Synthesizing Knowledge-Enhanced Features for Real-World Zero-Shot Food Detection

Pengfei Zhou, Weiqing Min, *Senior Member, IEEE*, Jiajun Song, Yang Zhang, and Shuqiang Jiang, *Senior Member, IEEE*

***Abstract*— Food computing brings various perspectives to computer vision like vision-based food analysis for nutrition and health. As a fundamental task in food computing, food detection needs Zero-Shot Detection (ZSD) on novel unseen food objects to support real-world scenarios, such as intelligent kitchens and smart restaurants. Therefore, we first benchmark the task of Zero-Shot Food Detection (ZSFD) by introducing FOWA dataset with rich attribute annotations. Unlike ZSD, fine-grained problems in ZSFD like inter-class similarity make synthesized features inseparable. The complexity of food semantic attributes further makes it more difficult for current ZSD methods to distinguish various food categories. To address these problems, we propose a novel framework ZSFDet to tackle fine-grained problems by exploiting the interaction between complex attributes. Specifically, we model the correlation between food categories and attributes in ZSFDet by multi-source graphs to provide prior knowledge for distinguishing fine-grained features. Within ZSFDet, Knowledge-Enhanced Feature Synthesizer (KEFS) learns knowledge representation from multiple sources (e.g., ingredients correlation from knowledge graph) via the multi-source graph fusion. Conditioned on the fusion of semantic knowledge representation, the region feature diffusion model in KEFS can generate fine-grained features for training the effective zero-shot detector. Extensive evaluations demonstrate the superior performance of our method ZSFDet on FOWA and the widely-used food dataset UECFOOD-256, with significant improvements by 1.8% and 3.7% ZSD mAP compared with the strong baseline RRFS. Further experiments on PASCAL VOC and MS COCO prove that enhancement of the semantic knowledge can also improve the performance on general ZSD. Code and dataset are available at https://github.com/LanceZPF/KEFS.**

***Index Terms*— Food detection, zero-shot detection, food computing, object detection, zero-shot learning.**

Manuscript received 31 July 2023; revised 6 January 2024; accepted 24 January 2024. Date of publication 6 February 2024; date of current version 12 February 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 61972378, Grant 62125207, Grant U1936203, and Grant U19B2040; and in part by the Chinese Association for Artificial Intelligence (CAAI)-Huawei MindSpore Open Fund. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Raja Giryes. *(Corresponding author: Weiqing Min.)*

Pengfei Zhou, Jiajun Song, and Yang Zhang are with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China, and also with the College of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: pengfei.zhou@vipl.ict.ac.cn; jiajun.song@vipl.ict.ac.cn; yang.zhang@vipl.ict.ac.cn).

Weiqing Min and Shuqiang Jiang are with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China, also with the College of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 100049, China, and also with the Institute of Intelligent Computing Technology, Chinese Academy of Sciences, Suzhou 215124, China (e-mail: minweiqing@ict.ac.cn; sqjiang@ict.ac.cn).

Digital Object Identifier 10.1109/TIP.2024.3360899

I. Introduction

FOOD computing, an interdisciplinary research area that combines machine learning and food science to understand food-related data, has received considerable attention in computer vision [1], [2], [3]. Its importance is spotlighted by its ability to enable future nutrition and health management through the analysis of visual content from food images [4]. Academically, it enriches computer vision research by providing complex real-world scenarios and fine-grained issues for challenging advanced image processing algorithms [5], [6], [7]. Industrially, it revolutionizes the food industry with enhanced agricultural strategies, automatic food processing, and personalized dietary assessment [8], [9], [10]. Food detection serves as a fundamental technique in food computing, applying object detection paradigms to various real-world scenarios [11], such as automatic settlement [1] and dietary assessment [10], [12].

However, current food detection models face significant challenges in real-world scenes like intelligent kitchens [13] and smart restaurants [11], where novel food classes constantly emerge. The conventional food detection systems based on detectors trained on fixed classes have difficulty recognizing new, unseen food categories, leading to limited practical utility. To this end, we formally benchmark Zero-Shot Food Detection (ZSFD) based on Zero-Shot Detection (ZSD) to bridge this gap by enabling the detection of unseen food objects without requiring labeled new data, which are occasionally unavailable for real-world food applications.

To detect unseen objects "in the wild" scenarios, ZSD employs two major approaches: mapping-based methods [14], [15], [16] and generation-based methods [17], [18], [19]. However, since the mapping-based models are trained only with seen features in training dataset, they are easily biased towards seen classes in Generalized Zero-Shot Detection (GZSD), where images contain objects from both seen classes and unseen classes. Recently, generation-based methods have emerged with better ZSD performance by addressing the bias problem based on generative models. Generative models such as Variational Auto-Encoder (VAE) and Generative Adversarial Networks (GAN) are used to synthesize unseen visual features from semantic vectors and transfer the zero-shot problem into supervised learning on both seen and synthesized unseen features. However, the performance of those methods is limited in ZSFD systems due to fine-grained problems like inter-class similarity and intra-class variability.

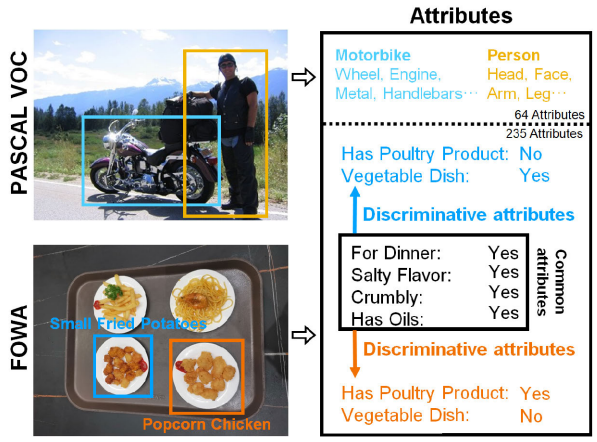


Fig. 1. Description of problems on ZSFD. Food objects have inter-class similarity and attribute complexity.

In the domain of ZSFD, generation-based methods require the synthesis of discriminative features. However, previous approaches meet difficulties in synthesizing distinguishable features for fine-grained food categories due to the problem of **inter-class similarity** and **attribute complexity**. As shown in Figure 1, compared with common objects that have distinct patterns and certain semantic parts, *Small Fried Potatoes* and *Popcorn Chicken* have similar visual features due to the inter-class similarity. Furthermore, common objects have simple attributes strongly associated with their category, while the complex attributes of food objects make food categories more likely to be confused. For example, the *Head* and *Face* attributes of the *People* can be strongly associated with a certain semantic part, which helps distinguish it from the *Motorbike*. In contrast, the same attributes of flavor and mouthfeel further confuse those two visually similar categories of *Small Fried Potatoes* and *Popcorn Chicken*. Complex attributes require rich semantic information to describe, and existing semantic representations from word embeddings are insufficient to capture discriminative information about food attributes.

Addressing these challenges necessitates a novel approach that leverages food domain knowledge, such as ingredient and hyperclass correlations. To this end, we introduce Zero-Shot Food Detector (ZSFDet), a new framework that utilizes domain knowledge from multiple sources, including ingredient correlation, hyperclass relationship and label co-occurrence probabilities modeled by knowledge, hyperclass and probability graphs, respectively. ZSFDet synthesizes discriminative and structured features for fine-grained food classes through:

1) Knowledge-Enhanced Feature Synthesizer (KEFS): KEFS adopts a multi-source graph fusion module with a knowledge encoder and attention modules to learn the knowledge representation from multi-source graph embeddings. The knowledge representation is fused with semantic content and then decoded into structure-aware synthesized features by a fusion decoder, which can exploit the interaction between semantic content from word vectors and attributes and knowledge representation.

2) Rich Semantic Representation: Only using word vectors cannot provide rich enough semantic information for fine-grained ZSFD. Unlike existing methods that rely on only word embeddings or semantic attributes, we use both food attributes and word embeddings for handling ZSFD. We explore different fusion strategies to explore the effective interaction of attribute embeddings and word embeddings to support further robust feature synthesizing.

3) Region Feature Diffusion Model (RFDM): To make sure the features we create are diversified and realistic, we use a novel 1-D diffusion model as the core generator in KEFS. Based on the probabilistic modeling of the denoising diffusion process, our proposed RFDM can enhance the training of zero-shot detectors and improve adaptability and accuracy in real-world scenarios via robust region feature synthesis.

To evaluate the effectiveness of ZSFD, we first adapt the existing food detection dataset UECFOOD-256 into a ZSFD dataset. However, most images in UECFOOD-256 contain only a single object, which is not appropriate to challenge detection algorithms. Moreover, UECFOOD-256 lacks multi-object real-world images with fine-grained attribute annotation. As attributes are required as side information for the fine-grained ZSFD task [20], we present a real-world dataset Food Objects With Attributes (FOWA), which contains 20,603 images collected in 10 real-world restaurant scenarios and 95,322 bounding box annotations. It is noted that FOWA has rich attributes annotated using the food knowledge graph (e.g., FoodKG [21]). The final established FOWA has 20,603 images and 228 classes with 235 attributes for each class. Compared with previous datasets, FOWA has a much richer bounding box and attribute annotation, which is essential for ZSFD.

To summarize, major contributions in this paper include the following three aspects:

- We propose a novel ZSFD framework ZSFDet that exploits the semantic interaction between attributes, word vectors and multi-source graphs, and learns effective zero-shot detectors via synthesizing knowledge-enhanced features that are inter-class separable.
- We benchmark the real-world ZSFD task and provide fine-grained attribute annotations in our established dataset FOWA. With challenging fine-grained categories and attribute annotations, our FOWA can also facilitate future research on ZSD.
- Extensive experiments show that our proposed ZSFDet achieves superior performance on the FOWA and UECFOOD-256. Moreover, ZSFDet also shows effectiveness on popular ZSD datasets PASCAL VOC and MS COCO, indicating good generalization ability.

II. Related Work

A. Food Detection

With the development of food computing [1], various research topics such as food recognition [5], [22], food detection [11] and multimodal food learning [6], [23] are bringing new challenges and perspectives to computer vision under real-world settings. As a fundamental technique in food computing, food detection locates and recognizes food objects, and further provides support for health-relevant applications [10]. Advances in general object detection promote

TABLE I
THE FOOD DETECTION RESULTS ON UNIMIB2016 AND OKTOBERFEST DATASETS (%)

Model	UNIMIB2016		Oktoberfest	
	mAP	AP50	mAP	AP50
EfficientDet-D3 [29]	83.6	90.4	66.3	91.3
Cascade R-CNN [30]	72.4	86.5	59.7	86.1
DETR [31]	72.2	84.2	58.1	84.0
Deformable DETR [32]	83.9	90.7	63.4	93.6
Food-specific DETR	**86.6**	**94.1**	**68.2**	**94.5**

early explorations in food detection [24]. Recently, researchers have been working on domain-adaptive methods to tackle particular issues in food images (e.g., fine-grained problems between food classes) and overcome limitations of data and annotations [25], [26]. Although promising results have been achieved via domain-specific food detection methods, it is still frustrating that high-performance food detection models can barely handle real-world tasks. For example, the Food-specific DETR in Table I is trained following the strategies in [5]. Though it has achieved nearly 95% AP50 on two food detection datasets, UNIMIB2016 [27] and Oktoberfest [28], the overall accuracy of it was still less than 20% when we deployed it in specific restaurant scenarios. A major reason is that meals of new classes update constantly in real-world scenarios, and food detection models trained with fixed seen classes can not deal with unseen food objects. To address this issue, we first build the Zero-Shot Food Detection (ZSFD) framework for detecting both seen and unseen food objects. Considering there are no available datasets for ZSFD with attribute annotations, we present the FOWA dataset with 235 attributes to evaluate ZSFD performance.

B. Zero-Shot Learning

Zero-Shot Learning (ZSL) has become an emerging research field that recognizes objects whose instances are not seen during training based on word vectors or semantic attributes as side information [33], [34]. Early ZSL exploration is dominated by mapping-based methods that embed features into the same space and search the nearest neighbor in the embedding space for input samples. Mapping-based ZSL methods can be broadly divided into three types according to the embedding space: mapping from visual space to semantic space [35], [36], mapping from semantic space to visual space [37], [38], or mapping the visual features and semantic vectors into a common latent space [39]. However, since models of mapping-based methods are only trained with seen features, these models can have a severe bias against seen classes in Generalized Zero-Shot Learning (GZSL), where both seen and unseen classes appear at test time. To address this, generation-based methods provide a new strategy that synthesizes features for unseen classes and transforms the ZSL problem into traditional supervised learning. Although the generation-based methods achieve better performance in realistic GZSL tasks, they still suffer from the hypersensitivity towards the correlation between side information and visual attributes [40]. In this paper, our framework is built based on generation-based methods since the real-world ZSFD scenarios require stronger GZSL ability. We extract fine-grained attributes from the food knowledge graph to tackle the hypersensitivity problem caused by the dependency on rich semantic information.

C. Zero-Shot Object Detection

Based on the theory of ZSL, Zero-Shot Detection (ZSD) detects objects whose instances are unseen during training [41], [42]. Earlier ZSD methods use mapping functions to align visual and semantic features [43], [44], [45], but they suffer from bias against seen classes in Generalized ZSD (GZSD), where both seen and unseen classes appear at test time. To address this, generation-based methods use generative models (e.g., Variational Autoencoder (VAE) and Generative Adversarial Networks (GAN)) to synthesize visual features for unseen classes and transform the ZSD problem into traditional supervised learning [17], [19], [46]. For example, Zhu et al. [17] propose an unseen feature generation framework based on VAE. Despite better GZSD performance, generation-based ZSD methods face the problem of synthesizing features that are less dispersed than real features, which limits further improvement. The latest models relieve this problem by designing loss functions that ensure the diversity of generated samples. For example, Hayat et al. [18] synthesize unseen features via GAN with the diversity regulation. Huang et al. [19] design a robust region feature synthesizer for generating diverse features. However, current ZSD methods often struggle to perform well in specific fine-grained scenarios, such as in food detection across broad categories [20]. The subtle differences and vast variety in food instances make it challenging to create accurate and diverse synthetic features for ZSFD. Therefore, current research aims to bridge this gap by modeling complex domain knowledge using fine-grained semantic or visual feature correlation. For example, ingredient correlation from food knowledge can be exploited by multi-source graph fusion for synthesizing discriminative features to distinguish food classes with complex attributes. It is worth noting that Li et al. [47] also use graph neural networks for modeling the correlation between object regions in images. Different from it, the graph fusion module in our proposed ZSFDet works as style supervision in feature generation that ensures the distribution of generated features is close to the real distribution via graph regulation.

III. METHOD

A. Problem Definition

We formally define Zero-Shot Food Detection (ZSFD). Assuming a training set $\mathcal{X}_s$ that totally includes M_s food images belonging to C_s seen classes is provided. $\mathcal{Y}_s = \{1, \ldots, C_s\}$ and $\mathcal{Y}_u = \{C_s + 1, \ldots, C\}$ are label sets of seen classes and unseen classes, where $\mathcal{Y}_s \cap \mathcal{Y}_u = \emptyset$, $C = C_s + C_u$ is the number of all classes, and C_u is the number of unseen classes. $\mathcal{Y} = \mathcal{Y}_s \cup \mathcal{Y}_u$ denotes the total label set. Let semantic vector set be $\mathcal{V} = \mathcal{V}_s \cup \mathcal{V}_u$, where $\mathcal{V}_s$ and $\mathcal{V}_u$ are semantic vector sets of seen and unseen classes. For each $y \in \mathcal{Y}$, the semantic vector $\boldsymbol{v}_y \in \mathcal{V}$ can be both attribute vector or word vector from language models (e.g., BERT [48]). During the inference,

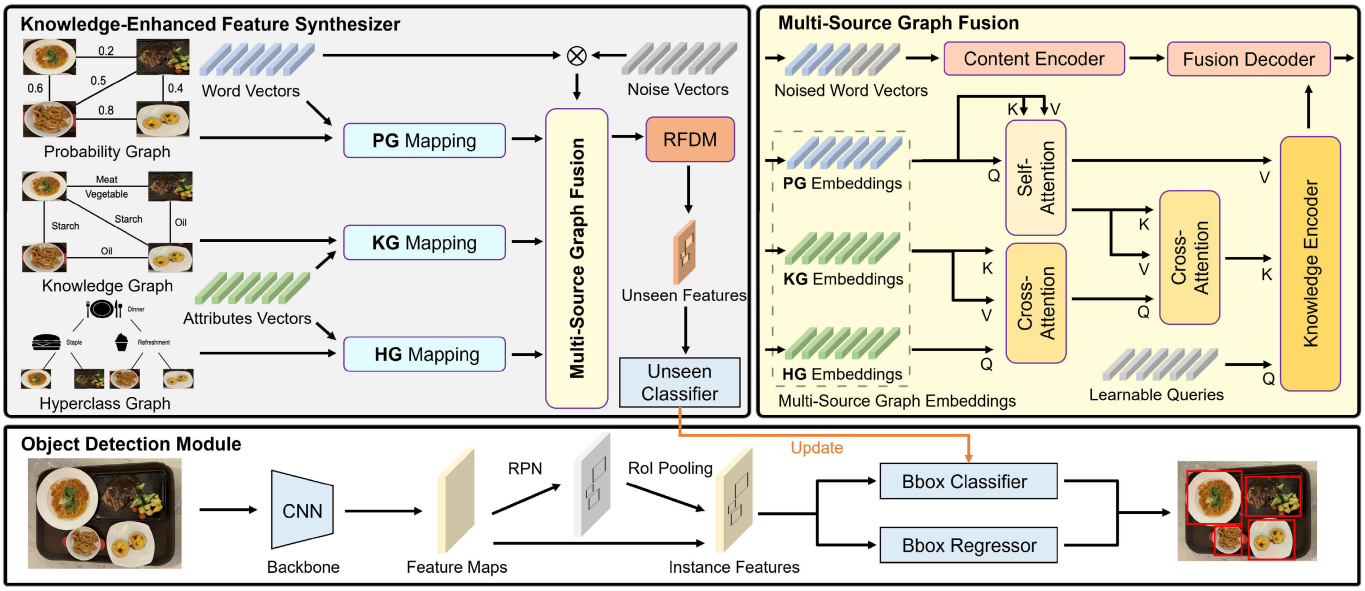


Fig. 2. Overview of our ZSFDet. The ZSFDet framework contains the Knowledge-Enhanced Feature Synthesizer (KEFS) and an object detection module. The object detector is first trained on the labeled seen data, and then the unseen classifier is trained on synthesized unseen features. ⊗ denotes the concatenation operation, "PG" denotes the probability graph, "KG" denotes the knowledge graph, "HG" denotes the hyperclass graph, and "RFDM" denotes the Region Feature Diffusion Model.

a test set $\mathcal{X}_t$ that contains both C_s seen classes and C_u unseen classes is given. The aim of ZSFD is to learn a detector on $\mathcal{X}_s$ with semantic vectors and detect unseen objects on $\mathcal{X}_t$. ZSFD also evaluates methods on a ZSD unseen set $\mathcal{X}_u \subset \mathcal{X}_t$ that only contains C_u unseen classes.

B. The Framework Overview

As illustrated in Figure 2, our framework ZSFDet contains the proposed Knowledge-Enhanced Feature Synthesizer (KEFS) and an object detection module. KEFS, denoted as $G(\cdot)$ is designed to understand and synthesize fine-grained food features through rich culinary knowledge, enabling our model to detect novel food objects that have no instances in training set. The detection module ω_d is a two-stage object detector with a feature extractor (backbone, Region Proposal Network, and RoI pooling layer) and a detection head that contains a bounding box regressor and a classifier.

We first train the detector on seen set $\mathcal{X}_s$ with corresponding images and ground-truth annotations. Then the trained detector is used to extract region features of instances from images in $\mathcal{X}_s$. Simultaneously, word vectors and attribute vectors from the Food Knowledge Graph (FKG) are obtained corresponding to seen classes $\mathcal{Y}_s$. For class set $\mathcal{Y}$, three graphs are separately defined, including a knowledge graph that models ingredient correlation, a probability graph that models label co-occurrence probability, and a hyperclass graph that models hyperclass relationship. Within KEFS, Multi-Source Graph Fusion (MSGF) learns the semantic knowledge representations from multi-source graph embeddings, and Region Feature Diffusion Model (RFDM) generates diverse unseen features based on the semantic knowledge representations. Applying the latest diffusion model on the fusion of knowledge representations and semantic content, KEFS can learn the robust mapping from semantic space to visual space, and synthesize discriminative features that are well-separated among fine-grained classes. An unseen classifier is further trained on synthesized unseen features. Finally, parameters in the detector are updated from parameters of unseen classifiers to gain the zero-shot detection ability. The details of MSGF and RFDM in KEFS will be discussed below.

C. Multi-Source Graph Fusion

In the Multi-Source Graph Fusion process, we start by converting food semantic vectors (word vectors and attribute vectors corresponding to food class) into a graph that represents how different food types relate to each other. In this graph, each point (node) is a food semantic vector connected by its correlation scores with other nodes (edges). A knowledge encoder with multiple attention modules is used to fuse multi-source graph embeddings into knowledge representation. In the other branch, word vectors concatenated with noise vectors (to ensure variations in the generated features) are sent to the content encoder to obtain the semantic content. Finally, the semantic content vectors and graph representation vectors are fused and decoded into visual features by the fusion decoder.

1) Multi-Source Graphs: We define $\mathcal{A} = \{\boldsymbol{A}^1, \boldsymbol{A}^2, \boldsymbol{A}^3\}$ as the set of adjacency matrices for multi-source graphs, where each adjacency matrix contains edges information in graphs that model the prior class correlations between nodes. We first take each food class and analyze its ingredients and characteristics, forming knowledge adjacency matrix $\boldsymbol{A}^1$ that models the attribute correlation in the knowledge graph. Next, we look at how these classes are grouped in broader categories to form a hyperclass adjacency matrix $\boldsymbol{A}^2$ that models hyperclass relationship and a probability adjacency matrix $\boldsymbol{A}^3$ that models label co-occurrence probability. Employing these correlations helps our model not only recognize foods it has seen before,

but also make guesses about unfamiliar foods by their 'culinary context' - much like humans make guesses about the taste of unfamiliar dishes by known similar dishes.

First, the attribute correlation from the knowledge graph is modeled by $\boldsymbol{A}^1$. On ZSFD, $\boldsymbol{A}^1$ represents the ingredient correlation of food classes in FKG. For example, if the i-th class *Steak* and the j-th class *Dried Beef* share r same ingredients that belong to the same group (e.g., *Beef Products*) in FKG, then entry $\boldsymbol{A}^1_{i,j} = r$. It means the value of edge that links the i-th and j-th nodes, which is correlation score, is assigned as r.

Second, the hyperclass relationship of classes is modeled by $\boldsymbol{A}^2$ of the hyperclass graph. Given class hierarchies in dataset, specific classes c_i can be inserted into a tree data structure as leaves. Assuming that the class tree has e levels, and the node at level 0 is the root, which is not considered as a cursor. The cursor of two leaf classes c_i and c_j is defined as the same ancestor node at the highest level. Therefore, the entry $\boldsymbol{A}^2_{i,j}$ in $\boldsymbol{A}^2$ is defined as:

$$\boldsymbol{A}^2_{i,j} = \begin{cases} l, & \text{if } c_i \text{ and } c_j \text{ have the cursor at the level } l \\ 0, & \text{if } c_i \text{ and } c_j \text{ do not have the cursor.} \end{cases} \tag{1}$$

Finally, we construct $\boldsymbol{A}^3$ to model conditional probability on training data and describe the statistical correlation of labels. Let $\boldsymbol{O}_{i,j}$ be the number of occurrence times of the pair of the i-th and the j-th class labels and $\boldsymbol{T}_i$ be the occurrence times of the i-th label, we can obtain $\boldsymbol{A}^3_{i,j} = \boldsymbol{O}_{i,j}/\boldsymbol{T}_i$. All $\boldsymbol{A}^1$, $\boldsymbol{A}^2$ and $\boldsymbol{A}^3$ are normalized and quantized into logical matrices by a threshold τ:

$$\boldsymbol{A}^k_{i,j} = \begin{cases} 1, & \text{if } \boldsymbol{A}^k_{i,j} \geq \tau \\ 0, & \text{if } \boldsymbol{A}^k_{i,j} < \tau, \end{cases} \tag{2}$$

where $\boldsymbol{A}^k$ is the k-th prior graph in set $\mathcal{A}$. We can further utilize various semantic knowledge in multiple graphs by multi-source graph mapping.

2) Multi-Source Graph Mapping: Multiple graph embeddings are obtained via the graph mapping $\psi^k(\cdot)$, which is implemented as graph convolution [49]. Let input semantic vectors $\boldsymbol{V} \in \mathbb{R}^{n\times d}$ be word vectors or attribute vectors, and the corresponding graph adjacency matrix be $\boldsymbol{A}^k \in \mathbb{R}^{n\times n}$, where n is the number of classes and d is the dimension of the semantic vector. Through two graph convolution layers, graph embeddings $\boldsymbol{E}^k \in \mathbb{R}^{n\times d}$ of the corresponding k-th prior graph in set $\mathcal{A}$ can be obtained:

$$\boldsymbol{E}^k = \psi^k(\boldsymbol{V}) = \hat{\boldsymbol{A}}^k(\rho(\hat{\boldsymbol{A}}^k \boldsymbol{V} \boldsymbol{W}^k_1))\boldsymbol{W}^k_2, \tag{3}$$

where $\rho(\cdot)$ denotes LeakyReLU [50], $\hat{\boldsymbol{A}}_k = \boldsymbol{D}_k^{-\frac{1}{2}} \boldsymbol{A}_k \boldsymbol{D}_k^{-\frac{1}{2}}$ is Laplacian normalized version of the correlation matrix, and $\boldsymbol{D}_k$ is the diagonal node degree matrix of $\boldsymbol{A}_k$. $\boldsymbol{W}^k_1 \in \mathbb{R}^{d\times d'}$ and $\boldsymbol{W}^k_2 \in \mathbb{R}^{d'\times d}$ are transformation matrices, and d' is the latent dimension.

3) Knowledge Encoder: The knowledge encoder $\varphi(\cdot)$ encodes the fused word and attribute graph embeddings $\boldsymbol{E}_f \in \mathbb{R}^{n\times z}$ and word graph embeddings $\boldsymbol{E}_{w2v} \in \mathbb{R}^{n\times z}$ from word vectors $\boldsymbol{V}_{w2v}$ into knowledge representation vectors $\boldsymbol{S} \in \mathbb{R}^{n\times d}$:

$$\boldsymbol{S} = \varphi(\boldsymbol{Q}, \boldsymbol{E}_f, \boldsymbol{E}_{w2v}) = \text{MHA}(\boldsymbol{Q}\boldsymbol{W}_Q, \boldsymbol{E}_f\boldsymbol{W}_K, \boldsymbol{E}_{w2v}\boldsymbol{W}_V), \tag{4}$$

where z is the dimension of embedded features and $\boldsymbol{Q} \in \mathbb{R}^{n\times z}$ is a set of learnable queries that are randomly initialized by Gaussian distribution $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. The key $\boldsymbol{E}_f$ is obtained by fusing the word and attribute graph embeddings via cross-attention modules, and the value $\boldsymbol{E}_{w2v}$ is obtained by encoding probability graph embeddings $\boldsymbol{E}^3_{w2v} \in \mathbb{R}^{n\times d}$ via self-attention modules. $\boldsymbol{W}_Q \in \mathbb{R}^{z\times d}$, $\boldsymbol{W}_K \in \mathbb{R}^{z\times d}$ and $\boldsymbol{W}_V \in \mathbb{R}^{z\times d}$ are corresponding weight matrices for query, key and value. MHA$(\cdot)$ denotes the multi-head attention mechanism.

4) Content Encoder and Fusion Decoder: Noise vectors $\boldsymbol{Z} \in \mathbb{R}^{n\times d}$ are sampled from the Gaussian distribution $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. As illustrated in Figure 3, the content encoder consists of two linear layers. It takes the concatenation of input word vectors $\boldsymbol{V}_{w2v}$ and noise vectors $\boldsymbol{Z}$ as input and output content vectors $\boldsymbol{N} \in \mathbb{R}^{n\times e}$, where e is the new embedding dimension:

$$\boldsymbol{N} = \text{FC2}(\text{FC1}(\boldsymbol{V}_{w2v} \otimes \boldsymbol{Z})), \tag{5}$$

where $\otimes$ denotes the concatenation operation, and FC1 and FC2 denote linear operations followed by LeakyReLU.

The fusion decoder leverages two Adaptive Instance Normalization (AdaIN) [51] blocks with two linear transformations to normalize the content with the distribution of knowledge representation:

$$\text{AdaIN}(\boldsymbol{N}, \boldsymbol{S}) = \sigma(\boldsymbol{S}) \left(\frac{\boldsymbol{N} - \mu(\boldsymbol{N})}{\sigma(\boldsymbol{N})} \right) + \mu(\boldsymbol{S}), \tag{6}$$

where $\mu(\cdot)$ and $\sigma(\cdot)$ are the mean and standard deviation. Finally, the last fully connected layer projects fusion embeddings into the synthesized instance feature $\boldsymbol{H} \in \mathbb{R}^{n\times a}$, where a is the dimension of visual instance features.

5) Graph Denoising Loss: To ensure that the learned knowledge representations are constrained to approach the distribution and regularized by prior graphs, we propose the graph denoising loss $\mathcal{L}_G$ based on the graphs defined in MSGF, which can ensure the diversity of synthesized features when conditioned on the knowledge representation $\boldsymbol{S}$:

$$\mathcal{L}_G = \mathbb{E}\left[-\frac{1}{C} \sum_{k=1}^{3} \sum_{i=1}^{C} y_i log(\hat{\boldsymbol{s}}_i) - \alpha \hat{\boldsymbol{s}}_i log(\phi(\boldsymbol{b}^k_i)) \right], \tag{7}$$

where y_i is the i-th class label, $\phi(\cdot)$ is the sigmoid function, $\hat{\boldsymbol{s}}_i = \phi(\boldsymbol{s}_i)$, $\boldsymbol{s}_i \in \mathbb{R}^n$ is the i-th row vector in matrix $\boldsymbol{S}$, $\boldsymbol{b}^k_i \in \mathbb{R}^n$

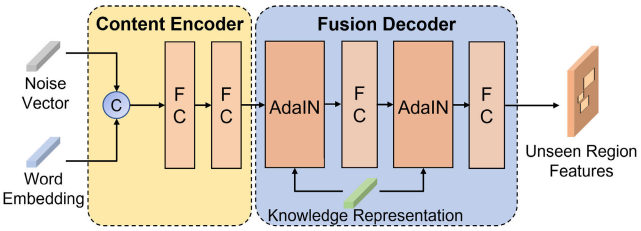


Fig. 3. Illustration of details in the content decoder and fusion decoder modules.

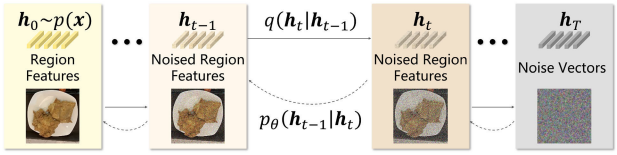


Fig. 4. The visual illustration of the diffusion process in RFDM. The forward process $q(\boldsymbol{h}_t|\boldsymbol{h}_{t-1})$ continually add Gaussian noise to $\boldsymbol{h}_{t-1}$ (from right to left), the reverse process $p_\theta(\boldsymbol{h}_{t-1}|\boldsymbol{h}_t)$ aims to denoise the noised feature vector $\boldsymbol{h}_t$.

Algorithm 1 Training Procedures of our ZSFDet

Input: $\mathcal{X}_s$, $\mathcal{X}_u$, $\mathcal{Y}_s$, $\mathcal{Y}_u$, $\mathcal{V}_{w2v}$, $\mathcal{V}_{att}$, $\mathcal{A}$
Output: Zero-shot detector with parameters ω_d
1: $\omega_d \leftarrow$ Train detector on $\mathcal{X}_s$ with annotations
2: $\mathcal{H}_s \leftarrow$ Extract region features from $\mathcal{X}_s$ via ω_d
3: $\boldsymbol{G} \leftarrow$ Initialize KEFS with $\mathcal{A}$
4: $\boldsymbol{G} \leftarrow$ Train $\boldsymbol{G}$ on $\mathcal{H}_s$, $\mathcal{V}_{w2v}$, $\mathcal{V}_{att}$, and $\mathcal{Y}_s$ by optimizing loss in Eq. 13
5: $\mathcal{H}_u \leftarrow$ Synthesize region features of unseen classes using the trained $\boldsymbol{G}$, $\mathcal{V}_{w2v}$, and $\mathcal{V}_{att}$
6: $\omega_{uc} \leftarrow$ Train unseen classifier ω_{uc} using $\mathcal{H}_u$ and $\mathcal{Y}_u$
7: $\omega_d \leftarrow$ Update parameters in ω_d with ω_{uc}
8: **return** ω_d

is the i-th row vector in matrix $\boldsymbol{A}^k\boldsymbol{S}$, and α is the trade-off factor that adjusts the contributions of two terms.

D. Region Feature Diffusion Model

Our ZSFDet uses a new generator named Regional Feature Diffusion Model (RFDM). The RFDM can generate diverse one-dimensional (1D) region feature vectors, closely mimicking variability in real-world culinary environments and enriching the diversity of synthesized food features. As Figure 4 illustrates, RFDM is based on the concept of modeling data distribution as a diffusion process, beginning from a basic prior distribution and progressively incorporating noise. The method learns to generate samples through a sequence of noise removal steps, reversing the diffusion process and retrieving the region feature.

Consider $\boldsymbol{h} \in \mathbb{R}^d$ as a 1D region feature vector. We propose $\boldsymbol{h}_T$ is generated through a diffusion process starting from the sample $\boldsymbol{h}_0 \sim p_0(\boldsymbol{h})$, where p_0 is the data distribution, and at each timestep $t = 1, \ldots, T$, Gaussian noise is added following a Markovian process. The noise level at each timestep is regulated by a scalar $\gamma_t \in (0, 1)$. The forward diffusion process $q(\boldsymbol{h}_t|\boldsymbol{h}_{t-1})$ for each timestep is described as:

$$\boldsymbol{h}_t = \sqrt{1-\gamma_t}\boldsymbol{h}_{t-1} + \sqrt{\gamma_t}\boldsymbol{z}_t, \tag{8}$$

where $\boldsymbol{z}_t \in \mathbb{R}^d$ is sampled from $\mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\boldsymbol{h}_0 = \boldsymbol{h}$. The RFDM seeks to reverse this process, represented by:

$$p_\theta(\boldsymbol{h}_{t-1}|\boldsymbol{h}_t) = \mathcal{N}(\boldsymbol{h}_{t-1}|\mu_\theta(\boldsymbol{h}_t, t), \Sigma_\theta(\boldsymbol{h}_t, t)\mathbf{I}), \tag{9}$$

where $\boldsymbol{h}_{t-1}$ and $\boldsymbol{h}_t$ represent the feature vector at timestep $t-1$ and t, respectively. $\mathcal{N}(\cdot)$ denotes Gaussian distribution, $\Sigma_\theta(\boldsymbol{h}_t, t)$ is a fixed covariance and $\mu_\theta(\boldsymbol{h}_t, t)$ is a predicted mean by RFDM. Specifically, RFDM uses the knowledge representations from MSGF as conditions to predict the noise for recovering the region feature based the predicted mean:

$$\boldsymbol{\mu}_\theta(\boldsymbol{h}_t, t) = \frac{1}{\sqrt{\beta_t}}(\boldsymbol{h}_t - \frac{1-\beta}{\sqrt{1-\bar{\beta}_t}}\boldsymbol{z}_\theta(\boldsymbol{h}_t, t)), \tag{10}$$

where $\beta_t = 1 - \gamma_t$, $\bar{\beta}_t = \prod_{i=1}^{t}\beta_i$ is the accumulated noise scalars, and $\boldsymbol{z}_\theta(\boldsymbol{h}_t, t)$ is the predicted noise parameterized by RFDM. Thus we can map $\boldsymbol{h}_t$ to $\boldsymbol{h}_0$ by applying a series of denoising functions $\boldsymbol{G}_t$:

$$\boldsymbol{h}_{t-1} = \boldsymbol{G}_t(\boldsymbol{h}_t, t, \boldsymbol{z}_\theta(\boldsymbol{h}_t, t); \theta), \tag{11}$$

where θ are the parameters of the MSGF module in RFDM. The denoising functions $\boldsymbol{G}_t$ are implemented by MSGF modules that share the same architecture but have different parameters for each timestep. The RFDM is trained by minimizing the mean squared error between the real noise $\boldsymbol{z}_t$ and the predicted noise $\boldsymbol{z}_\theta(\boldsymbol{h}_t, t)$ for all timesteps:

$$\begin{aligned}\mathcal{L}_R &= \mathbb{E}_{\boldsymbol{h},\boldsymbol{z}_t}[\sum_{t=1}^{T}||\boldsymbol{z}_t - \boldsymbol{z}_\theta(\boldsymbol{h}_t, t)||^2] \\ &= \mathbb{E}_{\boldsymbol{h},\boldsymbol{z}}[\sum_{t=1}^{T}||\boldsymbol{z}_t - \boldsymbol{G}_t(\boldsymbol{h}_t, t, \boldsymbol{z}_\theta(\boldsymbol{h}_t, t); \theta)||^2].\end{aligned} \tag{12}$$

E. Training Objective of KEFS

Given the seen instance feature collection $\mathcal{H}_s$ with word vector set $\mathcal{V}_{w2v}$ and attribute vector set $\mathcal{V}_{att}$ from $\mathcal{X}_s$, our goal is to learn a synthesizer $\boldsymbol{G}$: $(\mathcal{V}_{w2v} \times \mathcal{V}_{att} \times \mathcal{Z}) \mapsto \mathcal{H}$. The synthesizer $\boldsymbol{G}$ takes a word vector $\boldsymbol{v}_{vec} \in \mathcal{V}_{vec}$, an attribute vector $\boldsymbol{v}_{att} \in \mathcal{V}_{att}$ and a noise vector $\boldsymbol{z} \in \mathcal{Z}$ from the Gaussian distribution as input, and outputs the synthesized instance features $\tilde{\boldsymbol{h}} \in \mathcal{H}$. Specifically, the total training objective in KEFS comprises three parts: the conditional Wasserstein generative loss $\mathcal{L}_W$ [52] that is used to learn the knowledge representation as the condition of RFDM, the reconstruction loss $\mathcal{L}_R$ of RFDM and the Graph Denoising Loss $\mathcal{L}_G$:

$$\mathcal{L}_{Total} = \min_G \max_D \mathcal{L}_W + \lambda_1\mathcal{L}_R + \lambda_2\mathcal{L}_G, \tag{13}$$

where λ_1 and λ_2 are weights of losses. Training procedures of our method are summarized in Algorithm 1.

IV. The FOWA Dataset

In the FOWA dataset, images are collected in 10 restaurant scenarios and attributes are extracted from the food knowledge graph FoodKG [21]. Attributes from FoodKG include various ingredients and nutritional data of food entities. Since FoodKG contains massive food entities, it is difficult for humans to link classes in real food images to FoodKG. Thus, we first use the bi-directional matching [53] to efficiently match the closest food entities in FoodKG to classes in our dataset and form a similar list. We further use BERT to extract semantic vectors for each class and use the cosine distance to compute the similarity score between semantic vectors of entities from FoodKG and our dataset. Moreover, matched pairs with the highest similarity are filtered by a threshold $\epsilon = 0.7$. Finally, 228 classes are successfully linked to entities in FoodKG

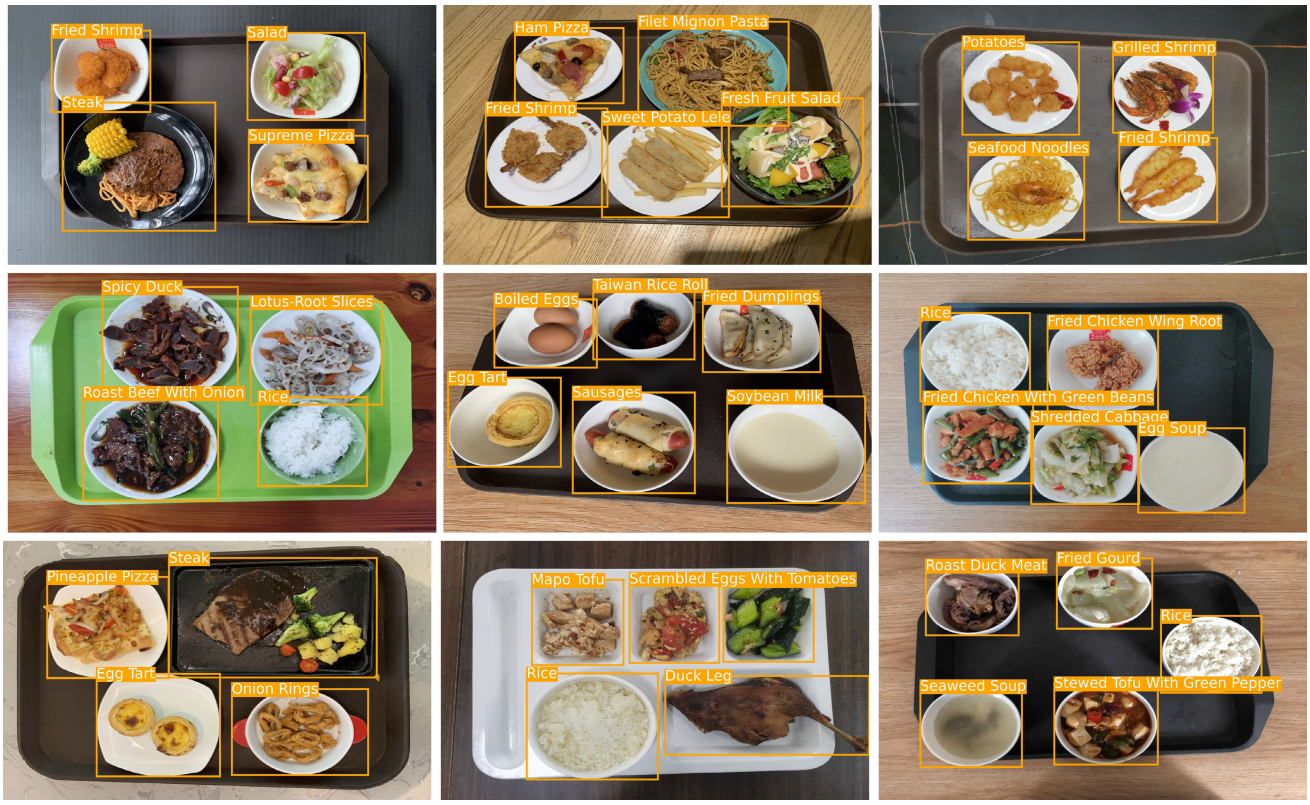


Fig. 5. Visualization examples of images and bounding boxes annotations in FOWA.

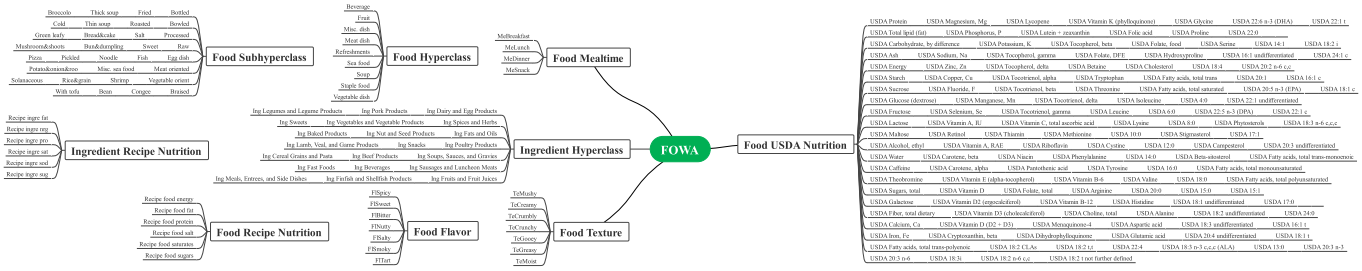


Fig. 6. Illustration of attributes in FOWA. Attributes in FOWA can be categorized into 9 hyperclasses.

and double-checked manually. As a result, the FOWA has 20,603 images and 228 classes, and each class is annotated with 235 attributes. FOWA also has three levels of classes, which include 9 hyperclasses (e.g., meat and vegetable), 21 subclasses (e.g., poultry and beef), and 228 specific classes. Since the Large Language Models (LLMs) were not yet available when the project was completed, most of the semantic attributes were manually annotated. Nowadays, leveraging advanced LLMs such as GPT-4 [54] and GPT-4V [55], we can automatically generate rich semantic attributes for hundreds of food classes directly. This technological advancement significantly reduces the laborious effort required in annotation, making the process of expanding our dataset more efficient to more comprehensively evaluate related methods.

As illustrated in Figure 5 and Figure 6, FOWA stands out with its rich attribute annotations covering a wide range from taste and texture to nutritional content, making it particularly suited for the complex tasks of Zero-Shot Food Detection (ZSFD). 235 attributes in FOWA can be categorized into 9 hyperclasses: *Food Mealtime*, *Food Flavor*, *Food Texture*, *Food Hyperclass*, *Food Subhyperclass*, *Ingredient Hyperclass*, *Food USDA Nutrition*, *Food Recipe Nutrition*, and *Ingredient Recipe Nutrition*. The dataset's long-tail distribution of bounding box annotations, as presented in Figure 7, presents a realistic and challenging scenario for models to tackle. We compare the proposed FOWA to existing food datasets like UECFOOD-256 in Table II, where we can see that FOWA provides a significantly larger set of bounding box annotations (95,322 compared to 28,429). Unlike common object detection datasets such as MS COCO, FOWA specializes in fine-grained attribute annotations crucial for zero-shot learning, offering a more challenging and relevant benchmark for ZSFD approaches. In terms of open-source, we are committed to making FOWA widely available to the research community.

V. Experiments

Dataset Splittings: Following the setting in [14] and [59], categories in FOWA are split into 184 seen classes and 44 unseen classes, and categories in UECFOOD-256 are split

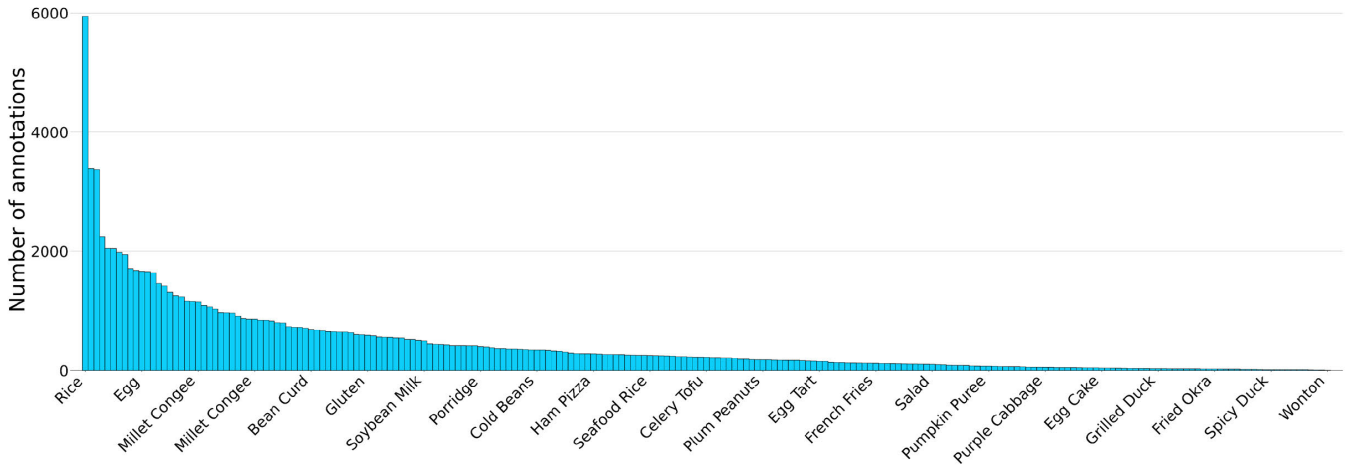


Fig. 7. Sorted numbers of bounding box annotations distribution in FOWA. Sampled class names are on the x-axis and the numbers of annotations are on the y-axis.

TABLE II
STATISTICS OF THE PROPOSED FOWA AND EXISTING DATASETS THAT ARE WIDELY USED FOR ZSD

Dataset	Att.	Classes			Images		
		S	U	Total	Training	Test	Total
PASCAL VOC [56]	64	16	4	20	10,728	10,834	21,562
MS COCO [57]	0	65	15	80	82,783	40,504	123,287
UECFOOD-256 [58]	0	205	51	256	20,452	5,732	26,184
FOWA	235	184	44	228	10,463	10,140	20,603

into 205 seen classes and 51 unseen classes, respectively. We also compare the ZSD performance of our proposed method ZSFDet with State-Of-The-Art (SOTA) ZSD methods on widely-used datasets: PASCAL VOC 2007+2012 [56] and MS COCO 2014 [57] using the same dataset splitting [18], [19], which is shown in Table II. Note that two different splits are adopted for MS COCO: 48/17 seen/unseen split and 65/15 seen/unseen split.

Implementation Details: We adopt the Faster-RCNN [61] with the ResNet-101 [62] as the backbone for fair comparisons. For training the synthesizer, Adam [63] is used with a learning rate of 1e-5. We use $\tau = 0.4$ as the threshold for logical adjacency matrices. Following the setting in [31], both the self-attention block and knowledge encoder in our approach include 6 layers [64]. The head number of multi-head attention is 4. We set $T = 100$ for the noise sampling process of RFDM. The linear start and the linear end for noise scalars are set to $\gamma_1 = 8.5\text{e-}4$ and $\gamma_T = 1.2\text{e-}2$. For each unseen class, we synthesize 500/500/500/250 features for FOWA/UECFOOD-256/PASCAL VOC/MS COCO to align with the setting in baseline methods [18], [19] for fair comparison. Since the ZSFDet is robust for hyper-parameters (the fluctuation range of mAP is less than 0.1% on three datasets under grid search), we set $\lambda_1 = 0.1$, $\lambda_2 = 0.1$ and $\alpha = 0.7$. Attributes of PASCAL VOC are provided by aPY [65]. Word embedding vectors of class names are extracted by BERT [48] for FOWA and UECFOOD-256, and by FastText [66] for PASCAL VOC and MS COCO.

Evaluation Metrics: Similar to previous works [14], [19], we use mean Average Precision (mAP) and Recall@100 with IoU threshold 0.5 for the evaluation of FOWA, UECFOOD-256 and PASCAL VOC. For MS COCO, we report mAP and Recall@100 with IoU thresholds of 0.4, 0.5, and 0.6, respectively. We also present per-class Average Precision (AP) to compare class-wise performance. The Harmonic Mean (HM) of seen and unseen is used for performance comparison under the setting of GZSD.

Comparison Methods: We compare our method with presentative classic ZSD methods and various state-of-the-art ZSD methods. These include: (1) **ConSE** [34] is a zero-shot learning method, and is reimplemented on the Faster R-CNN model for the ZSD task. Following the idea in [34], the training of the detector is on the object detection training set and inference with the semantic information; (2) **SB** [14] is a background-aware approach that considers external annotations from object instances belonging to neither seen nor unseen, which helps to address the confusion between unseen and background; (3) **DSES** [14] is a variant of SB that does not use background-aware representations but employs external data sources (e.g., large open vocabulary) for representation of background regions; (4) **SAN** [43] is a Faster-RCNN-based ZSD approach that jointly models the interplay between visual and semantic domain information (e.g. super-class information); (5) **HRE** [15] is a YOLO-based end-to-end ZSD approach that learns a direct mapping from region pixels to the space of class embeddings; (6) **PL** [59] is a RetinaNet-based ZSD approach that uses polarity loss for better alignment of visual features and semantic vectors; (7) **BLC** [45] integrates Cascade Semantic RCNN, semantic information flow and background learnable RPN into a unified ZSD framework; (8) **SU** [18] proposes a unified generation-based ZSD framework based on WGAN [52] and diversity loss; (9) **CZSD** [16] incorporates two semantics-guided contrastive learning networks to learn unseen visual representations by contrasting between region-category and region-region pairs; (10) **RRFS** [19] proposes an intra-class semantic diverging loss and inter-class structure preserving loss to synthesize robust region features for ZSD. (11) **TCB** [67] is a novel

TABLE III
COMPARISON WITH BASELINES ON FOWA (%)

Metric	Method	ZSD	GZSD		
			Seen	Unseen	HM
Recall@100	ConSE [34]	39.7	58.0	38.1	46.4
	PL [59]	40.1	53.9	39.6	45.7
	BLC [45]	41.2	55.3	40.5	46.8
	CZSD [16]	48.0	86.1	44.8	58.9
	SU [18]	45.3	82.3	44.1	57.4
	RRFS [19]	48.8	86.6	47.6	61.4
	SeeDS [60]	52.3	86.7	49.5	63.2
	ZSFDet	**53.5**	**87.0**	**50.1**	**63.6**
mAP	ConSE [34]	0.8	54.3	0.7	1.4
	PL [59]	1.0	50.8	0.7	1.4
	BLC [45]	1.1	51.1	0.9	1.8
	CZSD [16]	4.0	81.2	2.1	4.1
	SU [18]	3.9	79.1	2.3	4.5
	RRFS [19]	4.3	82.7	2.7	5.2
	SeeDS [60]	5.6	82.7	3.3	6.3
	ZSFDet	**6.1**	**82.8**	**3.6**	**6.9**

TABLE IV
COMPARISON ON UECFOOD-256 (%)

Metric	Method	ZSD	GZSD		
			Seen	Unseen	HM
Recall@100	ConSE [34]	54.4	50.1	38.2	43.3
	PL [59]	56.5	53.2	40.4	46.0
	BLC [45]	58.9	55.3	43.8	48.9
	CZSD [16]	60.7	**57.6**	45.5	50.8
	SU [18]	61.9	52.5	52.8	52.6
	RRFS [19]	64.8	54.9	55.1	55.0
	SeeDS [60]	74.0	55.2	61.4	58.1
	ZSFDet	**74.4**	57.0	**61.8**	**59.3**
mAP	ConSE [34]	11.3	19.7	9.0	12.4
	PL [59]	14.5	18.9	11.6	14.4
	BLC [45]	19.2	20.5	15.2	17.5
	CZSD [16]	22.0	**20.8**	16.2	18.2
	SU [18]	22.4	19.3	20.1	19.7
	RRFS [19]	23.6	20.1	22.9	21.4
	SeeDS [60]	27.1	20.2	26.0	22.7
	ZSFDet	**27.3**	21.9	**26.1**	**23.8**

two-way classification branch network for zero-shot detection that combines static and dynamic semantic vector branches. (12) **SeeDS** [60] proposes a Semantic Separable Diffusion Synthesizer to improve zero-shot food detection by enhancing region feature generation.

A. Comparisons With the State-of-the-Art

1) Evaluation on FOWA: Experimental results on FOWA are shown in Table III, where different ZSD methods are reimplemented for comparison. Compared with the baseline method RRFS of our framework, "ZSD", "Unseen" and "HM" are improved by 1.8%, 0.9% and 1.7% mAP, respectively. For Recall@100, "ZSD", "Unseen" and "HM" are improved by 4.7%, 2.5% and 2.2% mAP, respectively. Compared with the latest baseline SeeDS, the "HM" mAP of "GZSD" on our ZSFDet is higher by 0.6% on FOWA. The improvements demonstrate the proposed ZSFDet can evidently improve the ZSFD accuracy based on a robust feature synthesizer. Specifically, it utilizes the semantic knowledge from multi-source graph embeddings to help detectors achieve better ZSFD performance. We also observe that the mAP performance of all methods is low, which denotes that the larger number of classes makes ZSFD challenging, and there is more room for developing ZSFD methods. Note that the "Seen" performance has not been improved due to the same backbone detector for seen classes, which will be further explained in the ablation study section.

2) Evaluation on UECFOOD-256: We also reimplement baseline methods on UECFOOD-256 and show ZSFD results in Table IV. Compared with our baseline RRFS, "ZSD", "Unseen" and "HM" are improved by 3.7%, 3.2% and 2.4% mAP by our ZSFDet on UECFOOD-256, respectively. For Recall@100, "ZSD", "Unseen" and "HM" are improved by 9.6%, 6.7%, and 3.3%, respectively. Compared with the latest baseline SeeDS, the "HM" mAP of "GZSD" on our ZSFDet is higher by 1.1% on UECFOOD-256. These results further prove that the proposed ZSFDet framework with the MSGF and RFDM modules significantly improves the accuracy and recall ratio of ZSFD significantly, especially when applied to complex ZSFD scenarios in the setting of GZSD where seen and unseen food objects are presented simultaneously. We also observe that baseline methods cannot achieve as high an mAP as in ZSD when using the same data scale, demonstrating that the proposed ZSFD presents a challenging task with more room for developing future ZSFD methods.

TABLE V
COMPARISON OF MAP ON PASCAL VOC (%)

Model	ZSD	GZSD		
		Seen	Unseen	HM
ConSE [34]	52.1	59.3	22.3	32.4
SAN [43]	59.1	48.0	37.0	41.8
HRE [15]	54.2	62.4	25.5	36.2
PL [59]	62.1	-	-	-
BLC [45]	55.2	58.2	22.9	32.9
CZSD [16]	65.7	**63.2**	46.5	53.8
SU [18]	64.9	-	-	-
RRFS [19]	65.5	47.1	49.1	48.1
TCB [67]	59.3	61.0	29.8	40.0
ZSFDet	**69.2**	48.5	**50.8**	**49.6**

3) Evaluations on PASCAL VOC and MS COCO: Experimental results on PASCAL VOC are shown in Table V. Our ZSFDet outperforms all baselines under the ZSD setting, increasing the mAP from 65.5% to 69.2% compared with RRFS [19]. Furthermore, our method obtains better performance under a more challenging setting of GZSD. The "Seen", "Unseen" and "HM" are improved by 1.4%. 1.7% and 1.5% compared with the SOTA baseline RRFS. Results show that our method achieves a more balanced performance on the seen and unseen classes for GZSD. The class-wise AP performance on PASCAL VOC is reported in Table VI. We can observe that our approach achieves the best performance in 3 out of 4 classes.

We evaluate the ZSD performance on MS COCO with different IoU thresholds of 0.4, 0.5 and 0.6. As seen in Table VII, our method outperforms all baseline methods. For the "48/17" split, our method improves the mAP and

TABLE VI
COMPARISON OF CLASS-WISE AP AND MAP FOR DIFFERENT METHODS ON UNSEEN CLASSES OF PASCAL VOC (%)

Method	car	dog	sofa	train	mAP
SAN [43]	56.2	85.3	62.6	26.4	57.6
HRE [15]	55.0	82.0	55.0	26.0	54.5
PL [59]	63.7	87.2	53.2	44.1	62.1
BLC [45]	43.7	86.0	60.8	30.1	55.2
SU [18]	59.6	92.7	62.3	45.2	64.9
RRFS [19]	60.1	93.0	59.7	49.1	65.5
TCB [67]	62.9	72.9	**66.2**	35.2	59.3
ZSFDet	**63.7**	**94.3**	58.0	**60.8**	**69.2**

TABLE VII
ZSD PERFORMANCE ON MS COCO (%)

Model	Split	Recall@100			mAP
		IoU=0.4	IoU=0.5	IoU=0.6	IoU=0.5
SB [14]	48/17	34.5	22.1	11.3	0.3
DSES [14]	48/17	40.2	27.2	13.6	0.5
ConSE [34]	48/17	28.0	19.6	8.7	3.2
PL [59]	48/17	-	43.5	-	10.1
BLC [45]	48/17	51.3	48.8	45.0	10.6
CZSD [16]	48/17	56.1	52.4	47.2	12.5
RRFS [19]	48/17	58.1	53.5	47.9	13.4
TCB [67]	48/17	55.5	52.4	48.1	11.4
ZSFDet	48/17	**58.6**	**54.7**	**48.3**	**14.0**
ConSE [34]	65/15	30.4	23.5	10.1	3.9
PL [59]	65/15	-	37.7	-	12.4
BLC [45]	65/15	57.2	54.7	51.2	14.7
SU [18]	65/15	54.4	54.0	47.0	19.0
CZSD [16]	65/15	62.3	59.5	55.1	18.6
RRFS [19]	65/15	65.3	62.3	55.9	19.8
TCB [67]	65/15	62.5	59.9	55.1	13.8
ZSFDet	65/15	**66.5**	**64.2**	**56.7**	**20.3**

TABLE VIII
GZSD PERFORMANCE ON MS COCO (%)

Model	Split	Recall@100			mAP		
		S	U	HM	S	U	HM
ConSE [34]	48/17	43.8	12.3	19.2	37.2	1.2	2.3
PL [59]	48/17	38.2	26.3	31.2	35.9	4.1	7.4
BLC [45]	48/17	57.6	46.4	51.4	42.1	4.5	8.1
CZSD [16]	48/17	65.7	52.4	58.3	45.1	6.3	11.1
RRFS [19]	48/17	59.7	58.8	59.2	42.3	13.4	20.4
TCB [67]	48/17	**71.9**	52.4	**60.6**	**47.3**	4.9	8.8
ZSFDet	48/17	60.1	**60.7**	60.4	42.5	**14.3**	**21.4**
ConSE [34]	65/15	41.0	15.6	22.6	35.8	3.5	6.4
PL [59]	65/15	36.4	37.2	36.8	34.1	12.4	18.2
BLC [45]	65/15	56.4	51.7	53.9	36.0	13.1	19.2
SU [18]	65/15	57.7	53.9	55.7	36.9	19.0	25.1
CZSD [16]	65/15	62.9	58.6	60.7	**40.2**	16.5	23.4
RRFS [19]	65/15	58.6	61.8	60.2	37.4	19.8	26.0
TCB [67]	65/15	**69.3**	59.8	**64.2**	**39.9**	13.8	20.5
ZSFDet	65/15	59.3	**63.1**	61.1	37.5	**20.5**	**26.5**

TABLE IX
ABLATION STUDIES MEASURED BY MAP (%)

Dataset	Methods			ZSD	GZSD		
	Att.	MSGF	RFDM		S	U	HM
FOWA				4.3	82.7	2.7	5.2
	✓			4.5	82.8	2.8	5.4
	✓	✓		5.6	82.8	3.3	6.3
	✓	✓	✓	**6.1**	**82.8**	**3.6**	**6.9**
PASCAL VOC				65.8	48.4	49.2	48.4
	✓			66.1	48.5	49.7	49.1
	✓	✓		68.5	48.5	50.2	49.3
	✓	✓	✓	**69.2**	**48.5**	**50.8**	**49.6**

Recall@100 from 13.4% and 53.5% to 14.0% and 54.7% at IoU = 0.5 compared with RRFS. For the "65/15" split, our ZSFDet improves the mAP and Recall@100 from 19.8% and 62.3% to 20.3% and 64.2% at IoU = 0.5. As shown in Table VIII, our ZSFDet also outperforms the RRFS under the GZSD setting, where "S" denotes performance on seen classes and "U" denotes performance on unseen classes. The absolute "HM" performance gain of our method is 1.0% mAP and 1.2% Recall@100 for the "48/17" split, and 0.5% mAP and 0.9% Recall@100 for the "65/15" split. Compared with the lastest baseline TCB, the "HM" of "GZSD" mAP of our ZSFDet is higher by 9.6% on PASCAL VOC, by 12.6% on the "48/17" split of MS COCO, and by 5.0% on the "65/15" split of MS COCO, respectively. The experimental results on general ZSD datasets also demonstrate that our model exceeds existing methods in terms of both mAP and Recall@100.

B. Ablation Study

1) Contribution of Different Modules: We first conduct quantitative ablation analysis for the key modules, including the combination of attributes, MSGF, and RFDM. Table IX reports the "ZSD" and "GZSD" performance of mAP at IoU = 0.5 on FOWA and PASCAL VOC. As shown in Table IX, our ablation study reveals that incorporating both word vectors and attribute vectors ("Att.") enhances the ZSD by 0.2% on FOWA and 0.3% on PASCAL VOC, demonstrating the synthesizer's improved ability when leverages richer semantic information in zero-shot scenarios. We can observe that the "ZSD" performance has been improved by 1.3% on FOWA and 2.7% on PASCAL VOC, and the "U" performance has been improved by 0.6% on FOWA and 1.0% on PASCAL VOC compared with the baseline when MSGF is implemented. These improvements underscore the MSGF's effectiveness in enhancing feature synthesis and its adaptability across diverse dataset characteristics. Moreover, using RFDM to replace GAN as the core generator further improves mAP 0.5% of "ZSD" performance on FOWA and 0.7% on PASCAL VOC. It shows that a more powerful generator is crucial for the success of generative zero-shot framework. However, the "S" performance in GZSD has not been improved since classifier parameters for seen classes are mainly controlled by the backbone detector.

2) Contribution of Different Graphs and Fusion Strategies: Figure 8 compares the performance using three different graphs and fusion strategies in MSGF. We conduct this ablation study based on the GAN as the core generator to eliminate the effects of RFDM. The "HM" is improved to 6.2% mAP by the Knowledge Graph (KG) on FOWA, which is a notable increase compared to Hyperclass Graph (HG) and Probability Graph (PG) strategies. The gained performance by KG suggests its structured representation of domain knowledge

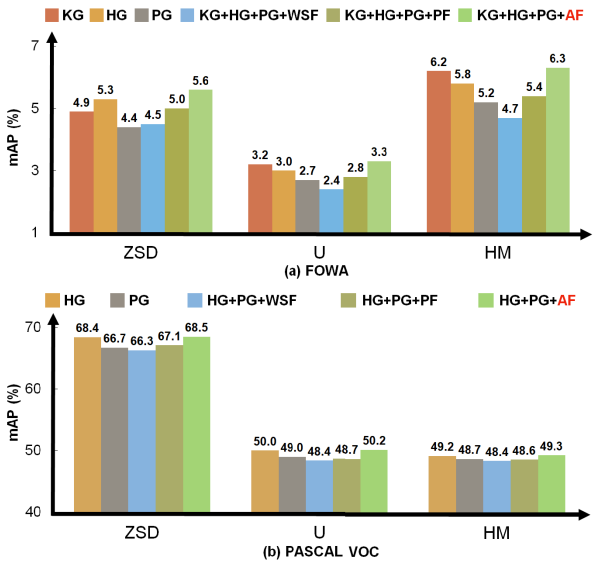


Fig. 8. Ablation studies on multi-source graphs and different fusion strategies on FOWA and PASCAL VOC (%).

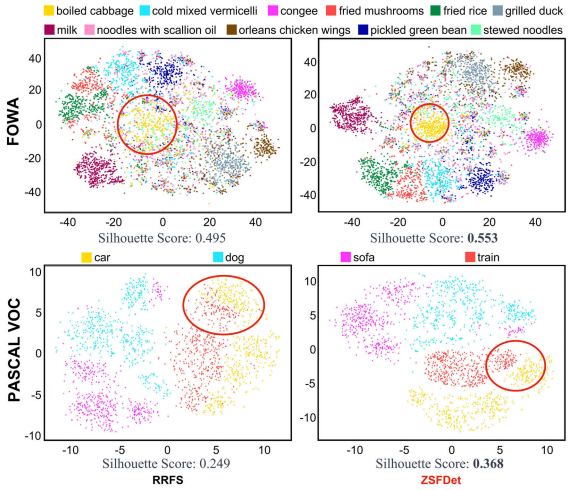


Fig. 9. The t-SNE visualization of synthesized features.

benefits the inter-class relationships and attribute correlations modeling that are crucial for ZSFD. The knowledge graph in PASCAL VOC is not designed due to the inaccessibility of the related knowledge graph data. Moreover, various fusion strategies are compared in Figure 8. The Weighted Sum Fusion (WSF) shows lesser performance improvements, as indicated by the 4.7% mAP on FOWA and 48.4% mAP on PASCAL VOC. In contrast, the Attention-based Fusion (AF) strategy significantly outperforms WSF and Product Fusion (PF) strategies, achieving 6.3% mAP on FOWA and 54.0% mAP on PASCAL VOC. This performance gain can be attributed to AF's ability to dynamically weigh different semantic sources, thereby providing a more adaptive and discriminative feature synthesis mechanism.

C. Qualitative Results

1) Feature Distribution Visualization: To further demonstrate the effectiveness of our model in optimizing the distribution structure of visual features, we utilize t-SNE [68] to visualize synthesized unseen features on FOWA and PASCAL VOC in Figure 9, where a quarter of the categories on FOWA are randomly selected to make visualization more clear and intuitive. The synthesized features from our model, as compared to the baseline RRFS, show well-defined clusters, as indicated by the silhouette scores: 0.553 versus 0.495 on FOWA, and 0.368 versus 0.249 on PASCAL VOC, respectively. The higher scores, especially for the FOWA dataset, suggest that our approach has resulted in more distinctive features that form well-separated clusters for different classes, facilitating the learning of a robust unseen classifier for ZSD.

2) Detection Results: We visualize ZSFD results on FOWA and UECFOOD-256 in Figure 10. For each dataset, the first row is detection results by baseline RRFS, and the second row is detection results by our method. As seen in Figure 10, the baseline method RRFS fails to detect several unseen food objects precisely, while our model provides more robust detection results. Due to the fine-grained problems of food, it is challenging to recognize various food categories, especially for visually similar food objects. The introduction of the knowledge graph in ZSFDet can utilize the correlation of ingredients for training an unseen classifier on inter-class separable synthesized features. For example, in the first column on FOWA, ZSFDet can discriminate the *Millet Congee* and the *Seaweed And Egg Soup* that both belong to *Soup* hyperclass via the difference of ingredients. Figure 10 also illustrates that the baseline method is confronted with the mistaken localization (e.g., the *Sofa* in the third sub-figure of the third row), the mistaken classification (the *Dog* in the second sub-figure of the fifth row), and the problem of low recall (the undetected *Umbrella* in the forth sub-figure of the fifth row). Compared with these, our approach also maintains accuracy in detecting general objects.

We provide more qualitative visualization results for failure case analysis in Figure 11. On FOWA, ZSFDet incorrectly identifies *Steamed Bun With Vegetable Stuffing* as *Steamed Bread with Brown Sugar*. This error can arise from the limited visual cues–the stuffing that differentiates the two is barely visible–and their similar semantic attributes. ZSFDet also tends to create multiple bounding boxes for the same object. Addressing this could involve tuning the Non-Maximum Suppression (NMS) parameters or integrating an end-to-end architecture with Hungarian loss. Interestingly, on the UECFOOD-256 dataset, we observed that similar-looking, non-target background objects can disrupt the detection of actual food objects. For example, too much deviation from the ground truth can be observed from the second case on UECFOOD-256 compared with the left correct detection on *Tensin Noodle*. This effect is likely due to the interference from background noise or mislabeling in the scene, which makes fine regression of the target box more difficult, resulting in a lower Intersection over Union (IoU) and greater positional error.

D. Discussion

To further provide a clear understanding of the framework's efficiency in real-world scenarios, we conduct model

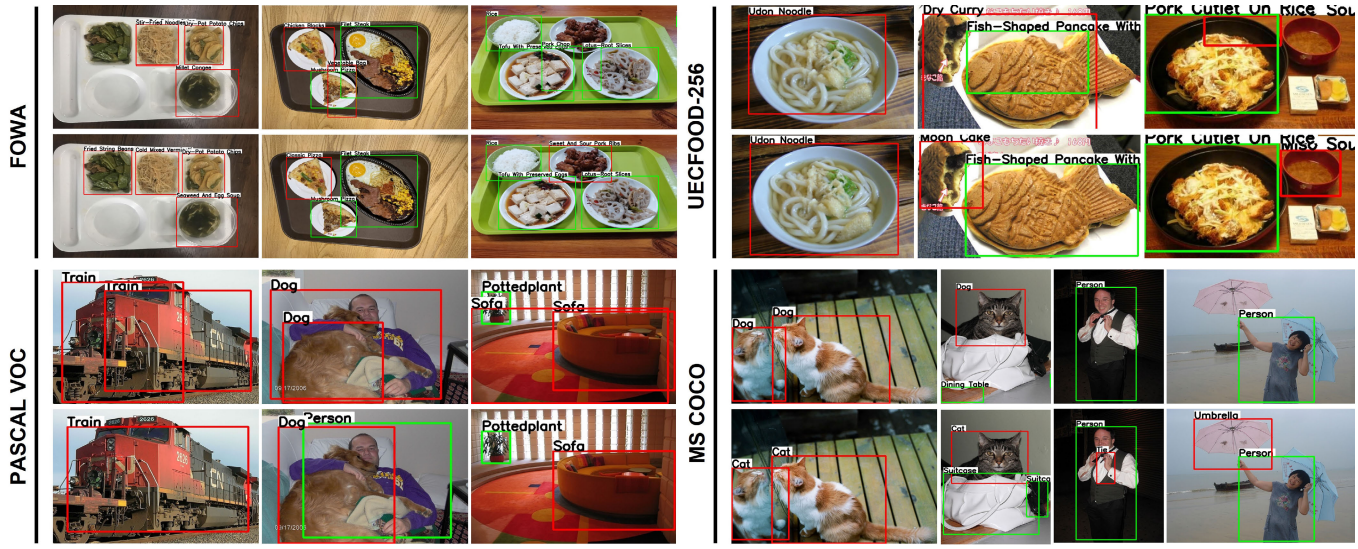


Fig. 10. Detection results by baseline RRFS (first row) and our approach (second row) on FOWA, UECFOOD-256, PASCAL VOC and MS COCO. Seen classes are shown with green and unseen with red. Zoom in for a better experience.

Fig. 11. More ZSFD visualization results for failure case studies on the FOWA and UEC-FOOD256.

complexity analysis in TABLE X following the rules as defined by [69]. The TABLE X compares the complexity of three methods RRF5, SeeDS, and ZSFDet on the FOWA dataset. With respect to the space complexity, the backbone detector contributes most parameters. ZSFDet has a slightly higher computational cost with 658MB and 145.38 GFLOPs, but it surpasses their performance across all metrics. The increased model size and FLOPs ensure a more robust generator and do not affect the inference efficiency. With respect to the time complexity, the training and test speed are barely affected as parallel computing benefits from the added modules. Compared with baselines, despite the close computational costs between SeeDS and ZSFDet, the latter's enhanced ability to detect unseen objects offers significant advantages in practical applications.

Our method achieves state-of-the-art performance on FOWA, UECFOOD-256, and two ZSD datasets by leveraging synthesized robust features from word vectors and attribute vectors. The KEFS in ZSFDet, through MSGF and RFDM, aligns the distribution of synthesized features with real visual features based on the learned knowledge from multi-source

TABLE X
MODEL COMPLEXITY ANALYSIS OF VARIOUS METHODS WITH THEIR MAP (%) ON FOWA

	Parameter	GFLOPs	FPS	ZSD	S	U	HM
RRFS	522MB	117.10	15.2	4.3	82.7	2.7	5.2
SeeDS	652MB	144.81	15.1	4.3	82.7	2.7	5.2
ZSFDet	658MB	145.38	15.1	6.1	82.8	3.6	6.9

graphs and achieves more accurate ZSD detectors on fine-grained classes. While ZSFDet excels in ZSFD, it also generalizes effectively to general ZSD, thanks to probability and hyperclass graphs encoded by multi-source graph mapping and self-attention mechanisms in MSGF. However, the computational increase may be challenging in resource-constrained environments, and attribute reliance might not be viable where data is limited or privacy-sensitive. Performance gains are less pronounced on common datasets like PASCAL VOC and MS COCO, possibly due to the lack of ingredient correlation in these datasets. Despite these challenges, the cross-attention fusion in our knowledge graph induces notable improvements. Future research should aim to mitigate these limitations, exploring how these methods can be simplified for different scenarios without performance loss.

VI. CONCLUSION

This paper benchmarks the Zero-Shot Food Detection (ZSFD) task with fine-grained food attribute annotations in FOWA, which are extracted from the food knowledge graph. We also propose a novel method ZSFDet based on Knowledge-Enhanced Feature Synthesizer (KEFS), which includes Multi-Source Graph Fusion (MSGF) and Region Feature Diffusion Model (RFDM). MSGF enhances feature synthesis based on the rich semantic information from multi-source graph embeddings. RFDM ensures synthesized unseen instance features are diversified and robust for learning the efficient zero-shot detector. Without bells and whistles,

the proposed ZSFDet achieves state-of-the-art performance on two ZSFD datasets FOWA and UECFOOD-256, and also outperforms strong baselines on ZSD datasets PASCAL VOC and MS COCO. Despite its effectiveness, the current computational and data demands highlight areas for improvement. Future directions include optimizing these aspects and integrating the latest large language models, potentially enhancing semantic analysis within an end-to-end Transformer-based detector framework for more reliable performance. Based on powerful vision-language pre-trained models, we envision open-vocabulary food detection becoming increasingly vital in addressing real-world challenges, from dietary monitoring to culinary exploration.

REFERENCES

[1] W. Min, S. Jiang, L. Liu, Y. Rui, and R. Jain, "A survey on food computing," *ACM Comput. Surv.*, vol. 52, no. 5, pp. 1–36, Sep. 2020.

[2] G. Vaidyanathan, "What humanity should eat to stay healthy and save the planet," *Nature*, vol. 600, no. 7887, pp. 22–25, Dec. 2021.

[3] H. Wang, G. Lin, S. C. H. Hoi, and C. Miao, "Learning structural representations for recipe generation and food retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3363–3377, Mar. 2023.

[4] W. Willett et al., "Food in the Anthropocene: The EAT–lancet commission on healthy diets from sustainable food systems," *Lancet*, vol. 393, no. 10170, pp. 447–492, 2019.

[5] W. Min et al., "Large scale visual food recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 8, pp. 9932–9949, Aug. 2023.

[6] J. Marín et al., "Recipe1M+: A dataset for learning cross-modal embeddings for cooking recipes and food images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 187–203, Jan. 2021.

[7] S. Jiang, W. Min, L. Liu, and Z. Luo, "Multi-scale multi-view deep feature aggregation for food recognition," *IEEE Trans. Image Process.*, vol. 29, pp. 265–276, 2020.

[8] B. Basso and J. Antle, "Digital agriculture to design sustainable agricultural systems," *Nature Sustainability*, vol. 3, no. 4, pp. 254–256, Apr. 2020.

[9] M. I. H. Khan, S. S. Sablani, R. Nayak, and Y. Gu, "Machine learning-based modeling in food processing applications: State of the art," *Comprehensive Rev. Food Sci. Food Saf.*, vol. 21, no. 2, pp. 1409–1438, 2022.

[10] W. Wang, W. Min, T. Li, X. Dong, H. Li, and S. Jiang, "A review on vision-based analysis for automatic dietary assessment," *Trends Food Sci. Technol.*, vol. 122, pp. 223–237, Apr. 2022.

[11] E. Aguilar, B. Remeseiro, M. Bolaños, and P. Radeva, "Grab, pay, and eat: Semantic food detection for smart restaurants," *IEEE Trans. Multimedia*, vol. 20, no. 12, pp. 3266–3275, Dec. 2018.

[12] A. Myers et al., "Im2Calories: Towards an automated mobile vision food diary," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1233–1241.

[13] D. Damen et al., "The EPIC-KITCHENS dataset: Collection, challenges and baselines," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 11, pp. 4125–4141, Nov. 2021.

[14] A. Bansal, K. Sikka, G. Sharma, R. Chellappa, and A. Divakaran, "Zero-shot object detection," in *Proc. ECCV*, 2018, pp. 384–400.

[15] D. Berkan, R. G. Cinbiş, and N. İkizler Cinbiş, "Zero-shot object detection by hybrid region embedding," in *Proc. BMVC*, 2018, pp. 56–68.

[16] C. Yan, X. Chang, M. Luo, H. Liu, X. Zhang, and Q. Zheng, "Semantics-guided contrastive network for zero-shot object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, pp. 1–15, Jan. 2022.

[17] P. Zhu, H. Wang, and V. Saligrama, "Don't even look once: Synthesizing features for zero-shot detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11693–11702.

[18] N. Hayat, M. Hayat, S. Rahman, S. Khan, S. W. Zamir, and F. S. Khan, "Synthesizing the unseen for zero-shot object detection," in *Proc. ACCV*, 2020, pp. 155–170.

[19] P. Huang, J. Han, D. Cheng, and D. Zhang, "Robust region feature synthesizer for zero-shot object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 7622–7631.

[20] S. Badirli, Z. Akata, G. Mohler, C. Picard, and M. M. Dundar, "Fine-grained zero-shot learning with DNA as side information," in *Proc. NeurIPS*, 2021, pp. 19352–19362.

[21] S. Haussmann et al., "FoodKG: A semantics-driven knowledge graph for food recommendation," in *Proc. ISWC*, 2019, pp. 146–162.

[22] L. Bossard, M. Guillaumin, and L. V. Gool, "Food-101–mining discriminative components with random forests," in *Proc. ECCV*, 2014, pp. 446–461.

[23] H. Wang, D. Sahoo, C. Liu, E. Lim, and S. C. Hoi, "Learning cross-modal embeddings with adversarial networks for cooking recipes and food images," in *Proc. CVPR*, 2019, pp. 11572–11581.

[24] L. Rachakonda, S. P. Mohanty, and E. Kougianos, "ILog: An intelligent device for automatic food intake monitoring and stress detection in the IoMT," *IEEE Trans. Consum. Electron.*, vol. 66, no. 2, pp. 115–124, May 2020.

[25] T. Ege and K. Yanai, "Simultaneous estimation of food categories and calories with multi-task CNN," in *Proc. 15th IAPR Int. Conf. Mach. Vis. Appl. (MVA)*, May 2017, pp. 198–201.

[26] W. Shimoda and K. Yanai, "Webly-supervised food detection with foodness proposal," *IEICE Trans. Inf. Syst.*, vol. 102, no. 7, pp. 1230–1239, 2019.

[27] G. Ciocca, P. Napoletano, and R. Schettini, "Food recognition: A new dataset, experiments, and results," *IEEE J. Biomed. Health Informat.*, vol. 21, no. 3, pp. 588–598, May 2017.

[28] A. Ziller, J. Hansjakob, V. Rusinov, D. Zügner, P. Vogel, and S. Günnemann, "Oktoberfest food dataset," 2019, *arXiv:1912.05007*.

[29] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10778–10787.

[30] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6154–6162.

[31] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. ECCV*, Aug. 2020, pp. 213–229.

[32] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," in *Proc. ICLR*, 2021.

[33] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 453–465, Mar. 2014.

[34] M. Norouzi et al., "Zero-shot learning by convex combination of semantic embeddings," in *Proc. ICLR*, 2014.

[35] R. Socher, M. Ganjoo, C. D. Manning, and A. Ng, "Zero-shot learning through cross-modal transfer," in *Proc. NeurIPS*, vol. 26, 2013.

[36] L. Ba, K. Swersky, and S. Fidler, "Predicting deep zero-shot convolutional neural networks using textual descriptions," in *Proc. Int. Conf. Comput. Vis.*, Dec. 2015, pp. 4247–4255.

[37] L. Zhang, T. Xiang, and S. Gong, "Learning a deep embedding model for zero-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2021–2030.

[38] E. Kodirov, T. Xiang, and S. Gong, "Semantic autoencoder for zero-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3174–3183.

[39] Z. Akata, M. Malinowski, M. Fritz, and B. Schiele, "Multi-cue zero-shot learning with strong supervision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 59–68.

[40] E. Schonfeld, S. Ebrahimi, S. Sinha, T. Darrell, and Z. Akata, "Generalized zero-and few-shot learning via aligned variational autoencoders," in *Proc. CVPR*, 2019, pp. 8247–8255.

[41] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-shot learning—A comprehensive evaluation of the good, the bad and the ugly," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 9, pp. 2251–2265, Sep. 2019.

[42] X. Dai et al., "Synthetic feature assessment for zero-shot object detection," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2023, pp. 444–449.

[43] S. Rahman, S. Khan, and F. Porikli, "Zero-shot object detection: Learning to simultaneously recognize and localize novel concepts," in *Proc. ACCV*, 2018, pp. 547–563.

[44] Z. Li, L. Yao, X. Zhang, X. Wang, S. Kanhere, and H. Zhang, "Zero-shot object detection with textual descriptions," in *Proc. AAAI*, 2019, pp. 8690–8697.

[45] Y. Zheng, R. Huang, C. Han, X. Huang, and L. Cui, "Background learnable cascade for zero-shot object detection," in *Proc. ACCV*, 2020, pp. 107–123.

[46] S. Zhao et al., "GTNet: Generative transfer network for zero-shot object detection," in *Proc. AAAI*, 2020, pp. 12967–12974.

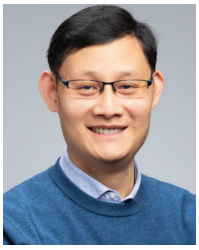
[47] H. Li, C.-M. Feng, Y. Xu, T. Zhou, L. Yao, and X. Chang, "Zero-shot camouflaged object detection," *IEEE Trans. Image Process.*, vol. 32, pp. 5126–5137, 2023.
[48] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL*, 2019, pp. 4171–4186.
[49] T. N. Kipf and W. Max, "Semi-supervised classification with graph convolutional networks," in *Proc. ICLR*, 2017.
[50] A. L. Maas et al., "Rectifier nonlinearities improve neural network acoustic models," in *Proc. ICML*, 2013, pp. 3–8.
[51] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1501–1510.
[52] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. ICML*, 2017, pp. 214–223.
[53] P. Zhou, K. Ying, Z. Wang, D. Guo, and C. Bai, "Self-supervised enhancement for named entity disambiguation via multimodal graph convolution," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 1, pp. 231–245, Jan. 2024.
[54] J. Achiam et al., "GPT-4 technical report," 2023, *arXiv:2303.08774*.
[55] Z. Yang et al., "The dawn of LMMs: Preliminary explorations with GPT-4 V(ision)," 2023, *arXiv:2309.17421*.
[56] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, pp. 303–338, Jun. 2010.
[57] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. ECCV*, 2014, pp. 740–755.
[58] Y. Kawano and K. Yanai, "Automatic expansion of a food image dataset leveraging existing categories with domain adaptation," in *Proc. ECCV Workshop*, 2015, pp. 3–17.
[59] S. Rahman, S. Khan, and N. Barnes, "Polarity loss: Improving visual-semantic alignment for zero-shot detection," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, pp. 1–13, Jun. 2022.
[60] P. Zhou, W. Min, Y. Zhang, J. Song, Y. Jin, and S. Jiang, "SeeDS: Semantic separable diffusion synthesizer for zero-shot food detection," in *Proc. ACM MM*, 2023, pp. 8157–8166.
[61] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. NeurIPS*, 2015, pp. 91–99.
[62] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[63] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015, pp. 1–15.
[64] A. Vaswani et al., "Attention is all you need," in *Proc. NeurIPS*, 2017, pp. 5998–6008.
[65] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1778–1785.
[66] T. Mikolov, É. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," in *Proc. LREC*, 2018, pp. 52–55.
[67] H. Li, J. Mei, J. Zhou, and Y. Hu, "Zero-shot object detection based on dynamic semantic vectors," in *Proc. ICRA*, 2023, pp. 9267–9273.
[68] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 86, pp. 2579–2605, 2008.
[69] D. Narayanan et al., "Efficient large-scale language model training on gpu clusters using megatron-LM," in *Proc. IEEE SC*, Nov. 2021, pp. 1–15.

Pengfei Zhou received the B.E. degree from the College of Computer Science, Zhejiang University of Technology, Hangzhou, China, in 2021. He is currently pursuing the M.E. degree with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. His research interests include multimedia processing, computer vision, and food computing.

Weiqing Min (Senior Member, IEEE) is currently an Associate Professor with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences (CAS). He has authored or coauthored more than 50 peer-referenced papers in relevant journals and conferences, including *Patterns* (Cell Press), *ACM Computing Surveys*, *Trends in Food Science and Technology*, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON IMAGE PROCESSING, *Food Chemistry*, ACM MM, AAAI, and IJCAI. His research interests include multimedia content analysis and food computing. He was a Senior Member of CCF. He was a recipient of the 2016 *ACM Transactions on Multimedia Computing, Communications, and Applications*, the Nicolas D. Georganas Best Paper Award, and the 2017 *IEEE Multimedia Magazine* Best Paper Award. He was the Guest Editor for the special issues on international journals, such as IEEE TRANSACTIONS ON MULTIMEDIA, IEEE MULTIMEDIA, and *Foods*.

Jiajun Song received the B.E. degree from the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China, in 2020, and the master's degree in computer science from the University of Chinese Academy of Sciences, Beijing, China, in 2023. He is currently pursuing the Ph.D. degree with the Renmin University of China. He was with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences. His research interests include food computing and fine-grained image recognition and retrieval.

Yang Zhang received the B.E. degree from Henan University, Kaifeng, China, in 2021. He is currently pursuing the M.E. degree with the University of Chinese Academy of Sciences, Beijing, China. He is with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing. His research interests include computer vision, food computing, and diffusion models.



Shuqiang Jiang (Senior Member, IEEE) is currently a Professor with the Institute of Computing Technology, Chinese Academy of Sciences (CAS), Beijing, China, and a Professor with the University of CAS. He is also with the Key Laboratory of Intelligent Information Processing, CAS. He has authored or coauthored more than 150 articles. He was supported by the National Science Fund for Distinguished Young Scholars in 2021, the NSFC Excellent Young Scientists Fund in 2013, and the Young Top-Notch Talent of Ten Thousand Talent Program in 2014. His research interests include multimedia analysis and multimodal intelligence. He is a Senior Member of CCF and a member of ACM. He has served as a TPC Member for more than 20 well-known conferences, including ACM Multimedia, CVPR, ICCV, IJCAI, AAAI, ICME, ICIP, and PCM. He received the Lu Jiaxi Young Talent Award from CAS in 2012 and the CCF Award of Science and Technology in 2012. He is the Vice Chair of the IEEE CASS Beijing Chapter and the ACM SIGMM China Chapter. He was the General Chair of ICIMCS in 2015 and the Program Chair of the 2019 ACM Multimedia Asia and PCM in 2017. He is an Associate Editor of *Multimedia Tools and Applications* and *ACM Transactions on Multimedia Computing, Communications, and Applications*.