill Aggregation: Boosting Multimodal Large Language Models via Cross-Model Knowledge Sharing. *arXiv preprint arXiv:2408.11567* **2024**.
64. Nair-Kanneganti, A.; Chan, T.J.; Goldfinger, S.; Mackay, E.; Anthony, B.; Pouch, A. Increasing LLM Response Trustworthiness Using Voting Ensembles. *arXiv preprint arXiv:2510.04048* **2025**.
65. Feng, T.; Zhang, H.; Lei, Z.; Han, P.; Patwary, M.; Shoeybi, M.; Catanzaro, B.; You, J. FusionFactory : Fusing LLM Capabilities with Multi-LLM Log Data. *arXiv preprint arXiv:2507.10540* **2025**.
66. Chen, L.; Davis, J.Q.; Hanin, B.; Bailis, P.; Stoica, I.; Zaharia, M.; Zou, J. Are More LLM Calls All You Need? Towards Scaling Laws of Compound Inference Systems. *arXiv preprint arXiv:2403.02419* **2024**.
67. Ashiga, M.; Jie, W.; Wu, F.; Voskanyan, V.; Dinmohammadi, F.; Brookes, P.; Gong, J.; Wang, Z. Ensemble Learning for Large Language Models in Text and Code Generation: A Survey. *arXiv preprint arXiv:2503.13505* **2025**.
68. Choi, H.K.; Zhu, X.; Li, Y. Debate or Vote: Which Yields Better Decisions in Multi-Agent Large Language Models? In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), 2025. Preprint version on arXiv Aug 2025, NeurIPS 2025 Spotlight.
69. Chen, H.; Wu, S.; Zhao, R.; Li, Y.; Xiong, C. Scaling Laws and Emergent Behaviors in Compound Inference with Multiple Large Language Models. *arXiv preprint arXiv:2406.09742* **2024**.
70. Dietterich, T.G. Ensemble methods in machine learning. In Proceedings of the Multiple Classifier Systems, First International Workshop (MCS). Springer, 2000, pp. 1–15. https://doi.org/10.1007/3-540-45014-9_1.
71. Neubeck, A.; Van Gool, L. Efficient Non-Maximum Suppression. In Proceedings of the Proceedings of the 18th International Conference on Pattern Recognition (ICPR). IEEE, 2006, Vol. 3, pp. 850–855. https://doi.org/10.1109/ICPR.2006.479.
72. Solovyev, R.; Wang, W.; Gabruseva, T. Weighted Boxes Fusion: Ensembling Boxes from Different Object Detection Models. *Image and Vision Computing* **2021**, *107*, 104117. https://doi.org/10.1016/j.imavis.2021.104117.
73. Guo, C.; Pleiss, G.; Sun, Y.; Weinberger, K.Q. On Calibration of Modern Neural Networks. In Proceedings of the Proceedings of the 34th International Conference on Machine Learning (ICML 2017); Precup, D.; Teh, Y.W., Eds. PMLR, 2017, Vol. 70, *Proceedings of Machine Learning Research*, pp. 1321–1330.