Pattern Recognition 173 (2026) 112928

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

LLM-informed global-local contextualization for zero-shot food detection

Xinlong Wang [a], Weiqing Min [b,c], Guorui Sheng [a,*], Jingru Song [a], Yancun Yang [a], Tao Yao [a], Shuqiang Jiang [b,c]

[a] School of Computer Science and Artificial Intelligence, Ludong University, Yantai, 264025, China
[b] State Key Laboratory of AI Safety, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100190, Beijing, China
[c] School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing, 100190, Beijing, China

ARTICLE INFO

Keywords:
Food computing
Zero-shot learning
Zero-shot detection
Fine-grained features

ABSTRACT

Zero-Shot Detection, the ability to detect novel objects without training samples, exhibits immense potential in an ever-changing world, particularly in scenarios requiring the identification of emerging categories. However, effectively applying ZSD to fine-grained domains, characterized by high inter-class similarity and notable intra-class diversity, remains a significant challenge. This is particularly pronounced in the food domain, where the intricate nature of food attributes—notably the pervasive visual ambiguity among related culinary categories and the extensive spectrum of appearances within each food category—severely constrains the performance of existing methods. To address these specific challenges in the food domain, we introduce Zero-Shot Food Detection with Semantic Space and Feature Fusion (ZeSF), a novel framework tailored for Zero-Shot Food Detection. ZeSF integrates two key modules: (1) Multi-Scale Context Integration Module (MSCIM) that employs dilated convolutions for hierarchical feature extraction and adaptive multi-scale fusion to capture subtle, fine-grained visual distinctions; and (2) Contextual Text Feature Enhancement Module (CTFEM) that leverages Large Language Models to generate semantically rich textual embeddings, encompassing both global attributes and discriminative local descriptors. Critically, a cross-modal alignment further harmonizes visual and textual features. Comprehensive evaluations on the UEC FOOD 256 and Food Objects With Attributes (FOWA) datasets affirm ZeSF's superiority, achieving significant improvements in the Harmonic Mean for the Generalized ZSD setting. Crucially, we further validate the framework's generalization capability on the MS COCO and PASCAL VOC benchmarks, where it again outperforms strong baselines. The source code will be publicly available upon publication.

1. Introduction

Food computing is an interdisciplinary field. It merges computer vision and food science. This field plays an vital role in analyzing and understanding food-related visual data [1,2]. With the aid of multimedia technologies, food computing now supports many applications. These applications include food safety monitoring [3], intelligent nutritional assessment [4], and personalized dietary guidance systems [5]. Among these, food detection is a core technology. It involves localizing and identifying specific food items in images. This capability enables practical systems such as automated retail checkout systems [6]. However, the vast and ever-evolving variety of food items in the real world presents a significant challenge. Traditional supervised detection models struggle with scalability and adaptability to emerging food categories due to high annotation costs and limited generalization [7].

Zero-Shot Detection (ZSD) enables the detection of unseen categories through knowledge transfer [8]. However, early methods relying on coarse semantics prove insufficient for the fine-grained nature of food [9]. Food objects present three primary challenges, as shown in Fig. 1. These challenges differ from those of rigid general objects that have consistent structural features. First, they exhibit high inter-class similarity, where different dishes share overlapping visual attributes (e.g., "braised spareribs" vs. "sweet and sour pork ribs"). Second, such dishes show high intra-class diversity, as the same dish varies markedly with preparation (e.g., "fried chicken steak"). Third, there is a lack of rigid structural constraints, leading to high compositional complexity and non-rigidity.

These factors collectively demand an architecture capable of processing multi-scale compositional features and domain-specific semantics. Generic ZSD frameworks do not adequately meet these requirements.

* Corresponding author.
E-mail addresses: xinlongwang@m.ldu.edu.cn (X. Wang), minweiqing@ict.ac.cn (W. Min), shengguorui@ldu.edu.cn (G. Sheng), songjingru@m.ldu.edu.cn (J. Song), Harryyang@ldu.edu.cn (Y. Yang), yaotao@ldu.edu.cn (T. Yao), sqjiang@ict.ac.cn (S. Jiang).

https://doi.org/10.1016/j.patcog.2025.112928
Received 20 June 2025; Received in revised form 20 November 2025; Accepted 14 December 2025
Available online 15 December 2025

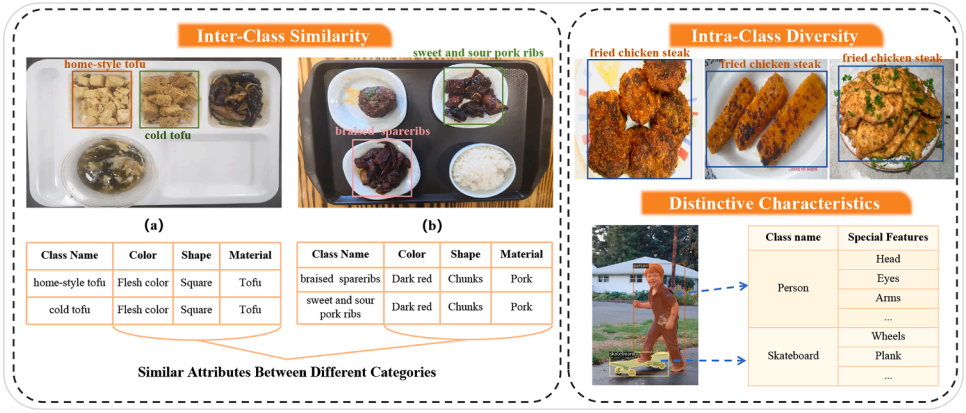

Fig. 1. Fine-grained visual challenges in ZSFD versus general object detection. Left: food objects exhibit high inter-class similarity (e.g., different tofu/spareribs dishes share color, shape, material). Right: food objects show high intra-class diversity (e.g., fried chicken steak varies by cooking style, plating). In contrast, general objects (e.g., person/skateboard) possess distinctive structural features that are consistent and discriminative.

In Generalized ZSD (GZSD), where seen and unseen categories coexist at inference, these difficulties are further exacerbated. While recent advancements in Large Language Models (LLMs) can provide richer semantic information [10]. However, effectively integrating them to resolve extreme visual-semantic ambiguities in the food domain remains an open challenge.

To this end, we propose Zero-Shot Food Detection with Semantic Space and Feature Fusion (ZeSF), a novel Zero-Shot Food Detection (ZSFD) framework. It features a purpose-built, dual-pathway "global-local" architecture. Our Multi-Scale Context Integration Module (MSCIM) is motivated by the compositional nature of food. It captures fine-grained ingredient details and balances local and global dish structures. Concurrently, the Contextual Text Feature Enhancement Module (CTFEM) mitigates high inter-class similarity. It achieves this by structuring LLM-generated descriptions into multi-granular representations, enabling precise discrimination. We posit that addressing these food-specific challenges will foster robust, generalizable principles. The core novelty of this work lies not in isolated components, but in a holistic design that tackles the dual challenges of fine-grained ZSFD via this synergistic architecture. Our main contributions are summarized as follows:

1. We propose ZeSF, a novel framework that aligns visual and textual features in a shared semantic space, ensuring strong generalization to unseen food categories while maintaining robust performance on seen ones.
2. To address the dual challenges of visual ambiguity and semantic overlap in food detection, we introduce two complementary modules. The MSCIM enhances visual representations by capturing both local details and global dish structure. Concurrently, the CTFEM structures culinary descriptions generated by LLMs into multi-granular embeddings. Together, these modules strengthen cross-modal alignment and improve discrimination under inter-class similarity and intra-class diversity.
3. Extensive experiments on UEC FOOD 256 and Food Objects With Attributes (FOWA) demonstrate superior ZSD and GZSD performance, while additional evaluations on MS COCO and PASCAL VOC confirm the effectiveness and efficiency of our integration strategy beyond the food domain.

The remainder of this paper is structured as follows: Section 2 reviews related works in ZSD and food computing. Section 3 details the architecture of the proposed ZeSF framework. Section 4 describes the experimental setup and presents the results and analysis. Finally, Section 5 concludes this work.

2. Related work

2.1. Zero-shot learning

Zero-Shot Learning (ZSL) recognizes unseen categories by aligning visual features with auxiliary semantics [11]. Early methods learned shared compatibility spaces with class prototypes [12] or defined attribute subspaces [13]. Recent Transformer-based approaches intensify visual-semantic fusion. Examples include progressive semantic injection to mitigate drift [14] and part-level grounding refinement [15]. Dynamic unary convolution in Transformers further enables adaptive receptive-field modulation [16]. This mechanism aligns closely with the multi-scale context modeling in our MSCIM. A parameter-efficient fine-tuning paradigm for Vision-Language Models (VLMs) is also relevant to the textual adaptation in CTFEM [17]. Despite these advances, a recent survey on fine-grained ZSL emphasizes that key challenges remain unresolved. These include subtle inter-class differences and high intra-class diversity [18]. Our work addresses these challenges in a detection setting. We mine structured, multi-attribute culinary descriptions from an LLM. We then couple them with a multi-dilation enhancement module. This strategy bridges semantic richness and spatial discriminability for fine-grained food detection.

2.2. Zero-shot object detection

Extending ZSL to localization tasks, ZSD methods are categorized into typically mapping-based [8,9,19] and generative-based [20,21] approaches. Mapping-based methods establish direct visual-semantic relationships. For instance, HRE was proposed to combine label and semantic embeddings [19]. Meanwhile, CZSD [9] enhanced unseen category detection through semantic-guided contrastive learning. SA [8] further advanced this by integrating class-adaptive contrastive loss into DEtection TRansformer (DETR) [22] for better alignment. Generative methods like GTNet [21] synthesize unseen features via GANs, or utilize synthetic data [23]. A key limitation of existing approaches is the seen-category bias. Models often misclassify unseen objects as seen categories, thus suppressing unseen performance. Our mapping-based approach alleviates this bias and addresses partial observation. This achieves a more balanced performance between seen and unseen categories, which is crucial for complex food data.

2.3. Zero-shot food detection

ZSFD targets the challenging task of localizing and identifying unseen food categories, which is complicated by fine-grained attributes and high intra-class diversity [7]. The predominant approach in this area

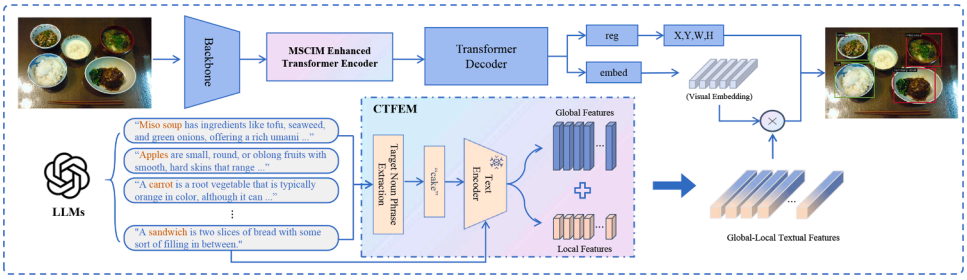

Fig. 2. Overall framework of our proposed ZeSF. An input image is processed by a backbone and the MSCIM-Enhanced Transformer Encoder to obtain multi-scale visual features. In parallel, CTFEM uses LLM-derived textual descriptions to form global and local semantic embeddings, which are then aligned with visual features for detection.

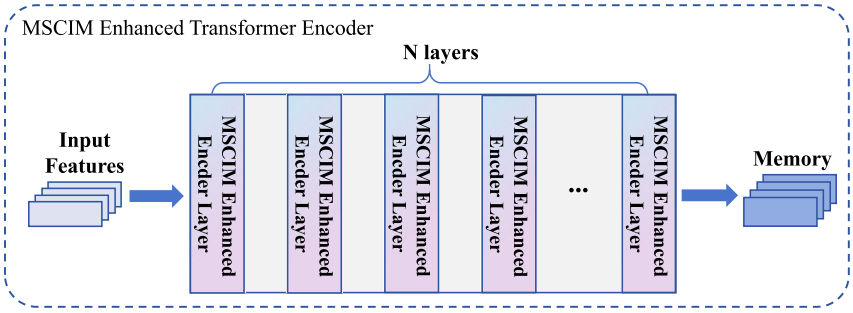

Fig. 3. Pipeline of the MSCIM-Enhanced Transformer Encoder, where an MSCIM is prepended to each of the N layers to enrich features for the standard MSDeform Attention.

relies on generative models to synthesize visual features for these unseen classes [7,24]. However, these methods struggle to produce authentic and discriminative synthetic features. In contrast, this paper proposes ZeSF, a mapping-based framework that circumvents feature generation. ZeSF directly aligns robust visual representations with rich semantic descriptions. Specifically, it leverages our multi-scale architecture to capture visual details and LLMs to generate comprehensive semantic cues. This direct alignment strategy offers a more effective paradigm for resolving the visual and semantic ambiguities inherent in food detection.

3. Method

3.1. Problem formulation

This study addresses the task of ZSFD (Zero-Shot Food Detection), aiming to accurately detect instances of novel food categories not seen during training. Given a training dataset $\mathcal{D}_s$ containing images I_i with object annotations (b_j, y_j) for N_s seen categories $\mathcal{Y}_s$, our objective is to train a model capable of detecting objects from both $\mathcal{Y}_s$ and a disjoint set of N_u unseen categories $\mathcal{Y}_u$ ($\mathcal{Y}_s \cap \mathcal{Y}_u = \emptyset$). Crucially, no training data is available for $\mathcal{Y}_u$. To facilitate the detection of these unseen classes, we leverage semantic descriptions $\mathcal{W} = \mathcal{W}_s \cup \mathcal{W}_u$ for all categories, where $\mathcal{W}_s \in \mathbb{R}^{N_s \times d}$ and $\mathcal{W}_u \in \mathbb{R}^{N_u \times d}$ denote the d-dimensional semantic embeddings for seen and unseen categories respectively. These descriptions, potentially generated by LLMs and processed by VLMs, enable aligning visual features with semantic representations for generalization. Evaluation is conducted on a test dataset $\mathcal{D}_t$ comprising objects from $\mathcal{Y}_s \cup \mathcal{Y}_u$. We consider two standard settings: ZSD, focused on detecting only $\mathcal{Y}_u$ objects, and GZSD (Generalized Zero-Shot Detection), which requires detecting objects from $\mathcal{Y}_s \cup \mathcal{Y}_u$ and is more challenging as it necessitates discrimination between seen and unseen classes during inference.

3.2. Scaling contextual understanding with transformers

As shown in Fig. 3, the MSCIM-Enhanced Transformer Encoder processes the input feature map $X \in \mathbb{R}^{B \times C \times H \times W}$ through N stacked layers, where B is the batch size, C the channel dimension, and H, W the spatial height and width. We adopt a Deformable DETR encoder and prepend a lightweight MSCIM (Multi-Scale Context Integration Module) to each encoder layer. As illustrated in Fig. 4, MSCIM acts as a pre-attention enhancer that produces a multi-dilation feature M, which is then fed into the standard MSDeform Attention [25]. Importantly, MSDeform Attention itself remains unchanged: its reference points, sampling offsets, and attention weights are learned independently of MSCIM.

MSCIM addresses the fine-grained yet cluttered statistics of images by implementing a SWDA-style sparse local self-attention operator [26] along three dilation pathways (d=1, 2, 3). For a query at (i, j), keys/values are sampled from:

$$\{(i', j') \mid i' = i + p \times d,\ j' = j + q \times d\}, \qquad -\frac{w}{2} \le p, q \le \frac{w}{2}, \tag{1}$$

and the output is computed via the scaled dot-product:

$$y_{ij} = \mathrm{Softmax}\left(\frac{Q_{ij} K_{ij,d}^{\top}}{\sqrt{d_k}}\right) V_{ij,d}, \tag{2}$$

where d_k is the key dimension. As shown in Fig. 4 right, the pathways cover complementary ranges: d=1 corresponds to local micro-texture, d=2 captures structural-global cues, and d=3 represents contextual-global layout. To best leverage these pathways under a fixed 8-head budget, we adopt a non-uniform allocation of $2, 3, 3$. With $C = 256$ and $h = 8$, each head has $d_h = 32$ channels, yielding per-path channel sizes of $[64, 96, 96]$. This allocation emphasizes structural and contextual cues with three heads each for $d = 2$ and $d = 3$, while assigning two heads to fine micro-texture at $d = 1$. Ablation studies validate this design as achieving the best seen-unseen trade-off (Table B.4).

Finally, the outputs from all eight dilated attention heads, $L_1, \ldots, L_8$, are integrated to form the MSCIM output M via:

$$M = W_O \cdot \mathrm{Concat}[L_1, L_2, \ldots, L_8] + b_O, \tag{3}$$

where $L_k \in \mathbb{R}^{B \times d_h \times H \times W}$ denotes the per-head feature map, h=8 is the total number of heads, and d_h=C/h is the head dimension. We view $W_O \in \mathbb{R}^{C \times C}$ with bias $b_O \in \mathbb{R}^C$ as a position-wise linear projection, equivalent to a 1×1 convolution, shared across spatial locations. The concatenation

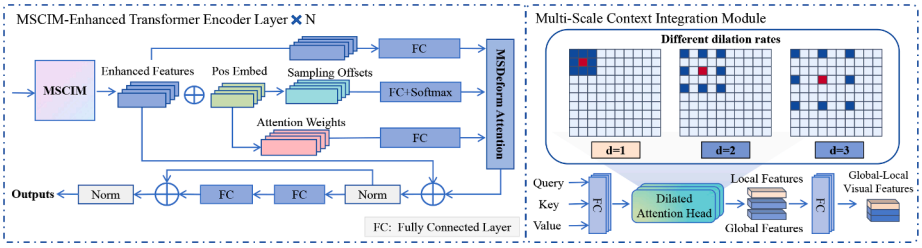

Fig. 4. Illustration of the MSCIM. Left: architecture of the MSCIM-Enhanced Transformer Encoder layer. The MSCIM block generates an enhanced feature map M via multi-dilation sparse self-attention, which is then consumed by the MSDeform Attention. Right: visualization of MSCIM's receptive fields. The three dilation pathways ($d = 1, 2, 3$) are designed to capture complementary features ranging from local micro-textures to global contextual layout.

is performed along the channel/head axis, producing $M \in \mathbb{R}^{B\times C\times H\times W}$ with the same spatial and channel dimensions as the input feature, now enriched with integrated multi-scale information. No additional normalization or gating is applied, keeping the enhancement lightweight. The feature M is then passed to the subsequent MSDeform Attention within the same encoder layer as shown in Fig. 4. Consequently, the only divisibility requirement is $C \bmod h = 0$, with C=256 and h=8 in our setup, and C does not need to be divisible by the number of dilation pathways, which is three. Implementation details are provided in Section 4.1.3.

3.3. Synthesizing context and detail for semantic refinement

Traditional ZSFD methods often rely on simplified text descriptions like "a photo of {class name}". This overlooks the rich visual and detailed characteristics of food items, crucial for distinguishing similar dishes. To capture these fine-grained details and enhance the model's semantic understanding, we incorporate LLM-enriched descriptions and introduce CTFEM (Contextual Text Feature Enhancement Module) to structurally fuse them.

3.3.1. LLM-based semantic description generation

To create powerful semantic representations that move beyond simple class labels, we utilized a LLM, GPT-4o, to generate detailed textual descriptions for each food category. These generated descriptions are designed to be rich and informative, capturing a wide array of observable characteristics such as key ingredients, colors, textures, and common preparation styles. For example, the category "braised spareribs" can be described as: "Tender pork ribs slow-cooked in a savory soy-based sauce until caramelized, with a glossy dark brown glaze, chunks of highly tender meat, often accompanied by onions or ginger." This provides a far deeper semantic understanding than the class name alone.

The high quality of these LLM-generated descriptions allows them to be used directly to create more informative semantic vectors without any manual post-processing. This enhanced semantic information is critical for enabling the model to accurately distinguish fine-grained categories and recognize novel food items. A detailed analysis of the impact of different text generation methods is presented in Section 4.5.4.

3.3.2. Contextual textual feature enhancement module

Effectively leveraging detailed semantic descriptions for ZSFD requires prioritizing relevant cues. While the core category name, serving as a local feature, is vital for identification, the descriptive content, serving as a global feature, provides crucial details for fine-grained distinction. To balance these aspects, we propose a dual-granularity global-local semantic fusion strategy within the CTFEM to generate enhanced textual representations.

As illustrated in Fig. 2, CTFEM integrates global and local textual features by extracting them from the LLM-generated descriptions.

Firstly, the global textual feature P_G is extracted from the entire detailed description D using the pre-trained CLIP text encoder ψ_{CLIP}, as it is known to produce effective textual embeddings aligned with visual features:

$$P_G = \psi_{\text{CLIP}}(D). \tag{4}$$

Secondly, to obtain the local textual feature P_L, we utilize the CLIP text encoder on the core category name cls:

$$P_L = \psi_{\text{CLIP}}(cls). \tag{5}$$

The category name cls is precisely defined as the primary noun phrase directly matching the predefined class label from our dataset used in the LLM prompt, ensuring P_L strictly represents the intended category's embedding. Finally, the enhanced global-local semantic vector P is a weighted sum of P_G and P_L:

$$P = \gamma P_G + (1-\gamma)P_L. \tag{6}$$

Here, $\gamma \in [0, 1]$ is a coefficient balancing global and local contributions, fixed once and reused on a validation set. This global-local weighting effectively combines rich contextual details with the precise category identity, yielding a more comprehensive and discriminative representation for diverse and visually similar food items. We opt for this straightforward linear combination as it provides an effective and interpretable way to balance the two feature types without introducing additional network parameters or risking overfitting.

A fixed, parameter-free fusion with $\gamma = 0.6$ is used across all datasets. We found this simple approach avoids the overfitting observed with learnable gating, preserves the interpretability of global vs. local roles, and enables a "semantic-capacity amplifier" effect when using rich LLM descriptions.

3.4. Loss function

Our composite loss function is a weighted sum of three components designed to address classification, localization, and zero-shot knowledge transfer:

$$\mathcal{L} = \lambda_{\text{cls}}\mathcal{L}_{\text{cls}} + \lambda_{\text{bbox}}\mathcal{L}_{\text{bbox}} + \lambda_{\text{con}}\mathcal{L}_{\text{con}}, \tag{7}$$

where the localization term is $\mathcal{L}_{\text{bbox}} = \lambda_{\text{iou}}\mathcal{L}_{\text{iou}} + \lambda_{\text{giou}}\mathcal{L}_{\text{giou}}$.

The loss components are weighted to reflect the challenges of fine-grained food detection. First, we use focal loss $\mathcal{L}_{\text{cls}}$ to handle class imbalance, with $\lambda_{\text{cls}} = 1.0$. Second, for localization, we combine IoU and GIoU to address deformable food items: GIoU supplies stable gradients for low-overlap boxes in early training, while IoU refines the fit. We assign higher weights, $\lambda_{\text{iou}} = 5.0$ and $\lambda_{\text{giou}} = 2.0$, to prioritize accurate cropping of ingredients and context. Finally, to enable generalization, we incorporate the class-aware contrastive loss $\mathcal{L}_{\text{con}}$ from [8] with a small weight of $\lambda_{\text{con}} = 0.05$, which gently guides visual–semantic alignment for unseen classes without over-constraining feature variance or harming seen-class performance.

Table 1
Performance comparison (%) on UEC FOOD 256 under ZSD and GZSD settings. † indicates methods based on generative models.

Metric	Method	ZSD	GZSD		
			Seen	Unseen	HM
Recall@100	PL [32]	56.5	53.2	40.4	46.0
	BLC [33]	58.9	55.3	43.8	48.9
	CZSD [9]	60.7	57.6	45.5	50.8
	SU† [34]	61.9	52.5	52.8	52.6
	RRFS† [35]	64.8	54.9	55.1	55.0
	SeeDS† [7]	74.0	55.2	61.4	58.1
	ZSFDet† [24]	74.4	57.0	61.8	59.3
	SA [8]	92.9	71.2	**86.9**	78.3
	Ours	**93.7**	**86.9**	86.4	**86.7**
mAP	PL [32]	14.5	18.9	11.6	14.4
	BLC [33]	19.2	20.5	15.2	17.5
	CZSD [9]	22.0	20.8	16.2	18.2
	SU† [34]	22.4	19.3	20.1	19.7
	RRFS† [35]	23.6	20.1	22.9	21.4
	SeeDS† [7]	27.1	20.2	26.0	22.7
	ZSFDet† [24]	27.3	21.9	26.1	23.8
	SA [8]	24.2	18.6	24.9	21.3
	Ours	**29.1**	**22.2**	**27.0**	**24.4**

Table 2
Performance comparison (%) on FOWA under ZSD and GZSD settings. † indicates methods based on generative models.

Metric	Method	ZSD	GZSD		
			Seen	Unseen	HM
Recall@100	ConSE [36]	39.7	58.0	38.1	46.4
	PL [32]	40.1	53.9	39.6	45.7
	BLC [33]	41.2	55.3	40.5	46.8
	CZSD [9]	48.0	86.1	44.8	58.9
	SU† [34]	45.3	82.3	44.1	57.4
	RRFS† [35]	48.8	86.6	47.6	61.4
	SeeDS† [7]	52.9	87.0	49.8	63.3
	ZSFDet† [24]	53.5	87.0	50.1	63.6
	SA [8]	89.3	**97.5**	65.5	78.3
	Ours	**90.9**	96.8	**73.9**	**83.8**
mAP	ConSE [36]	0.8	54.3	0.7	1.4
	PL [32]	1.0	50.8	0.7	1.4
	BLC [33]	1.1	51.1	0.9	1.8
	CZSD [9]	4.0	81.2	2.1	4.1
	SU† [34]	3.9	79.1	2.3	4.5
	RRFS† [35]	4.3	82.7	2.7	5.2
	SeeDS† [7]	5.9	82.8	3.5	6.7
	ZSFDet† [24]	6.1	82.8	3.6	6.9
	SA [8]	7.7	**90.5**	5.4	10.1
	Ours	**10.2**	89.8	**8.8**	**16.0**

4. Experiments

4.1. Experimental settings

4.1.1. Benchmark datasets

We evaluate our framework on two fine-grained food datasets: UEC FOOD 256 [27] and FOWA [24]. To validate the generalization, we also report results on the large-scale MS COCO [28] and PASCAL VOC [29]. Table A.2 summarizes the statistics and class splits for all datasets.

4.1.2. Evaluation metrics

We follow standard ZSD (Zero-Shot Detection) and GZSD (Generalized Zero-Shot Detection) detection protocols with a disjoint split of classes into seen ($\mathcal{Y}_s$) and unseen ($\mathcal{Y}_u$). In ZSD we report the unseen mean Average Precision (mAP) at IoU 0.5 (mAP on $\mathcal{Y}_u$). In GZSD we report seen mAP (S), unseen mAP (U), and their Harmonic Mean (HM, $\mathrm{HM} = \frac{2SU}{S+U}$) as the primary balanced metric. For UEC FOOD 256, FOWA and PASCAL VOC, we report mAP and Recall@100 at IoU 0.5. For MS COCO, we additionally report Recall@100 at IoU 0.4 and 0.6 to probe localization robustness.

4.1.3. Implementation details

We use a ResNet-50 backbone [30] for all datasets (Swin-T [31] for PASCAL VOC). Training adopts multi-scale resizing. Specifically, the shorter side ranges from 480 to 800 pixels, and the longer side is capped at 1333 pixels. During inference, the shorter side is fixed at 800 pixels. Features are channel-mapped to $C = 256$ for an 8-head Transformer. For the MSCIM (Multi-Scale Context Integration Module), we apply a non-uniform head split of 2, 3, 3 ($d = 1, 2, 3$). Similarly, the CTFEM (Contextual Text Feature Enhancement Module) uses a fixed weight $\gamma = 0.6$. Complete hyperparameters, including augmentation, optimizer, and loss weights, are detailed in Appendix Table A.1.

4.2. Comparisons with the state-of-the-art method

4.2.1. Results on UEC FOOD 256

Table 1 presents the performance comparison on UEC FOOD 256. Compared to the mapping-based baseline SA, our method improves the ZSD mAP from 24.2% to 29.1% and the GZSD HM from 21.3% to 24.4%. Furthermore, ZeSF establishes a favorable balance against generative methods such as ZSFDet. While achieving a higher Unseen mAP, our model also boosts the Seen Recall@100 from 57.0% to 86.9%. This results in a higher HM, indicating that our approach improves generalization without compromising performance on seen classes.

4.2.2. Results on FOWA

Table 2 presents the results on the FOWA dataset, where ZeSF outperforms baselines including SA, ZSFDet, and CZSD. While SA achieves a higher seen-class mAP, our method improves unseen-class performance from 5.4% to 8.8%. This results in a higher GZSD HM of 16.0% compared to 10.1%, indicating a more robust generalized model.

Compared to generative methods such as ZSFDet, our direct-mapping approach achieves a higher HM score of 16.0% compared to 6.9%. This suggests advantages in feature quality by avoiding low-fidelity synthesis. Fig. 6 visually corroborates this claim, where ZeSF yields compact, well-separated clusters for unseen classes on UEC FOOD 256 and FOWA, unlike the scattered overlaps observed in SA.

4.2.3. Multi-domain validation: MS COCO & PASCAL VOC

Although ZeSF is food-centric, we evaluate its transferability to general objects to validate the broader applicability of its principles, namely multi-scale visual aggregation and dual-granularity semantic fusion. These experiments indicate generalization without claiming universal superiority.

Evaluations on MS COCO (with 48/17 and 65/15 splits) and PASCAL VOC (with a 16/4 split) follow standard ZSD protocols. No hyperparameters or architecture were retuned. Only class descriptions were regenerated using the same offline LLM pipeline as for food datasets.

Tables 3–5 present the results. On MS COCO ZSD, ZeSF achieves 22.0% and 24.3% mAP on the 48/17 and 65/15 splits. This represents improvements of +2.5% and +0.3% over SA. For GZSD, it yields the highest HM for Recall@100, 66.1% and 69.2%, and for mAP, 22.8% and 27.8%, showing balanced seen-unseen calibration. On PASCAL VOC, ZeSF attains 69.6% ZSD mAP and 57.0% GZSD HM. It boosts Seen from 64.8% to 67.2% while maintaining Unseen at 49.5% compared to 49.3%.

While effective, these gains are notably smaller than those on food datasets. For example, the GZSD HM improvements are +0.1% and +0.9% on COCO and +1.0% on VOC, compared to +3.1% and +5.9% on food datasets. This empirically validates the food-centric optimization. Specifically, MSCIM's multi-dilation excels on food's compositional hierarchy but offers limited extra value for rigid general objects.

Table 3
Performance comparison (%) on MS COCO under ZSD setting. † indicates methods based on generative models.

Method	Split	Recall@100			mAP	Split	Recall@100			mAP
		IoU = 0.4	IoU = 0.5	IoU = 0.6	IoU = 0.5		IoU = 0.4	IoU = 0.5	IoU = 0.6	IoU = 0.5
CZSD [9]	48/17	56.1	52.4	47.2	12.5	65/15	62.3	59.5	55.1	18.6
SU† [34]	48/17	–	–	–	–	65/15	54.4	54.0	47.0	19.0
PL [32]	48/17	–	43.5	–	10.1	65/15	–	37.7	–	12.4
BLC [33]	48/17	51.3	48.8	45.0	10.6	65/15	57.2	54.7	51.2	14.7
RRFS† [35]	48/17	58.1	53.5	47.9	13.4	65/15	65.3	62.3	55.9	19.8
TCB [37]	48/17	55.5	52.4	48.1	11.4	65/15	62.5	59.9	55.1	13.8
SeeDS† [7]	48/17	59.2	55.3	48.5	14.0	65/15	66.4	63.8	56.5	20.1
ZSFDet† [24]	48/17	58.6	54.7	48.3	14.0	65/15	66.5	64.2	56.7	20.3
SA [8]	48/17	76.7	73.0	68.8	19.5	65/15	88.0	85.3	81.9	24.0
M-RRFS† [20]	48/17	64.0	60.9	55.5	15.1	65/15	68.6	65.4	59.1	20.3
Ours	**48/17**	**86.3**	**82.9**	**79.0**	**22.0**	**65/15**	**89.1**	**86.5**	**83.4**	**24.3**

Table 4
Performance comparison (%) on MS COCO under GZSD settings. † indicates methods based on generative models.

Method	Split	Recall@100			mAP			Split	Recall@100			mAP		
		Seen	Unseen	HM	Seen	Unseen	HM		Seen	Unseen	HM	Seen	Unseen	HM
CZSD [9]	48/17	65.7	52.4	58.3	45.1	6.3	11.1	65/15	62.9	58.6	60.7	**40.2**	16.5	23.4
SU† [34]	48/17	–	–	–	–	–	–	65/15	57.7	53.9	55.7	36.9	19.0	25.1
PL [32]	48/17	38.2	26.3	3.2	35.9	4.1	7.4	65/15	36.4	37.2	36.8	34.1	12.4	18.2
BLC [33]	48/17	57.6	46.4	51.4	42.1	4.5	8.1	65/15	56.4	51.2	53.9	36.0	13.1	19.2
RRFS† [35]	48/17	59.7	58.8	59.2	42.3	13.4	20.4	65/15	58.6	61.8	60.2	37.4	19.8	26.0
TCB [37]	48/17	71.9	52.4	60.6	47.3	4.9	8.8	65/15	69.3	59.8	64.2	39.9	13.8	20.5
SeeDS† [7]	48/17	60.1	**60.8**	60.5	42.5	14.5	21.6	65/15	59.3	62.5	60.9	37.5	20.3	26.3
ZSFDet† [24]	48/17	60.1	60.7	60.4	42.5	14.3	21.4	65/15	59.3	**63.1**	61.1	37.5	20.5	26.5
SA [8]	48/17	78.4	49.7	60.8	34.0	**17.0**	22.7	65/15	79.7	55.8	65.7	35.5	21.7	26.9
M-RRFS† [20]	48/17	63.3	60.1	61.7	42.7	15.0	22.2	65/15	67.0	63.7	65.3	37.9	19.8	26.0
Ours	48/17	**86.6**	53.4	**66.1**	**51.5**	14.6	**22.8**	65/15	**81.5**	60.2	**69.2**	37.0	**22.3**	**27.8**

Table 5
Performance comparison (%) on PASCAL VOC under ZSD and GZSD settings. † indicates methods based on generative models.

Method	ZSD	GZSD		
		Seen	Unseen	HM
ConSE	52.1	59.3	22.3	32.4
PL [32]	62.1	–	–	–
BLC [33]	55.2	58.2	22.9	32.9
SU† [34]	64.9	–	–	–
CZSD [9]	65.7	63.2	46.5	53.8
RRFS† [35]	65.5	47.1	49.1	48.1
SeeDS† [7]	68.9	48.5	50.6	49.5
ZSFDet† [24]	69.2	48.5	50.8	49.6
M-RRFS† [20]	67.0	48.5	**52.6**	50.5
TCB [37]	59.3	61.0	29.8	40.0
SA [8]	68.7	64.8	49.3	56.0
Ours	**69.6**	**67.2**	49.5	**57.0**

Similarly, CTFEM's fusion is particularly effective with culinary semantics but less so without preparation-induced diversity.

4.2.4. Qualitative evaluation

Fig. 5 provides a qualitative comparison against the baseline SA and a representative generative method RRFS. It illustrates the generalization capability of our method on UEC FOOD 256 and FOWA, highlighting our framework's effectiveness in challenging fine-grained scenarios.

A primary observation is our model's ability to detect unseen categories in red bounding boxes, which baseline methods often miss. For example, in the first row of UEC FOOD 256, our method accurately localizes and classifies the unseen dish "dry curry". In contrast, the baseline incorrectly predicts it as "fried rice." On FOWA, our approach detects multiple unseen food items within complex tray settings, such as "celery tofu" in the third row. These items are overlooked by the baseline.

Furthermore, beyond just detecting unseen classes, our method also indicates higher confidence and more precise localization for seen categories. As shown in the fourth and fifth rows, our model's bounding boxes are often tighter and more accurately placed compared to the SA and RRFS. These results clearly show that our framework enhances ZSD capabilities, leading to more reliable food recognition.

4.3. Comparison with open-vocabulary detector

To validate our framework's advantages over general-purpose vision-language detectors, we compare against Grounding DINO [38]. As shown in Table C.7, zero-shot Grounding DINO achieves 5.6 % HM on FOWA. Fine-tuning on seen classes improves seen mAP to 63.6 % after 15 epochs but negatively impacts unseen mAP, dropping from 7.2 % to 1.3 %. This reduces HM to 2.55 %. In contrast, our method achieves 16.0 % HM with balanced performance, reaching 89.8 % on seen classes and 8.8 % on unseen classes. This validates that fine-grained ZSFD requires domain-optimized architectures such as MSCIM and CTFEM, together with explicit semantic guidance from LLM-generated descriptions, which prevent the forgetting observed in Grounding DINO. See C.1 for detailed analysis.

4.4. Feature visualization

The t-SNE [39] visualizations in Fig. 6 confirm our method's ability to effectively learn high-fidelity features for unseen classes. The baseline SA's features are scattered and overlapping, yielding low silhouette scores of 0.304 on UEC FOOD 256 and 0.331 on FOWA. In contrast, our method produces more compact and well-separated clusters, improving these scores to 0.532 and 0.551, respectively. This improved feature space directly results from our framework's synergistic design. It effectively aligns discriminative visual features from MSCIM with rich semantic embeddings from CTFEM to enforce clear inter-class boundaries.

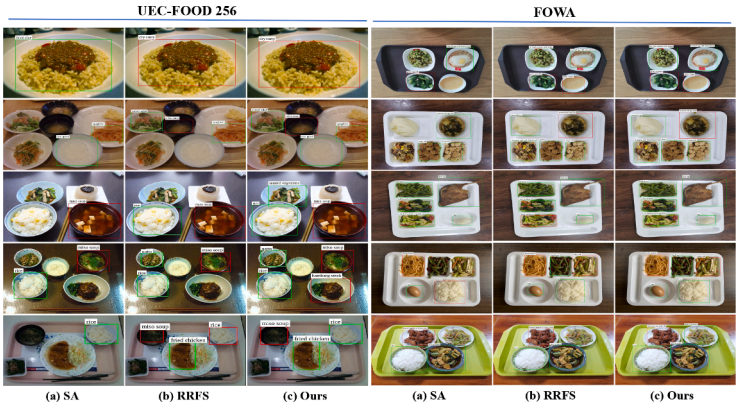

Fig. 5. Qualitative results on UEC-FOOD 256 (left) and FOWA (right). Our method (c) is compared against baselines SA (a) and RRFS (b). Red and green boxes denote unseen and seen classes, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

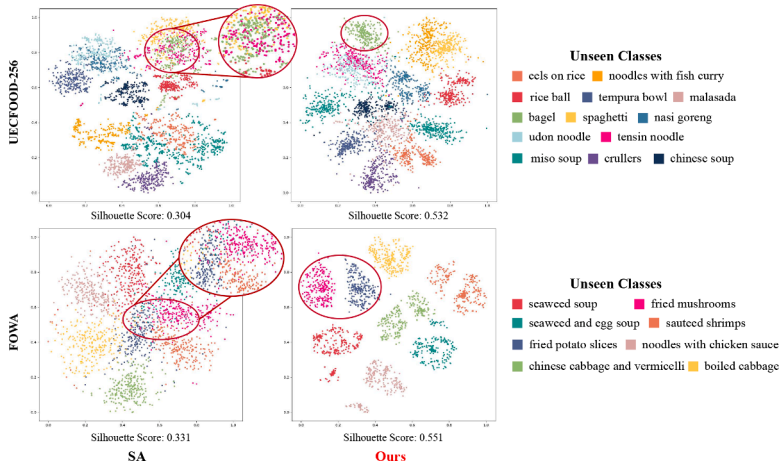

Fig. 6. Comparison of t-SNE visualizations of unseen class feature embeddings. The baseline SA (left) exhibits scattered and overlapping distributions. In contrast, our method (right) forms compact and well-separated clusters, increasing the silhouette score from 0.304 to 0.532 on UEC FOOD 256.

Table 6
Ablation study (%) on the contribution of different components on UEC FOOD 256 and FOWA.

Dataset	Components			ZSD	GZSD		
	MSCIM	CTFEM	LLM Prompts		Seen	Unseen	HM
UEC FOOD 256	–	–	–	24.2	18.6	24.9	21.3
	✓	–	–	25.8	17.3	25.6	20.6
	–	✓	–	25.4	17.6	26.3	21.1
	–	–	✓	26.8	18.5	25.5	21.4
	✓	✓	–	25.8	18.4	25.8	21.5
	✓	–	✓	26.4	18.9	26.0	21.9
	–	✓	✓	27.6	19.0	26.8	22.2
	✓	✓	✓	**29.1**	**22.2**	**27.0**	**24.4**
FOWA	–	–	–	7.7	**90.5**	5.4	10.1
	✓	–	–	8.0	86.4	5.9	11.0
	–	✓	–	7.9	89.9	6.8	12.7
	–	–	✓	9.0	90.4	7.2	13.3
	✓	✓	–	8.0	89.4	7.1	13.1
	✓	–	✓	9.1	86.8	8.6	15.6
	–	✓	✓	9.8	90.1	8.2	15.1
	✓	✓	✓	**10.2**	89.8	**8.8**	**16.0**

4.5. Ablation experiments

To analyze component effectiveness, we conduct ablation experiments on both UEC FOOD 256 and FOWA, with results reported in Table 6.

4.5.1. Component effectiveness analysis

Under plain class-name semantics, neither MSCIM nor CTFEM alone yields large HM gains as shown in Table 6. This aligns with their design as semantic-capacity amplifiers rather than standalone boosters. Both modules reserve representational headroom for enriched LLM descriptions. Specifically, these descriptions encode plating structure, ingredient co-occurrence, and preparation cues. Such details are absent in hand-crafted prompts such as “a photo of a {class}”.

With LLM-generated descriptions, their coordination produces substantial improvements. On UEC FOOD 256, HM increases from 21.3 % to 24.4 %, and on FOWA, from 10.1 % to 16.0 %. This reflects MSCIM’s multi-scale receptive aggregation grounding enriched semantics, and CTFEM’s global-local fusion stabilizing margins. The modest standalone shifts are intentional, tuned for structured semantic infusion. This is corroborated by the t-SNE visualizations discussed in Section 4.4, which confirm higher-fidelity representations. Further validation is provided by the semantic scaling analysis in Section 4.5.4 and efficiency analysis in Section 4.6. These results indicate that the gains stem from principled co-design rather than new atomic operators.

4.5.2. Effectiveness of MSCIM

The comparison between our full model and the variant without MSCIM reveals consistent performance variations. To further validate the internal design, we conducted additional ablation on the parallel branches as reported in Table B.3. The removal of any single dilation branch, whether $d = 1$, $d = 2$, or $d = 3$, leads to performance

degradation compared to the full three-branch configuration. Notably, removing the $d = 2$ branch causes the most significant decline, indicating the importance of intermediate-scale feature aggregation.

To evaluate head allocation variants, we ablated on UEC FOOD 256 under ZSD and GZSD settings. The balanced configuration of 2, 3, and 3 achieves the highest scores. Specifically, it yields 29.1 % for ZSD and 24.4 % for GZSD HM. This outperforms alternatives like the split of 2, 4, and 2 with an HM of 23.1 % and the split of 2, 2, and 4 with an HM of 22.0 %. The variant without MSCIM yields an HM of 22.2 %, while unbalanced splits such as 4, 2, and 2 reach 17.6 %, showing notable drops. Relative ΔHM values further underscore the stability and effectiveness of the 2, 3, and 3 split.

4.5.3. Effectiveness of CTFEM

The CTFEM module indicates its value through structured processing of detailed LLM-generated descriptions. The observed performance variations in the configuration without CTFEM indicate the importance of converting rich textual descriptions into structured global and local semantic features. This provides more robust representations than basic embeddings for fine-grained class discrimination.

4.5.4. Impact of LLM-generated descriptions

We validate the use of LLM-generated descriptions through two ablations detailed in B.3. First, framework performance scales with the quality of the semantic source as shown in Table B.5. Upgrading from handcrafted prompts to GPT-4o yields a GZSD HM gain of +3.1 % on UEC FOOD 256. Second, our approach outperforms prompt-learning methods such as CoOp. These results confirm that explicit culinary knowledge from LLMs is essential for ZSFD. It cannot be effectively replaced by learned continuous prompts.

4.6. Computational cost analysis

A detailed cost analysis validates the efficiency of our framework as provided in C.3. As shown in Table C.9, ZeSF achieves a favorable balance between performance and cost. It outperforms the baseline SA with faster inference speed and is more compute-efficient than heavier generative models. Its reliance on LLMs is a practical offline step with no inference cost.

4.7. Focusing where it matters

To further validate the effectiveness of MSCIM, we use Grad-CAM [40] to visualize attention regions in Fig. D.2. The visualizations reveal that integrating MSCIM enhances target localization. In contrast to the scattered attention of the baseline, our model concentrates focus on relevant object regions. This allows it to capture fine-grained details and identify multiple objects in complex scenes. Consequently, the attention is more continuous and accurate.

4.8. Impact of semantic quality on performance

A significant finding is that the framework's performance scales with the quality of the LLM-generated descriptions. Upgrading from handcrafted templates to GPT-4o descriptions yields GZSD HM improvements. Specifically, it provides a gain of +2.9 % on UEC FOOD 256 and +2.9 % on FOWA. Interestingly, GPT-3.5 reveals a trade-off. It shows a slight HM dip on UEC FOOD 256 but improves the unseen-class mAP by +1.9 % in ZSD. This underscores the sensitivity of the model to semantic granularity. This reliance on LLMs is a practical, one-time offline preprocessing step that has no impact on inference speed or cost. A full discussion is provided in B.2.

5. Conclusion

Our primary contribution is ZeSF, a visual-semantic framework specifically tailored to fine-grained ZSFD (Zero-Shot Food Detection) and validated on UEC FOOD 256 and FOWA. Its design principles, namely expanding receptive fields through the MSCIM (Multi-Scale Context Integration Module) and fusing global with local semantics, also yield consistent gains on MS COCO and PASCAL VOC. This indicates transfer beyond food-specific distributions without dataset-specific tuning. The synergy between visual encoding at multiple scales and semantic enrichment across dual granularities offers a valuable blueprint for other researchers. The modular nature of MSCIM and CTFEM (Contextual Text Feature Enhancement Module) makes them transferable components for tackling fine-grained zero-shot challenges in other domains, such as retail product recognition, biological species identification, or defect detection in manufacturing. For complementary evidence, Appendix Table C.8 reports class-wise APs on challenging categories.

Despite these strengths, we identify several avenues for future improvement as well as acknowledge the following limitations. First, the framework's performance is intrinsically linked to the descriptive quality of the upstream LLM. While this shows a desirable "future-proof" scalability that benefits from advancements in language modeling, it also highlights a practical dependency that must be considered for deployment. Second, our computational analysis reveals that while ZeSF achieves faster inference than the baseline, its absolute throughput may still challenge applications demanding extreme efficiency, such as real-time processing on edge devices. These system-level limitations, along with specific performance challenges, directly inform our future research agenda.

We specifically aim to address performance issues identified in our failure case analysis as shown in Fig. D.1, such as struggles with severe occlusion and high visual ambiguity. To this end, future work will explore lightweight attention mechanisms and knowledge distillation to build a more efficient model. This will reduce dependency on large-scale models. Application-wise, we aim to deploy and refine ZeSF for real-world intelligent dining systems, explicitly tackling robustness to varied lighting conditions and occlusions. We also plan to extend the framework from detection to fine-grained ingredient recognition and quantity estimation. This is an essential step toward real-time nutritional analysis and truly intelligent food-centric ecosystems.

CRediT authorship contribution statement

Xinlong Wang: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Conceptualization; **Weiqing Min:** Supervision, Project administration; **Guorui Sheng:** Supervision, Project administration, Conceptualization; **Jingru Song:** Writing – review & editing, Supervision, Methodology; **Yancun Yang:** Methodology; **Tao Yao:** Supervision; **Shuqiang Jiang:** Supervision, Project administration, Conceptualization.

Data availability

Data will be made available on request.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The study was supported by the Beijing Natural Science Foundation (JQ24021) and the National Natural Science Foundation of China (62125207 and 62472411).

Table A.1
Additional reproducibility details from the configuration.

Aspect	Setting
Backbone	ResNet-50
Neck out channels	256
Transformer Encoder Decoder layers	6
Queries	900
Normalization (Backbone)	FrozenBN
Normalization (Neck)	GroupNorm
Image format	RGB
Input Channels	3
Optimizer	AdamW
Scheduler	MultiStepParam
Milestones	90,000
Iterations	30,000
Batch size	8
Eval / checkpoint period	2000 iters
Gradient clipping	max_norm = 0.1
Model EMA	Enabled
Contrastive temperature	$\tau = 20.0$
NMS (postproc)	0.7

Table A.2
Dataset statistics for all benchmarks. "Annotations" counts labeled bounding boxes. Splits: UEC FOOD 256 (205/51), FOWA (184/44), MS COCO 2014 (65/15), PASCAL VOC 2007 + 2012 (16/4).

Datasets	Classes		Annotations	Images		
	Seen	Unseen		Train	Test	Total
UEC FOOD 256	205	51	28,429	20,452	5732	26,184
FOWA	184	44	95,322	10,463	10,140	20,603
MS COCO	65	15	548,745	62,300	10,815	73,115
PASCAL VOC	16	4	52,090	10,728	10,834	21,562

Appendix A. Experimental details

A.1. Implementation details

Table A.1 provides additional reproducibility details from our active configuration. All settings are shared by both ZSD and GSD evaluations across every dataset, with no hyperparameter changes between modes.

A.2. Dataset statistics

Table A.2 summarizes the statistics for all benchmarks used in our experiments.

Appendix B. Additional ablation studies

B.1. Detailed MSCIM ablations

We provide detailed ablation studies on the architectural components of MSCIM. Table B.3 analyzes the contribution of each dilation branch. Table B.4 evaluates different head allocation strategies for MSCIM.

B.2. Semantic sensitivity discussion

This section provides a detailed analysis of the framework's sensitivity to the quality of semantic descriptions, as summarized in the main paper. A key finding from our semantic source ablation (Section 4.5.4) is that the framework's performance scales positively with the quality of LLM-generated descriptions. Upgrading from GPT-3.5-Turbo to GPT-4o yields a significant GZSD HM improvement of 3.5 % on UEC FOOD 256 and 1.2 % on FOWA. Interestingly, this scaling effect is not uniform: on the highly fine-grained UEC FOOD 256 dataset, GPT-3.5 descriptions improve ZSD (+1.9 %) and GZSD Unseen (+1.3 %) scores over the baseline but reduce seen-class performance, leading to a slight HM decrease (-0.6 %). This suggests not architectural fragility, but a shifting trade-off between seen-class discrimination and unseen-class generalization, depending on the granularity of semantic input. We interpret this as a characteristic of an architecture optimized to leverage rich semantic information. While simple hand-crafted prompts provide a robust but low-ceiling baseline, our framework is designed to parse complex details. Less discriminative descriptions, such as those from GPT-3.5 for nuanced food categories, may not fully activate the model's specialized mechanisms, whereas the highly structured semantics from GPT-4o align well with the design, resolving the trade-off and substantially lifting overall performance. This positions our approach as a scalable

Table B.3
Ablation study (%) on the contribution of different dilation branches within the MSCIM.

Dataset	Different dilation	ZSD	GZSD		
			Seen	Unseen	HM
UEC FOOD 256	Without dilation $d = 1$	27.7	19.8	26.7	22.8
	Without dilation $d = 2$	26.7	17.9	26.2	21.2
	Without dilation $d = 3$	28.5	20.4	26.9	23.2
	Ours	**29.1**	**22.2**	**27.0**	**24.4**
FOWA	Without dilation $d = 1$	8.7	85.6	7.7	14.1
	Without dilation $d = 2$	8.4	89.6	6.8	12.6
	Without dilation $d = 3$	9.2	**89.9**	8.0	14.8
	Ours	**10.2**	89.8	**8.8**	**16.0**

Table B.4
Ablation study of head allocation variants for MSCIM on UEC FOOD 256: performance comparison (%) under ZSD and GZSD settings. ΔHM is the relative change in HM compared to the best split (2,3,3).

Head Split ($d = 1, 2, 3$)	ZSD	GZSD		
		Seen	Unseen	HM
Without MSCIM	27.6	19.0	26.8	22.2
(2,2,4)	28.5	18.7	26.9	22.0
(2,4,2)	28.9	20.2	27.0	23.1
(4,2,2)	26.5	14.3	23.0	17.6
(3,3,2)	26.8	16.3	25.9	20.0
(2,3,3)	**29.1**	**22.2**	**27.0**	**24.4**

Table B.5
Ablation study (%) on the impact of textual auxiliary information generation methods.

Dataset	LLM Prompt	LLM Model	ZSD	GZSD		
				Seen	Unseen	HM
UEC FOOD 256	✘	–	25.8	18.4	25.8	21.5
	✓	GPT-3.5-Turbo	27.7	17.0	**27.1**	20.9
	✓	GPT-4o	**29.1**	**22.2**	27.0	**24.4**
FOWA	✘	–	8.0	89.4	7.1	13.1
	✓	GPT-3.5-Turbo	8.7	89.2	8.1	14.8
	✓	GPT-4o	**10.2**	**89.8**	**8.8**	**16.0**

Table B.6
Ablation study (%) on UEC FOOD 256 under ZSD and GZSD settings: impact of different class description strategies. All variants share the same detector backbone, optimization schedule, and data augmentation, differing only in the class description strategy. CoOp results are the mean result over 3 random seeds.

Method	ZSD	GZSD		
		Seen	Unseen	HM
Hand-crafted Prompts	24.2	18.6	24.9	21.3
CoOp (Context tokens = 8)	24.4	17.6	23.9	20.2
CoOp (Context tokens = 16)	25.4	18.2	24.4	20.8
LLM-generated Prompts	**29.1**	**22.2**	**27.0**	**24.4**

and forward-compatible system that can readily benefit from future improvements in language models without architectural modification.

The practical viability of this approach is supported by two factors. First, the use of LLMs is a one-time, offline preprocessing step, eliminating runtime costs or API dependencies during inference. Second, the increasing availability and power of capable open-source LLMs, such as Llama 3, significantly enhance the long-term accessibility of this methodology. While this study establishes the core principle, we acknowledge that a comprehensive robustness analysis remains a broad undertaking. A systematic evaluation across a wider spectrum of open-source models and the exploration of targeted prompt engineering techniques to enhance adaptability in resource-constrained settings are valuable directions for future work.

B.3. Semantic source ablations

To isolate the impact of textual auxiliary information, we compare three semantic sources: a hand-crafted prompt template ("a photo of a [class]"), GPT-3.5-Turbo generated descriptions, and GPT-4o generated descriptions. As shown in Table B.5, replacing the template with GPT-3.5-Turbo yields mixed but informative effects: on UEC FOOD 256, ZSD increases from 25.8 % to 27.7 % and the GZSD Unseen score rises from 25.8 % to 27.1 % (+1.3 %), but the Seen score drops (18.4 % → 17.0 %, -1.4 %) leading to a slight HM decrease (21.5 % → 20.9 %, -0.6 %). On FOWA, however, GPT-3.5-Turbo improves all open-set sensitive metrics, including ZSD (8.0 % → 8.7 %, +0.7 %), Unseen (7.1 % → 8.1 %, +1.0 %), and HM (13.1 % → 14.8 %, +1.7 %). Upgrading to GPT-4o produces consistent gains across both datasets: for UEC FOOD 256, HM increases to 24.4 % (+2.9 % vs. baseline; +3.5 % vs. GPT-3.5) with a notable Seen boost (18.4 % → 22.2 %, +3.8 %) while retaining the Unseen improvement (27.0 %, +1.2 % vs. baseline). For FOWA, GPT-4o lifts ZSD to 10.2 % (+2.2 %), Unseen to 8.8 % (+1.7 %), and HM to 16.0 %(+2.9 %). These results indicate a smooth scaling effect: stronger, richer descriptions alleviate the minor trade-off observed with GPT-3.5-Turbo and raise the overall performance ceiling without introducing instability.

Beyond comparing different LLM sources, we further validate the necessity of LLM-generated descriptions by comparing against CoOp, a representative prompt learning method. As shown in Table B.6, LLM-generated descriptions (29.1 % ZSD, 24.4 % GZSD HM) substantially outperform both hand-crafted prompts (24.2 % ZSD, 21.3 % HM) and CoOp-learned prompts (25.4 % ZSD, 20.8 % HM with 16 tokens). Notably, CoOp performs worse than hand-crafted prompts in GZSD (-0.5 %/-1.1 %) on UEC FOODD 256, suggesting overfitting to seen classes. This confirms that explicit culinary knowledge from LLMs is necessary for ZSFD, and cannot be effectively replaced by learned continuous prompts.

Appendix C. Additional quantitative analysis

C.1. Comparison with open-vocabulary detectors

To validate our framework's advantages over general-purpose vision-language detectors, we compared with Grounding DINO, a strong open-vocabulary detector. We evaluated Grounding DINO on FOWA under GZSD using zero-shot (pre-trained model with class names) and fine-tune (continued training on seen classes for varying epochs following official configuration) settings, using identical training data for fair comparison. As shown in Table C.7, zero-shot Grounding DINO achieved 5.6 % HM (4.6 % seen, 7.2 % unseen), reflecting domain gap with general datasets. Fine-tuning revealed catastrophic forgetting: at 5 epochs, seen mAP rose to 37.6 % but unseen dropped to 2.5 % (HM 4.69 %); at 15 epochs, seen peaked at 63.6 % but unseen collapsed to 1.3 % (HM 2.55 %), with no recovery at 25 epochs. This reveals that without explicit semantic guidance, fine-tuning biases toward seen-class patterns and forgets pre-trained associations. In contrast, our method achieved 16.0 % HM (89.8 % seen, 8.8 % unseen) after 12 epochs—6.3× higher than fine-tuned Grounding DINO and 6.8× higher unseen mAP, demonstrating effective seen-unseen balance. The performance gap stems from three designs: (1) *LLM-Generated Semantic Scaffolding*—rich GPT-4o descriptions serve as explicit anchors preventing overfitting, whereas Grounding DINO relies solely on class names; (2) *CT-FEM's Dual-Granularity Fusion*—decomposing descriptions into global attributes (P_G) and local identifiers (P_L) with fixed fusion ($\gamma = 0.6$) maintains balance, whereas Grounding DINO's single embedding becomes biased; (3) *MSCIM's Food-Centric Architecture*—multi-dilation pathways capture food's compositional structure optimally. This demonstrates that general-purpose open-vocabulary detectors face fundamental challenges in fine-grained ZSFD, which our framework overcomes through domain-optimized design and explicit semantic guidance.

Table C.7
Grounding DINO vs. our method on the FOWA dataset under the GZSD setting. Fine-tuning follows the official configuration (~2.5k iterations per epoch; text encoder frozen). Extended fine-tuning increases seen mAP but degrades unseen mAP, reducing HM. The ZSD column is omitted as unseen-only filtering yields an identical unseen mAP.

VLM Model	Traing Epochs	GZSD		
		Seen	Unseen	HM
Grounding DINO (Zero-Shot)	0	4.6	7.2	5.6
Grounding DINO (Fine-tune)	5	37.6	2.5	4.69
	10	39.5	2.3	4.35
	15	63.6	1.3	2.55
	25	62.1	1.3	2.55
Ours	**12**	**89.8**	**8.8**	**16.0**

Table C.8
Class-wise AP (%) on selected challenging categories of the FOWA dataset. The 15 classes include conceptually similar pairs such as diced chicken with green pepper vs. pork with green pepper, general vs. specific instances such as millet congee vs. congee, and different preparations of the same ingredient such as roast chicken wings vs. fried chicken wings. Seen and unseen classes are distinguished to assess generalization. The SA column reports the baseline performance, and the Ours column shows results from our method.

Category	Type	SA	**Ours**
diced chicken with green pepper	Seen	75.56	**77.11**
pork with green pepper	Seen	92.71	**94.25**
sour and spicy shredded potatoes	Seen	89.52	**90.58**
stir fried shredded potato	Seen	**62.55**	62.19
roast chicken wings	Seen	**98.76**	97.23
fried chicken wings	Unseen	6.58	**8.65**
congee	Unseen	7.02	**8.49**
millet congee	Unseen	78.35	**83.08**
seaweed and egg soup	Unseen	29.46	**31.12**
seaweed soup	Unseen	14.39	**15.98**
egg soup	Unseen	5.05	**6.42**
celery tofu	Unseen	9.61	**11.19**
green vegetables and mushrooms	Unseen	26.13	**27.78**
fried mushrooms	Unseen	10.28	**11.86**
apple	Unseen	6.25	**7.72**

C.2. Class-wise AP analysis on challenging categories

To better understand the model's behavior on challenging cases, we report class-wise mAP for 15 representative categories from the FOWA dataset (Table C.8). These categories were selected to highlight three types of difficulty: (1) pairs with high conceptual similarity (e.g., diced chicken with green pepper vs. pork with green pepper), (2) general vs. specific instances (e.g., millet congee vs. congee), and (3) different preparations of the same ingredient (e.g., roast vs. fried chicken wings). We distinguish between seen classes (present in training) and unseen

Table C.9
Complexity analysis comparing our method with baselines on the UEC FOOD 256. Inference metrics measured on a single NVIDIA GeForce RTX 2080 Ti GPU (batch size = 1).

Method Group	Method	#Params (M)	FLOPs (G)	FPS	VRAM (GB)	ZSD	GZSD		
							Seen	Unseen	HM
Generating-based	RRFS	60.2	538.8	14.4	3.5	23.6	20.1	22.9	21.4
	ZSFDet	60.5	539.4	14.7	3.5	27.3	21.9	26.1	23.8
Mapping-based	BLC	27.0	334.5	6.0	6.1	19.2	20.5	15.2	17.5
	SA	51.2	403.1	2.7	3.4	24.2	18.6	24.9	21.3
	Ours	**51.5**	**412.7**	**3.5**	**3.4**	**29.1**	**22.2**	**27.0**	**24.4**

classes (zero-shot) to evaluate generalization. The table compares our method (Ours) with the baseline SA, showing that while performance is strong on seen classes, unseen categories remain more challenging, reflecting the inherent difficulty of fine-grained ZSFD.

C.3. Computational cost analysis

This section provides a detailed breakdown of the computational cost analysis summarized in Section 4.6. Table C.9 presents a comprehensive comparison of parameters, FLOPs, and performance metrics across different methods. A direct comparison with the strongest mapping-based baseline, SA, reveals that the substantial accuracy gains of ZeSF are achieved with only a marginal increase in computational overhead. Specifically, ZeSF adds only 0.3M parameters and 9.6 GFLOPs but delivers a significant +3.1 % improvement in HM. This indicates the high efficiency of our proposed modules, a conclusion further supported by performance density metrics. As shown in Table C.9, ZeSF exhibits superior HM-per-FLOP (0.0591 vs. 0.0529) and HM-per-parameter (0.474 vs. 0.416) ratios compared to SA. Furthermore, the practical inference metrics highlight the efficiency of our architectural design. Despite the minor increase in theoretical FLOPs, ZeSF's inference speed is notably faster than SA's (3.5 FPS vs. 2.7 FPS), suggesting that the sparse, non-uniform structure of our MSCIM maps very effectively to modern GPU architectures. The peak VRAM usage remains identical at 3.4 GB, indicating no significant memory overhead.

This favorable efficiency profile also holds when comparing ZeSF to generative methods. As shown in Table C.9, our model uses fewer parameters (51.5M vs. 60M) and a significantly lower theoretical computational load (413 GFLOPs vs. 539 GFLOPs) than both RRFS and ZSFDet, while still achieving a higher HM. Although the generative methods exhibit a higher raw FPS, our framework delivers the best overall balance between top-tier performance (HM) and computational cost (Params/FLOPs). Moreover, this efficiency advantage extends to the training process. Generative approaches often rely on complex, multi-stage pipelines (e.g., pre-training on seen classes, training a feature generator, and then fine-tuning a classifier), which can be cumbersome. In contrast, our framework is trained end-to-end in a single, unified stage, significantly simplifying the development workflow. This finding reinforces our central argument that targeted architectural enhancements are a more efficient path to superior performance than computationally heavy feature synthesis.

Appendix D. Qualitative analysis

D.1. Failure case analysis

To better understand the limitations of our framework, we analyze representative failure cases on the FOWA and UEC FOOD 256 datasets (Fig. D.1). Two main error types are observed. The first is misclassification due to high visual similarity, which occurs when different classes share attributes such as color, shape, or texture. For example, a green apple may be misclassified as egg due to its uniform surface and reflection under specific lighting, fried potato slices may be confused with thousand-leaf tofu because of their similar shape, and a complex soup may be mistaken for green curry due to overlapping color cues from vegetables and broth. The second type is missed detections in cluttered or occluded scenes, where the model sometimes fails to detect small or partially hidden items, such as tofu in a dense dish. These cases highlight the persistent challenges of fine-grained recognition, where subtle attribute differences and scene complexity play a critical role. Addressing these issues will be an important direction for future work, such as incorporating lightweight attention mechanisms or leveraging knowledge distillation to improve robustness.

D.2. Grad-CAM visualizations

We use Grad-CAM to visualize the attention patterns of our model and the baseline, as shown in Fig. D.2. These visualizations provide qualitative evidence that our complete framework learns more discriminative attention patterns compared to the baseline, indicating the effectiveness of our architectural specialization when integrated with rich semantics.

Supplementary material

Supplementary material associated with this article can be found in the online version at 10.1016/j.patcog.2025.112928.

Fig. D.1. Representative failure cases on the FOWA (top row) and UEC FOOD 256 (bottom row) datasets. Two main error types are observed: (1) misclassification caused by high visual similarity in color, shape, or texture (e.g., fried potato slices vs. thousand-leaf tofu), and (2) missed detections in cluttered or occluded scenes (e.g., undetected tofu).

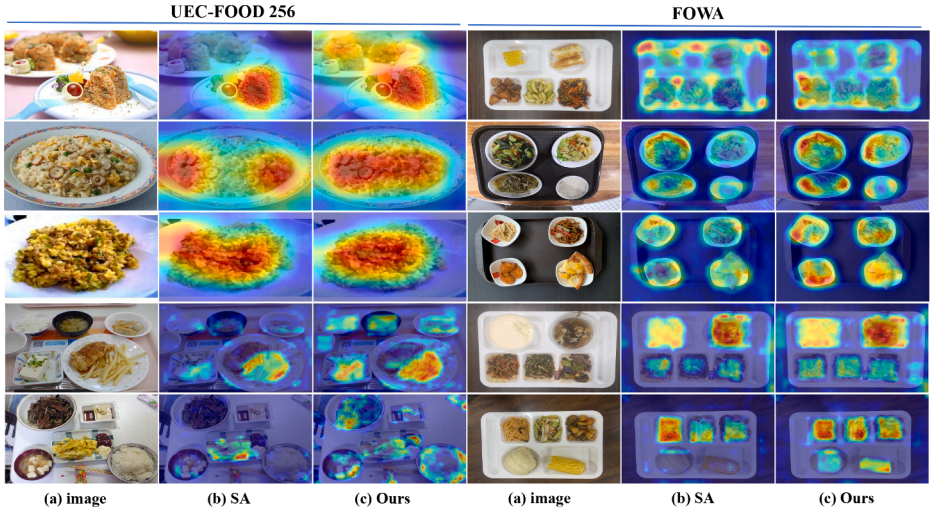

Fig. D.2. Grad-CAM visualizations of our model and the baseline on the UEC FOOD 256 and FOWA datasets. The first column shows the original images, the second column displays the results of the baseline model (SA), and the third column presents the results of our method.

References

[1] W. Min, S. Jiang, L. Liu, Y. Rui, R. Jain, A survey on food computing, ACM Comput. Surv. 52 (5) (2019) 1–36.

[2] W. Min, Z. Wang, Y. Liu, M. Luo, L. Kang, X. Wei, X. Wei, S. Jiang, Large scale visual food recognition, IEEE Trans. Pattern Anal. Mach. Intell. 45 (8) (2023) 9932–9949.

[3] X. Wang, Y. Bouzembrak, A.O. Lansink, H.J. Van Der Fels-Klerx, Application of machine learning to the monitoring and prediction of food safety: a review, Compr. Rev. Food Sci. Food Saf. 21 (1) (2022) 416–434.

[4] Y. Liu, W. Min, S. Jiang, Y. Rui, Convolution-enhanced Bi-Branch adaptive transformer with cross-task interaction for food category and ingredient recognition, IEEE Trans. Image Process. 33 (2024) 2572–2586.

[5] S. Forouzandeh, M. Rostami, K. Berahmand, R. Sheikhpour, Health-aware food recommendation system with dual attention in heterogeneous graphs, Comput. Biol. Med. 169 (2024) 107882.

[6] R. Zhang, J. Hu, Z.-Z. Li, C. Wang, C.-L. Liu, FGPR: a large-scale dataset and benchmark for fine-grained product retrieval, Pattern Recognit. 172 (2025) 112523.

[7] P. Zhou, W. Min, Y. Zhang, J. Song, Y. Jin, S. Jiang, SeeDS: semantic separable diffusion synthesizer for zero-shot food detection, in: Proc. ACM Int. Conf. Multimed., 2023, pp. 8157–8166.

[8] H. Liu, L. Zhang, J. Guan, S. Zhou, Zero-shot object detection by semantics-aware DETR with adaptive contrastive loss, in: Proc. ACM Int. Conf. Multimed., 2023, pp. 4421–4430.

[9] C. Yan, X. Chang, M. Luo, H. Liu, X. Zhang, Q. Zheng, Semantics-guided contrastive network for zero-shot object detection, IEEE Trans. Pattern Anal. Mach. Intell. 46 (3) (2022) 1530–1544.

[10] A. Radford, J.W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, et al., Learning transferable visual models from natural language supervision, in: Proc. Int. Conf. Mach. Learn., 2021, pp. 8748–8763.

[11] Y. Gao, W. Feng, R. Xiao, L. He, Z. He, J. Lv, C. Tang, Improving generalized zero-shot learning via cluster-based semantic disentangling representation, Pattern Recognit. 150 (2024) 110320.

[12] Y. Yi, G. Zeng, B. Ren, L.T. Yang, B. Chai, Y. Li, Prototype rectification for zero-shot learning, Pattern Recognit. 156 (2024) 110750.

[13] L. Zhou, Y. Liu, X. Bai, N. Li, X. Yu, J. Zhou, E.R. Hancock, Attribute subspaces for zero-shot learning, Pattern Recognit. 144 (2023) 109869.

[14] S. Chen, W. Hou, S. Khan, F. Khan, Progressive semantic-guided vision transformer for zero-shot learning, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2024, pp. 23964–23974.

[15] Z. Chen, Z. Zhao, J. Guo, J. Li, Z. Huang, SVIP: semantically contextualized visual patches for zero-shot learning, in: Proc. IEEE Int. Conf. Comput. Vis., 2025, pp. 3346–3356.

[16] H. Duan, Y. Long, S. Wang, H. Zhang, C.G. Willcocks, L. Shao, Dynamic unary convolution in transformers, IEEE Trans. Pattern Anal. Mach. Intell. 45 (11) (2023) 12747–12759.

[17] H. Duan, S. Shao, B. Zhai, T. Shah, J. Han, R. Ranjan, Parameter efficient fine-tuning for multi-modal generative vision models with Möbius-inspired transformation, Int. J. Comput. Vis. 133 (7) (2025) 4590–4603.

[18] J. Guo, Z. Rao, Z. Chen, J. Zhou, D. Tao, Fine-grained zero-shot learning: Advances, challenges, and prospects, arXiv preprint arXiv:2401.17766 (2024).

[19] B. Demirel, R.G. Cinbis, N. Ikizler-Cinbis, Zero-shot object detection by hybrid region embedding, in: Proc. Br. Mach. Vis. Conf., 2018, pp. 1–13.

[20] P. Huang, D. Zhang, D. Cheng, L. Han, P. Zhu, J. Han, M-RRFS: a memory-based robust region feature synthesizer for zero-shot object detection, Int. J. Comput. Vis. 132 (10) (2024) 4651–4672.

[21] S. Zhao, C. Gao, Y. Shao, L. Li, C. Yu, Z. Ji, N. Sang, GTNet: generative transfer network for zero-shot object detection, in: Proc. AAAI Conf. Artif. Intell., 2020, pp. 12967–12974.

[22] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, S. Zagoruyko, End-to-end object detection with transformers, in: Proc. Eur. Conf. Comput. Vis., 2020, pp. 213–229.

[23] P. Andreini, M. Tanfoni, S. Bonechi, M. Bianchini, Leveraging synthetic data for zero-shot and few-shot circle detection in real-world domains, Pattern Recognit.

172 (2025) 112407.
[24] P. Zhou, W. Min, J. Song, Y. Zhang, S. Jiang, Synthesizing knowledge-enhanced features for real-world zero-shot food detection, IEEE Trans. Image Process. 33 (2024) 1285–1298.
[25] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, J. Dai, Deformable DETR: deformable transformers for end-to-end object detection, in: Proc. Int. Conf. Learn. Represent., 2021, pp. 1–16.
[26] J. Jiao, Y.-M. Tang, K.-Y. Lin, Y. Gao, A.J. Ma, Y. Wang, W.-S. Zheng, DilateFormer: multi-scale dilated transformer for visual recognition, IEEE Trans. Multimed. 25 (2023) 8906–8919.
[27] Y. Kawano, K. Yanai, Automatic expansion of a food image dataset leveraging existing categories with domain adaptation, in: Proc. Eur. Conf. Comput. Vis., 2014, pp. 3–17.
[28] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft COCO: common objects in context, in: Proc. Eur. Conf. Comput. Vis., 2014, pp. 740–755.
[29] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, The PASCAL visual object classes (VOC) challenge, Int. J. Comput. Vis. 88 (2) (2010) 303–338.
[30] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 770–778.
[31] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: hierarchical vision transformer using shifted windows, in: Proc. IEEE Int. Conf. Comput. Vis., 2021, pp. 10012–10022.
[32] S. Rahman, S. Khan, N. Barnes, Polarity loss: improving visual-semantic alignment for zero-shot detection, IEEE Trans. Neural Netw. Learn. Syst. (2022) 4066–4078.
[33] Y. Zheng, R. Huang, C. Han, X. Huang, L. Cui, Background learnable cascade for zero-shot object detection, in: Proc. Asian Conf. Comput. Vis., 2020, pp. 107–123.
[34] N. Hayat, M. Hayat, S. Rahman, S. Khan, S.W. Zamir, F.S. Khan, Synthesizing the unseen for zero-shot object detection, in: Proc. Asian Conf. Comput. Vis., 2020, pp. 155–170.
[35] P. Huang, J. Han, D. Cheng, D. Zhang, Robust region feature synthesizer for zero-shot object detection, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2022, pp. 7622–7631.
[36] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G.S. Corrado, J. Dean, Zero-shot learning by convex combination of semantic embeddings, in: Proc. Int. Conf. Learn. Represent., 2014, pp. 1–9.
[37] H. Li, J. Mei, J. Zhou, Y. Hu, Zero-shot object detection based on dynamic semantic vectors, in: Proc. IEEE Int. Conf. Robot. Autom., 2023, pp. 9267–9273.
[38] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, Q. Jiang, C. Li, J. Yang, H. Su, J. Zhu, L. Zhang, Grounding DINO: marrying DINO with grounded pre-training for open-set object detection, in: Proc. Eur. Conf. Comput. Vis., 2024, pp. 38–55.
[39] M.L. van der, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (11) (2008) 2579–2605.
[40] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-CAM: visual explanations from deep networks via gradient-based localization, in: Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 618–626.