International Journal of Computer Vision
https://doi.org/10.1007/s11263-024-02126-3

S.I. : OPEN-WORLD VISUAL RECOGNITION

Open-Vocabulary Animal Keypoint Detection with Semantic-Feature Matching

Hao Zhang[1,2] · Lumin Xu[3] · Shenqi Lai[4] · Wenqi Shao[2] · Nanning Zheng[1] · Ping Luo[5] · Yu Qiao[2] · Kaipeng Zhang[2]

Received: 11 December 2023 / Accepted: 16 May 2024

Abstract

Current image-based keypoint detection methods for animal (including human) bodies and faces are generally divided into fully supervised and few-shot class-agnostic approaches. The former typically relies on laborious and time-consuming manual annotations, posing considerable challenges in expanding keypoint detection to a broader range of keypoint categories and animal species. The latter, though less dependent on extensive manual input, still requires necessary support images with annotation for reference during testing. To realize zero-shot keypoint detection without any prior annotation, we introduce the *O*pen-*V*ocabulary *K*eypoint *D*etection (OVKD) task, which is innovatively designed to use text prompts for identifying arbitrary keypoints across any species. In pursuit of this goal, we have developed a novel framework named Open-Vocabulary *K*eypoint *D*etection with *S*emantic-feature *M*atching (KDSM). This framework synergistically combines vision and language models, creating an interplay between language features and local keypoint visual features. KDSM enhances its capabilities by integrating *D*omain *D*istribution *M*atrix *M*atching (DDMM) and other special modules, such as the *V*ision-*K*eypoint *R*elational *A*wareness (VKRA) module, improving the framework's generalizability and overall performance. Our comprehensive experiments demonstrate that KDSM significantly outperforms the baseline in terms of performance and achieves remarkable success in the OVKD task. Impressively, our method, operating in a zero-shot fashion, still yields results comparable to state-of-the-art few-shot species class-agnostic keypoint detection methods. Codes and data are available at https://github.com/zhanghao5201/KDSM.

Keywords Open vocabulary · Open set · Keypoint detection · Pose estimation

1 Introduction

Animal keypoint detection, a fundamental task in computer vision, is dedicated to identifying and localizing animals' keypoints within images. This task is pivotal for extensive analysis of animal (including human) bodies and faces. The

Communicated by Hong Liu.

This work was done during Hao Zhang's internship at Shanghai Artificial Intelligence Laboratory.

✉ Nanning Zheng
nnzheng@mail.xjtu.edu.cn

✉ Kaipeng Zhang
zhangkaipeng@pjlab.org.cn

Hao Zhang
zhanghao520@stu.xjtu.edu.cn

Lumin Xu
luminxu@link.cuhk.edu.hk

Shenqi Lai
laishenqi@qq.com

Wenqi Shao
shaowenqi@pjlab.orn.cn

Ping Luo
pluo@cs.hku.edu

Yu Qiao
qiaoyu@pjlab.org.cn

[1] National Key Laboratory of Human–Machine Hybrid Augmented Intelligence, National Engineering Research Center for Visual Information and Applications, and Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an 710049, Shaanxi, China

[2] Shanghai AI Laboratory, Xuhui, Shanghai, China

[3] The Chinese University of Hong Kong, Hong Kong, China

[4] Zhejiang University, Hangzhou, China

[5] The University of Hong Kong, Hong Kong, China

Published online: 25 June 2024

accurate location of these keypoints plays a vital role in various applications, ranging from in-depth behavioral studies to automated monitoring systems, such as animal pose tracking (Patel et al., 2023) and automatic assessment of animal pain (Feighelstein et al., 2022; Pessanha et al., 2023).

Traditional keypoint detection methodologies have primarily centered around developing complex neural network architectures (Andriluka et al., 2014; Fang et al., 2017; Newell et al., 2016; Tu et al., 2023; Wang et al., 2020; Xu et al., 2024; Zhang et al., 2023, 2024) and training them with datasets of annotated images to identify keypoints within specific species and keypoint categories. This strategy necessitates substantial manual labeling for each newly investigated species, often resulting in the creation of specialized datasets for these species (Brown et al., 2020; Khan et al., 2020; Koestinger et al., 2011; Lin et al., 2014; Labuguen et al., 2021), a process known to be both time-consuming and labor-intensive. For instance, compiling the AnimalWeb dataset (Khan et al., 2020) required a substantial manual labeling effort totaling 6,833 man-hours from both experts and trained volunteers. Despite such extensive manual efforts, the relatively limited availability and smaller size of animal keypoint datasets, compared to those for humans, present significant challenges in extending keypoint detection to new keypoint categories and animal species. The AnimalWeb dataset includes fewer than 239 annotations per species, in sharp contrast to the human-focused AFLW dataset (Koestinger et al., 2011), which contains 25,993 annotations. Furthermore, some species in the AnimalWeb dataset are represented by only a single annotated image, making cross-species keypoint detection even more challenging, especially for species that lack annotations. Advanced few-shot species class-agnostic keypoint detection methods, as extensively detailed in studies like (Shi et al., 2023; Xu et al., 2022), represent progress in reducing the reliance on extensive manual annotations to adapt new keypoint categories and animal species. As illustrated in Fig. 1a, these methods necessitate a small number of annotated support images for keypoint references during testing. In this paper, we further accomplish a more challenging task, which detects arbitrary keypoint in a zero-shot fashion without prior annotation during testing. Zero-shot keypoint detection could facilitate more convenient in-depth behavioral studies (Patel et al., 2023) and the development of automated monitoring systems (Feighelstein et al., 2022; Pessanha et al., 2023) for new species and keypoint categories.

The potential of vision-language models (VLMs) (Jia et al., 2021; Radford et al., 2021) inspires our approach. VLMs have shown success in joint modeling of visual and text information, contributing to their exceptional zero-shot learning ability in various tasks, including object detection, semantic segmentation, video classification, and others (Weng et al., 2023; Xu et al., 2023; Yao et al., 2022). However, there is a lack of research specifically addressing keypoint detection methods within this context. Motivated by VLM advancements, we introduce the language-driven *O*pen-*V*ocabulary *K*eypoint *D*etection (OVKD) task (unless otherwise specified, OVKD always refers to language-driven OVKD). Specifically, OVKD is designed to identify a broad spectrum of (*animal species*, *keypoint category*) pairs, including those not encompassed in the original training dataset. The term {*keypoint category*} refers to specific categories of keypoints, such as "eyes" and "nose." On the other hand, {*animal species*} represents a combination of the "target keypoint detection task" and the corresponding animal species, encompassing categories like "dog body," "dog face," "cat face," and "cat body." As shown in Fig. 1b, OVKD uses the image and text description of the keypoints to realize keypoint detection.

Building upon this concept, our initial strategy involves adopting a baseline framework (see Fig. 2 that utilizes language models to obtain text embeddings for the descriptions of (*animal species*, *keypoint category*) pairs. Then the baseline integrates the text embeddings with visual features using matrix multiplication and generates keypoint heatmaps. However, the limitation of this simple feature aggregation becomes evident in its lack of effective interaction between text and local visual features, hindering its ability to comprehend the local features of images and accurately localize specific keypoints. To address this, we emphasize the need for deeper interaction between text and local visual features of the image.

To overcome the limitations of the baseline framework in OVKD, we develop an advanced framework named Open-Vocabulary *K*eypoint *D*etection with *S*emantic-feature *M*atching (KDSM). KDSM introduces a *D*omain *D*istribution *M*atrix *M*atching (DDMM) technique and incorporates other special modules, such as a *V*ision-*K*eypoint *R*elational *A*wareness (VKRA) module, a keypoint encoder, a keypoint adapter, a vision head, and a vision adapter, among others. The VKRA module uses attention blocks to enhance the interaction between text embeddings and local keypoint features. This facilitates a deeper exploration and understanding of the complex relationships between various local keypoint locations and text prompts during training. Considering that the combinations of (*animal species*, *keypoint category*) pairs are virtually infinite, it becomes impractical to construct a heatmap channel for every pair like fully supervised and few-shot species class-agnostic keypoint detection methods. Therefore, we propose DDMM, which utilizes clustering techniques to group the text features of {*keypoint category*}. It allows semantically similar keypoint descriptions of different species to share a ground-truth heatmap channel representation during training. After grouping, the matching loss between text and heatmap features can be used to further align text features and keypoint visual features. During test-

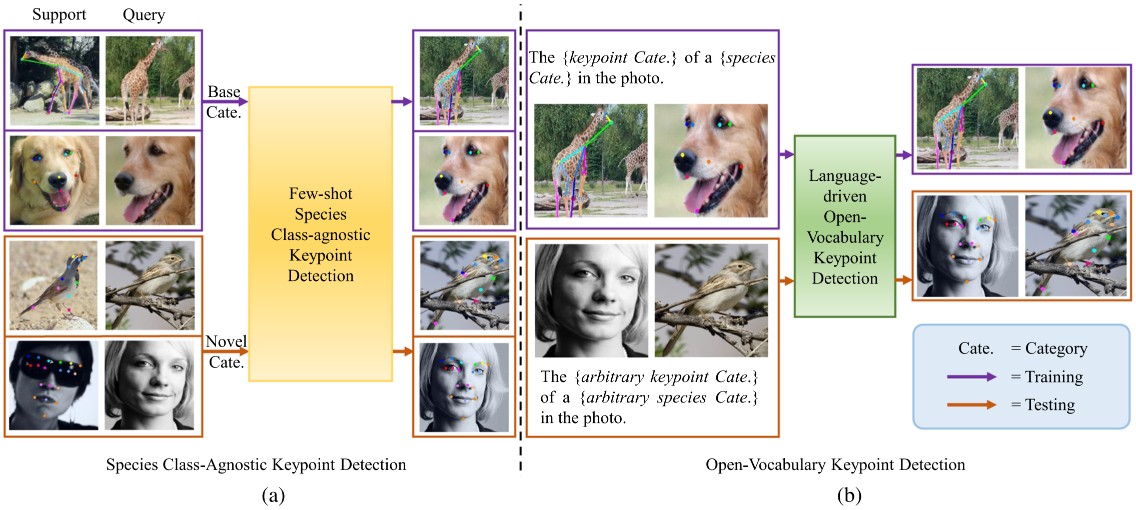

Support
Query
Base Cate.
Novel Cate.
Few-shot Species Class-agnostic Keypoint Detection
Species Class-Agnostic Keypoint Detection
(a)
The {*keypoint Cate.*} of a {*species Cate.*} in the photo.
The {*arbitrary keypoint Cate.*} of a {*arbitrary species Cate.*} in the photo.
Language-driven Open-Vocabulary Keypoint Detection
Cate. = Category
= Training
= Testing
Open-Vocabulary Keypoint Detection
(b)

Fig. 1 Few-shot species class-agnostic keypoint detection vs. language-driven open-vocabulary keypoint detection. **a** Current few-shot species class-agnostic keypoint detection needs support images for guidance during training and testing to detect keypoints in new species. **b** Language-driven OVKD aims to use text prompts that embed both {*animal species*} and {*keypoint category*} as semantic guidance to localize arbitrary keypoints of any species

ing, DDMM assigns new text descriptions to specific groups, enabling the capability of zero-shot keypoint detection.

We conduct extensive experiments to evaluate the efficacy of our proposed method. The results emphatically demonstrate that our KDSM framework excels in OVKD, significantly surpassing the performance of the baseline framework. Notably, KDSM exhibits impressive zero-shot capabilities and comparable performance to the state-of-the-art few-shot species class-agnostic keypoint detection methods. The primary contributions of our research are summarized as follows:

- We introduce the task of OVKD, designed to utilize text prompts for detecting a diverse range of keypoint categories across different animal species in a zero-shot fashion.
- We propose a pioneering approach, termed KDSM, to tackle the challenging OVKD task. DDMM technique and VKRA module are designed to model cross-species relationships and exchange vision-language information respectively.
- Extensive experiments show that KDSM excels in OVKD, surpassing the baseline framework substantially. Despite operating in a zero-shot manner, KDSM achieves comparable results with state-of-the-art few-shot keypoint detection methods.

2 Related Works

Traditionally, the main research direction in keypoint detection has been fully supervised methods. This approach concentrated on improving keypoint detection accuracy via advancements in neural network architectures (Andriluka et al., 2014; Fang et al., 2017; Newell et al., 2016; Tu et al., 2023; Wang et al., 2020; Xu et al., 2024; Zhang et al., 2023, 2024) and the development of new species datasets (Brown et al., 2020; Koestinger et al., 2011; Labuguen et al., 2021; Lin et al., 2014). However, these methods are confined to specific species or keypoint categories, limiting their adaptability to new types. Emerging few-shot category-agnostic keypoint detection techniques have started to address this, reducing the need for extensive annotations for novel species with a small number of annotated support images. We take this a step further by removing the necessity for image labeling and using language models to detect keypoints in a zero-fashion, open-vocabulary approach. In Sect. 2.1, we will present the few-shot category-agnostic keypoint detection methods. Section 2.2 will introduce related works on open-vocabulary learning, and Sect. 2.3 will discuss the recent integration of language models with vision tasks.

2.1 Advancements in Few-Shot Species Class-Agnostic Keypoint Detection

A significant advancement in keypoint detection is the advent of few-shot species class-agnostic techniques (Xu et al., 2022), which can identify keypoints across various animal species without category-specific training. However, these techniques commonly rely on “support images” during the training and testing phases. This reliance, characteristic of methods like MAML (Finn et al., 2017), Fine-tune (Nakamura &Harada, 2019), FS-ULUS (Lu &Koniusz, 2022), POMNet (Xu et al., 2022), and CapeFormer (Shi et al., 2023), limits their applicability to new species or keypoints.

Specifically, POMNet (Xu et al., 2022) initially proposed the few-shot species class-agnostic keypoint detection task and created the MP-100 expert dataset for it. CapeFormer (Shi et al., 2023) presents a two-stage framework incorporating techniques like a query-support refine encoder and a similarity-aware proposal generator for category-agnostic detection, shifting focus from heatmap prediction to keypoint position regression. In contrast, our proposed OVKD task moves away from reliance on support images. OVKD leverages text prompts containing both {*animal species*} and {*keypoint category*}, offering semantic guidance for detecting any keypoint in any species. This novel approach is aligned with zero-shot learning principles and marks a stride towards open-world animal body and facial keypoint detection.

2.2 Exploring Open-Vocabulary Learning in Computer Vision

Open-vocabulary learning, a burgeoning field in computer vision, has been explored in various tasks, including object detection (Bangalath et al., 2022; Yao et al., 2022), semantic segmentation (Li et al., 2022; Xu et al., 2023), 3D object recognition (Weng et al., 2023; Zhu et al., 2023) and video classification (Ni et al., 2022; Qian et al., 2022). The advent of vision-language models like CLIP (Radford et al., 2021) and ALIGN (Jia et al., 2021) has underscored their potential in tasks that require simultaneous processing of visual and text data, ideal for open-world learning scenarios.

While existing open-vocabulary learning research excels in image-level classification (Zhu et al., 2023), per-pixel classification (Li et al., 2022), and mask classification (Xu et al., 2023), keypoint detection poses a unique challenge. It demands not only a global understanding of the image but also precise localization of specific keypoints. To tackle this, we propose a novel technique called "Domain Distribution Matrix Matching." This technique transforms keypoint detection into a task of aligning semantic-feature distributions from input text prompts with the detected heatmaps, thereby enhancing the accuracy and efficiency of the detection process.

2.3 Leveraging Language Models for Vision Tasks

Leveraging language models for vision tasks has ushered in a new era of methodologies that significantly enhance machines' understanding and interpretation of visual data. Some works (Jia et al., 2021; Radford et al., 2021) utilize contrastive learning between language features and image features from vast collections of (image, text) pairs (e.g., 400 million in CLIP) to establish connections between language and visuals, which have marked the rapid development of using language models to aid in vision tasks. Specifically, open-vocabulary learning methods (Bangalath et al., 2022; Xu et al., 2023) employ pre-trained language and vision models to identify objects or scenes within images, showcasing extraordinary flexibility and adaptability. Furthermore, Large Vision-Language Models (Chen et al., 2023; Lin et al., 2024) integrate CLIP's image encoder into language models, greatly facilitating tasks like Image Captioning and Visual Question Answering, among others. Additionally, language models gradually play a crucial role in basic visual tasks such as language-assisted image generation (Li et al., 2024; Rombach et al., 2022) and multi-person pose estimation under occlusion (Hu et al., 2023). Meanwhile, applying language models to vision tasks also presents numerous ethical and security challenges (Zhang et al., 2024b). Therefore, our work is dedicated to sensibly utilizing language models to assist in the open-vocabulary keypoint detection task.

3 Method

In this section, we begin by defining the *O*pen-*V*ocabulary *K*eypoint *D*etection (OVKD) task in Sect. 3.1. We then present a baseline framework in Sect. 3.2, which offers a straightforward solution to the task. In Sect. 3.3, we introduce our proposed Open-Vocabulary *K*eypoint *D*etection with *S*emantic-feature *M*atching (KDSM) framework, outlining its unique design and capability.

3.1 Problem Formulation: Open-Vocabulary Keypoint Detection

We introduce a novel task termed OVKD for animal (including human) body and face keypoint localization. The goal of OVKD is to develop a framework capable of detecting arbitrary keypoints in images, even if the animal species or keypoint category is not present in the training data. The advancements in vision-language models such as CLIP (Radford et al., 2021), allow the keypoint detectors to take advantage of powerful language models to achieve language-driven OVKD.

For OVKD, text prompts are leveraged to guide the framework in understanding the semantic information and locating specific keypoints. Assuming we have a training set $\mathcal{D}_{train}$ and a test set $\mathcal{D}_{test}$, $\mathcal{D}_{train} = \{(\mathbf{I}, T(s_i, k_j)_{j=1}^{\mathbb{K}_{s_i}}, G(s_i, k_j)_{j=1}^{\mathbb{K}_{s_i}}\}_{i=1}^{\mathbb{S}}$, $\mathcal{D}_{test} = \{(\mathbf{I}, T(s_i', k_j')_{j=1}^{\mathbb{K}'_{s_i'}}, G(s_i', k_j')_{j=1}^{\mathbb{K}'_{s_i'}}\}_{i=1}^{\mathbb{S}'}$. Here, $\mathbf{I}$ represents images, $T(s_i, k_j)$ denotes the text prompts constructed based on species s_i and keypoint category k_j, and $G(s_i, k_j)$ denotes the ground-truth heatmaps constructed based on the locations of the species s_i and keypoint category k_j in the images $\mathbf{I}$. $\mathbb{S}$ and $\mathbb{K}_{s_i}$ represent the number of species and the number of keypoint categories of species s_i in the training set, respectively, while $\mathbb{S}'$ and $\mathbb{K}'_{s'_i}$ represent the number of species and the number of keypoint categories of species s'_i in the test set, respectively. The test set includes (*animal species*, *keypoint category*) pairs not covered in the training dataset, requiring the detector to identify arbitrary keypoints as per the text prompts.

3.2 Baseline: A Simple Framework for OVKD

To tackle the challenging OVKD task, we build a baseline framework that can predict arbitrary keypoint categories of any animal species as shown in Fig. 2. The baseline method constructs text prompts for the OVKD task and extracts text embedding using a Text_Encoder. The Vision_Encoder is applied to extract visual features of the input image simultaneously. Then, the visual and text features are integrated by matrix multiplication to output heatmaps of keypoints defined by text prompts.

3.2.1 Text Prompts Construction

In this step, we utilize the template "The {*keypoint category*} of a {*animal species*} in the photo." to assist language models in effectively grasping the task. For example, if "giraffe body" is the animal species and "neck" is the keypoint category, the prompt becomes: "The neck of a giraffe body in the photo." This consistent template is applied across various animals and keypoints, with placeholders adjusted accordingly. Utilizing this template enables the language model to concentrate on the interplay between animal species and keypoints, facilitating smooth generalization to new species and keypoints within the open-vocabulary framework. For the training and testing processes, the prompt construction is automatically generated using labeled datasets, i.e., {*keypoint category*} and {*animal species*} information. If users are testing the system via an API, manual input of category information is indeed necessary, which is consistent with the open-vocabulary learning works mentioned in Sect. 2.2.

3.2.2 Text Feature Extraction

Employing the pre-trained CLIP Text_Encoder (Radford et al., 2021), we process the preprocessed text prompts $T = \{T_1, T_2, \ldots, T_K\}$ for an image with K text prompts:

$$\mathbf{T} = \text{Keypoint\_Adapter}(\text{Text\_Encoder}(T)), \tag{1}$$

where $\text{Text\_Encoder}(T) \in \mathbb{R}^{K \times C_0}$ represents the extracted text features. Keypoint_Adapter is a two-layer Multi-layer Perceptron (MLP) used to refine these features and make them compatible with the image feature representations. This refinement produces a semantic feature space $\mathbf{T} \in \mathbb{R}^{K \times C}$ (with $K = 100, C = 64$ in our setup). K represents the maximum number of keypoint categories for each species that can be handled, which can be adjusted as long as it is greater than the maximum number of keypoint categories across all species. Due to the differences in the number of keypoints among different species, we insert $K - K_{valid}$ fixed invalid placeholder text features, where K_{valid} denotes the number of valid text prompts. The text features of the invalid placeholders are derived from the prompt "There is not the keypoint we are looking for."

3.2.3 Vision Feature Extraction

Given an input image I, we train a Vision_Encoder and a Vision_Head to extract image features:

$$\mathbf{V} = \text{Vision\_Head}(\text{Vision\_Encoder}(I)), \tag{2}$$

where $\mathbf{V} \in \mathbb{R}^{C \times hei. \times wid.}$ ($hei. = 64, wid. = 64$ in our implementation) represents vision feature. We utilize ResNet (He et al., 2016) as the backbone of the Vision_Encoder, which is known to be effective in extracting hierarchical visual features from images. The Vision_Head, inspired by SimpleBaseline (Xiao et al., 2018), is composed of three deconvolutional layers. These layers serve to upsample the low-resolution feature maps acquired from the image encoder, thereby successfully recovering spatial information and enabling accurate keypoint localization.

3.2.4 Keypoint Heatmap Prediction

The objective of this framework is to predict keypoint localization by aggregating semantic text and spatial visual features. To calculate the similarity between the text feature and pixel-level visual representation, the extracted features are combined through matrix multiplication:

$$\mathbf{H} = \mathbf{T} \times \mathbf{V}, \tag{3}$$

where $\mathbf{H} \in \mathbb{R}^{K \times hei. \times wid.}$ denotes predicted heatmaps. The framework supports multiple text prompt inputs for detecting several keypoints simultaneously. The model training is supervised using Mean Squared Error (MSE) loss between these predicted heatmaps $\mathbf{H}$ and the ground-truth heatmaps $\mathbf{G} \in \mathbb{R}^{K \times hei. \times wid.}$. In the construction of $\mathbf{G}$, each valid channel of the heatmap corresponds to a specific text prompt, which is in the form of "The {*keypoint category*} of a {*animal species*} in the photo." We apply a 2D Gaussian with a standard deviation of 2 pixels, centered on the ground-truth location of the keypoint described by the prompt. The process of generating the Gaussian kernel is consistent with HRNet (Wang et al., 2020) and POMNet (Xu et al., 2022).

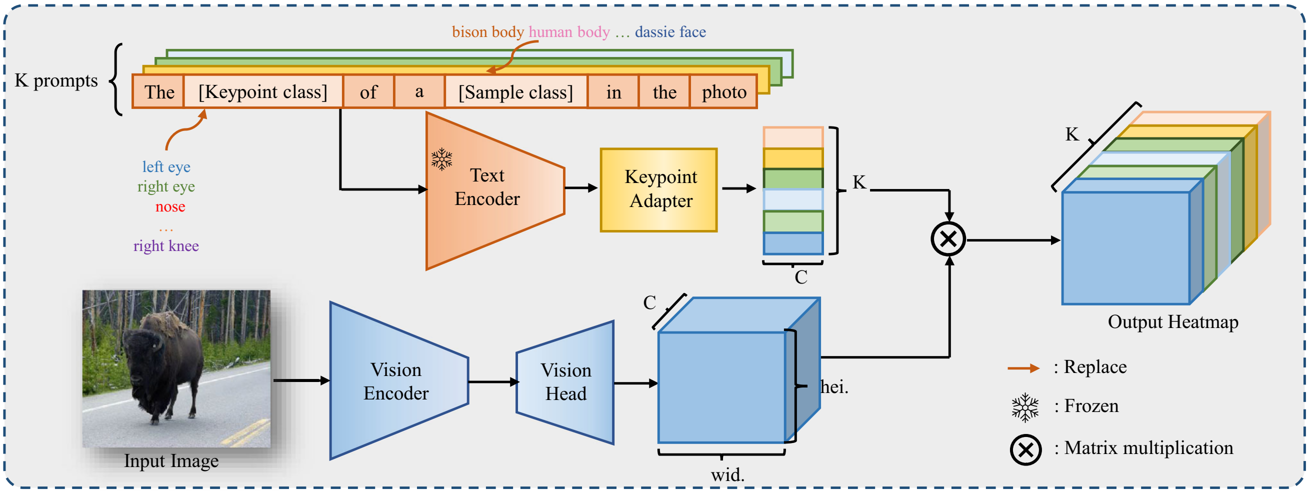


Fig. 2 An overview of the baseline method for OVKD. The baseline comprises a Vision_Encoder, a Text_Encoder, a Vision_Head and a Keypoint_Adapter. The Keypoint_Adapter is applied to optimize the relevance of text features with the image features and produce the text feature with the shape of C × K, where C and K represent the number of channel and text prompts, respectively. The Vision_Head produces the visual feature with the shape of C×hei.×wid., where hei. and wid. represent the height and width, respectively

Only the first K_{valid} heatmaps are valid for **G**, the other $(K - K_{valid})$ heatmaps are set to zero matrices. During the loss computation, only the first K_{valid} channels of **G** and **H** are used. During training, the Text_Encoder remains frozen, while other parameters are trainable. The matrix multiplication operation conducts a transformation of visual features to the output heatmap spaces, driven by the semantic information contained in the text prompts.

3.3 Open-Vocabulary Keypoint Detection with Semantic-feature Matching

In this section, we propose a novel framework, namely KDSM, to address the limitations of the baseline OVKD framework. The baseline framework just uses simple feature aggregation, which fails to effectively capture the intricate relationship between text and local visual features and establish clear connections between them, leading to less than optimal keypoint detection. Therefore, KDSM proposes *D*omain *D*istribution *M*atrix *M*atching (DDMM) and adopts some special modules to address the above problems, such as a *V*ision-*K*eypoint *R*elational *A*wareness (VKRA) module, a keypoint encoder, a keypoint adapter, a vision head, and a vision adapter, among others.

As depicted in Fig. 3, KDSM initially constructs text prompts and extracts text features similarly to the baseline approach. However, it then employs the VKRA module to facilitate a deeper exploration and understanding of the complex relationships between various local keypoint locations and text prompts during training. Finally, DDMM is proposed to capture cross-species keypoint-level relationships to further enhance the generalization ability of KDSM. Notably, KDSM supports multiple text prompt inputs for detecting several keypoints simultaneously.

3.3.1 Vision-Keypoint Relational Awareness Module

Within our framework, the VKRA module, incorporating a series of Transformer blocks inspired by Pan et al. (2020), is an essential design. It comprises two main components: Self-Attention (Vaswani et al., 2017) (Self_Attn.) and Cross-Attention (Carion et al., 2020) (Cross_Attn.). The self-attention layers are designed to enhance the interaction among text embeddings of the given sample. They amalgamate keypoint features as follows:

$$\mathbf{Y}_t = \text{Self\_Attn.}(\text{Text\_Encoder}(\mathbf{T})). \tag{4}$$

The refined keypoint features $\mathbf{Y}_t$ elucidate the relationships of text semantic concepts among the keypoints of a specific species.

The cross-attention layers use the output features from the Vision_Encoder as the query, while the refined features $\mathbf{Y}_t$ serve as the key and value. This mechanism facilitates interaction between the context-aware visual features Vision_Encoder(I) and the refined features $\mathbf{Y}_t$ to enhance the vision representation:

$$\widetilde{\mathbf{V}} = \text{Cross\_Attn.}(\text{Vision\_Encoder}(I), \mathbf{Y}_t) \tag{5}$$

The updated visual features $\widetilde{\mathbf{V}}$ effectively capture the relationships between local visual features and keypoint text features,

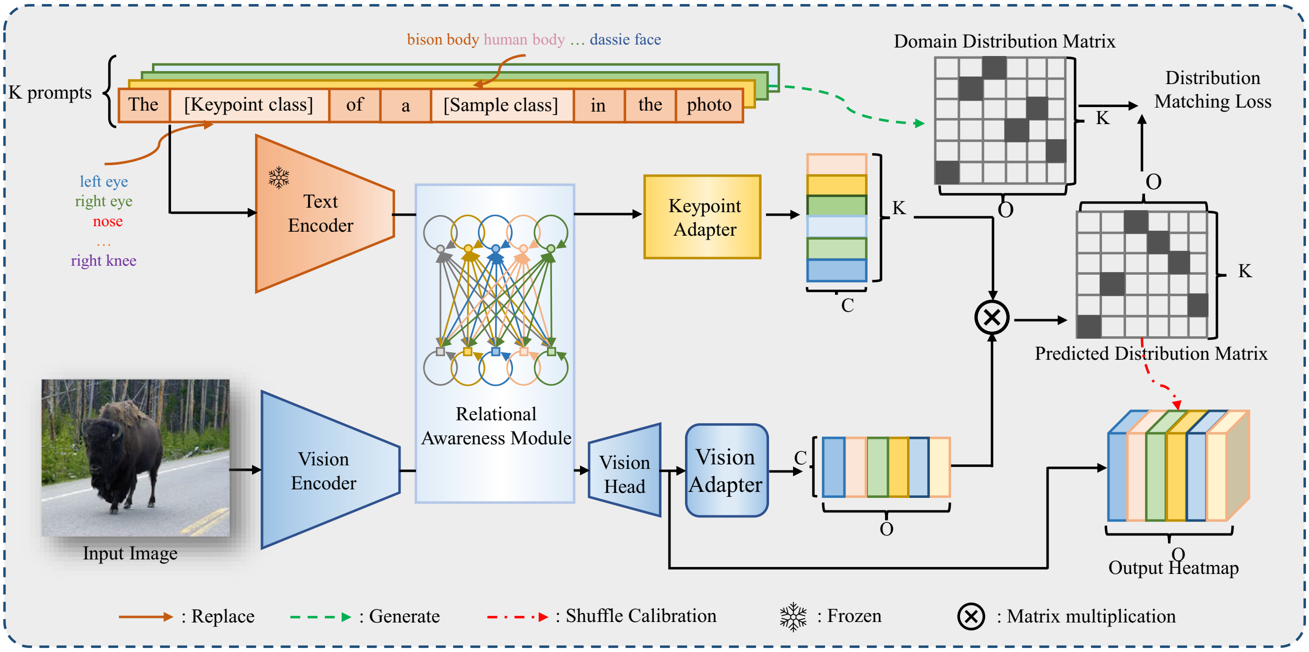


Fig. 3 An overview of KDSM. KDSM comprises a Vision_Encoder, a Text_Encoder, a Keypoint_Adapter, a Vision_Adapter and a Vision_Head similar to the baseline. The vision-keypoint relational awareness module adjusts visual features according to their associations with keypoints. The Vision_Adapter is employed to modify the feature shape so that it matches the text features' shape. Similarity is calculated between the adjusted features and text semantic features, resulting in a predicted distribution matrix. The predicted distribution matrix and the text domain distribution matrix are then utilized to compute matching loss

bridging the gap between vision representation and the keypoint.

3.3.2 Domain Distribution Matrix Matching

To model cross-species keypoint-level relationships, we propose a domain distribution matrix that links keypoint categories to corresponding output heatmaps. Assuming we have 78 species and 15 keypoint categories per species, this leads to 1170 hypothetical (*animal species*, *keypoint category*) combinations. Directly representing each combination with a unique heatmap channel to model the above relationships is impractical. There exists cross-species commonality at the keypoint category level for OVKD since the keypoints of different animals may be similar. The similarity could be grasped during training by dividing all the keypoint categories into several groups and learning keypoint categories in the same group together. Therefore, we opt to represent multiple keypoint categories of different species using a single channel of ground-truth heatmaps. By grouping all keypoint categories and learning them collectively within these groups (where each group corresponds to one heatmap channel), we enhance the efficiency of the training process and avoid unnecessary computational expenditure. Notably, only keypoint categories of different species that are clustered into the same group will share the same ground-truth heatmap channel, and all keypoint categories of the same species are clustered into different groups (ground-truth heatmap channels) in our setting. During testing, a new (*animal species*, *keypoint category*) combination is assigned to one of the predefined groups based on the predicted distribution matrix. The heatmap representation of the selected group is then utilized to detect keypoints for that specific combination. Consequently, domain distribution matrix matching plays a crucial role in enhancing the prediction of new keypoint categories across various species.

Specifically, we apply K-means clustering to all training set keypoint categories, dividing them into O groups based on text embeddings generated by the Text_Encoder from $\{keypoint\ category\}$ terms, we set that all $\{keypoint\ category\}$ of the same species must belong to different groups in the clustering process. We then pre-compute a binary domain distribution matrix $\mathbf{D} \in \mathbb{R}^{K \times O}$ (setting $O = 100$) for each training sample, based on its keypoint categories. Here, K is a constant no smaller than any sample's maximum keypoint count. We set $\mathbf{D}_{ij} = 1$ when the i-th keypoint falls into the j-th group. If a sample's keypoint count K' is less than K, $\mathbf{D}_{ij} = 0$ for $i \in [K' + 1, K]$ and $j \in [1, O]$.

To learn group selection, we predict the distribution matrix. First, the updated visual features $\widetilde{\mathbf{V}}$ are merged with the original features to strengthen visual representation. These features pass through the Vision_Head, identical to the baseline, to generate heatmaps $\mathbf{H}' \in \mathbb{R}^{O\times hei.\times wid.}$. The Vision_Adapter then generates the visual features $\mathbf{V}'$ from these heatmaps, and the Keypoint_Adapter adapts the original text embeddings and generates $\mathbf{T}'$. Finally, we measure the similarity between the adjusted visual features $\mathbf{V}' \in \mathbb{R}^{C\times O}$ and the adjusted text embeddings $\mathbf{T}' \in \mathbb{R}^{K\times C}$ to create a predicted distribution matrix $\mathbf{P} \in \mathbb{R}^{K\times O}$:

$$\mathbf{P} = \mathbf{T}' \times \mathbf{V}'. \tag{6}$$

3.3.3 Loss Function

The matching loss L_{match}, computed as the cross-entropy loss between the predicted distribution matrix $\mathbf{P}$ and the domain distribution matrix $\mathbf{D}$, aims to align keypoint categories with heatmap channels:

$$L_{match} = -\sum_{i=1}^{K}\sum_{j=1}^{O} \mathbf{D}_{ij} \log \mathbf{P}_{ij}. \tag{7}$$

During training, we utilize the annotated domain distribution matrix $\mathbf{D}$ to determine the j-th heatmap position for the i-th prompt, addressing the mismatch between predicted heatmap ordering and prompt sequencing. During testing, alignment is achieved using the predicted domain distribution matrix $\mathbf{P}$. Subsequently, heatmaps $\mathbf{H}'$ produced by Vision_Head are reorganized across channels based on $\mathbf{D}$ during training or $\mathbf{P}$ during testing to ensure correct alignment with their respective prompts. This reordering involves identifying the index o of the element 1 in the ith row of $\mathbf{D}$ or $\mathbf{P}$, signifying the oth channel of $\mathbf{H}'$ as matching the ith prompt. PyTorch functions like "torch.index_select" facilitate this reordering process. The reordered heatmaps $\mathbf{H} \in \mathbb{R}^{O\times hei.\times wid.}$ are then evaluated against the ground-truth heatmaps $\mathbf{G} \in \mathbb{R}^{O\times hei.\times wid.}$ using the Mean Squared Error (MSE) loss. The initial K_{valid} channels of $\mathbf{G}$ correspond to keypoint locations identified by the K_{valid} text prompts, while the remaining $O - K_{valid}$ channels are treated as invalid zero matrices. The overall training loss for KDSM is defined as:

$$L_{total} = \alpha L_{match} + \beta MSE(\mathbf{H}, \mathbf{G}) \tag{8}$$

where α and β are the balance weights, and they are set to $1e^{-6}$ and 1 unless otherwise specified. The process of generating $\mathbf{G} \in \mathbb{R}^{O\times hei.\times wid.}$ ($O = 100$, $hei. = 64$, $wid. = 64$ in our implementation.) mirrors that of the baseline in Sect. 3.2, except for the total number of channels. In KDSM, the total number of channels O in $\mathbf{G}$ is predefined as the number of clusterings, distinct from K in the baseline, which represents the number of prompts.

3.3.4 Inference Process

During the inference phase, when presented with an input image and corresponding text prompts, KDSM replicates its training methodology to estimate the keypoint heatmaps and the predicted distribution matrix. This process involves a detailed analysis for each keypoint category k. Specifically, we search for the maximum value in the k-th row of the predicted distribution matrix $\mathbf{P}$, which identifies the index of the corresponding heatmap channel for that particular keypoint.

Once the indexes are determined, the heatmaps are carefully reordered and calibrated to align with these indexes, thus serving as the final prediction results. This step is crucial in ensuring the accuracy of our keypoint localization. Subsequently, the keypoint localization is precisely decoded as the coordinates that correspond to the highest scores within these reordered heatmaps.

In our experiment, we simply use the maximum value indexing as mentioned earlier. Our statistical analysis showed no samples of different keypoints corresponding to the same heatmap. However, variations in the test set might result in overlapping assignments, which motivates us to develop a fast-indexing algorithm (Algorithm 1. Algorithm 1 does not affect the accuracy of our experimental results. Furthermore, due to semantic similarities and pose variations, it is normal and acceptable for multiple keypoints to occasionally map to the same heatmap. Therefore, Algorithm 1 is offered as an optional solution, allowing users to choose based on their specific requirements.

Algorithm 1 Assign Heatmaps to Keypoints Based on the Predicted Domain Distribution Matrix During Inference

```
Require: Predicted Domain Distribution Matrix $\mathbf{P}$ of size $K \times O$, where
    $K$ is the number of keypoints and $O$ is the number of heatmaps.
Ensure: $L$: A list of heatmap indices assigned to each keypoint.
Initialize a priority queue $Q$.
Initialize an empty set of assigned heatmaps $A_O$.
Initialize an empty set of assigned keypoints $A_K$.
Initialize the assignments list $L$ with $-1$ for each keypoint.
for $i = 1$ to $K$ do
    for $j = 1$ to $O$ do
        Add $(\mathbf{P}[i, j], i, j)$ to the priority queue $Q$.
    end for
end for
while not $Q$.isEmpty() AND $|A_K| < K$ do
    Extract the maximum score entry $(score, k, o)$ from $Q$.
    if $k \notin A_K$ AND $o \notin A_O$ then
        Assign heatmap $o$ to keypoint $k$: $L[k] = o$.
        Add $k$ to the set of assigned keypoints $A_K$.
        Add $o$ to the set of assigned heatmaps $A_O$.
    end if
end while
return $L$
```

4 Experiments

4.1 Open-Vocabulary Evaluation Protocol

4.1.1 Dataset Split

MP-100 (Xu et al., 2022) is introduced for category-agnostic pose estimation, which contains over 20K instances covering 100 sub-categories and 8 super-categories (human hand, human face, animal body, animal face, clothes, furniture, and vehicle). However, some of the keypoint categories in MP-100, such as those for clothes and furniture, lack practical semantic information and are not suitable for language-driven OVKD. Thus, we selected a subset of 78 animal categories (including humans) with keypoint annotations that have specific, meaningful semantic information. We call this subset "MP-78", including COCO (Lin et al., 2014), AFLW (Koestinger et al., 2011), OneHand10K (Wang et al., 2018), AP-10K (Yu et al., 2021), Desert Locust (Graving et al., 2019), MascaquePose (Labuguen et al., 2021), Vinegar Fly (Pereira et al., 2019), AnimalWeb (Khan et al., 2020), CUB-200 (Welinder et al., 2010).

MP-78 encompasses more than 14,000 images accompanied by 15,000 annotations. For keypoint types possessing semantic meaning, albeit lacking a precise definition or description, we employ ChatGPT to query and acquire the names of these keypoints. For example, we use a query like "How to anatomically describe the second joint of the index finger?" to obtain the name of a specific keypoint. All these queries are performed manually, and then we build the dataset MP-78.

It is essential to clarify that in this paper, {*animal species*} refers to a combination of "target keypoint detection task + animal species." For instance, the face and body of a dog are categorized as two distinct {*animal species*} entities (i.e., "dog face" and "dog body"), based on the specific keypoint detection task. This means that our definition of species extends beyond mere biological classification, encapsulating task-specific categories within each animal.

To evaluate the generalization ability of OVKD to different keypoint categories and animal species, we design two settings, that is "Setting A: Diverse Keypoint Categories" for new {*keypoint category*}, and "Setting B: Varied Animal Species" for new {*animal species*} like (Xu et al., 2022). All zero-shot settings strictly fall under "transductive generalized zero-shot learning (Pourpanah et al., 2022)".

In Setting A, we divide the keypoint categories associated with each of the 78 species into two parts: seen {*keypoint category*} and unseen {*keypoint category*}. During training, we only used the seen categories, while the unseen categories were reserved for testing. For fair evaluation, we randomly split seen {*keypoint category*} for each species to form seen {*keypoint category*} sets. We form five different train/test sets splits.

In Setting B, MP-78 is split into train/test sets, with 66 {*animal species*} for training, and 12 {*animal species*} for testing. To ensure the generalization ability of the framework, we evaluate the framework performance on five splits like (Xu et al., 2022), where each {*animal species*} is treated as a novel one on different splits to avoid {*animal species*} bias.

4.1.2 Evaluation Metrics

We employ the Probability of Correct Keypoint (PCK) and Normalized Mean Error (NME) metrics to assess the accuracy of keypoint detection. To mitigate category bias, we compute and present the average PCK and average NME across all dataset splits. This approach ensures a balanced and thorough evaluation of our model's performance in keypoint detection.

PCK measures the accuracy of a predicted keypoint by comparing its normalized distance to the actual ground-truth location, with respect to a predefined threshold (σ). In line with the methodologies of POMNet (Xu et al., 2022) and CapeFormer (Shi et al., 2023), we report PCK@0.2 results in our experiments, setting σ to 0.2 for each category across all dataset splits. Additionally, we report PCK@0.05, where σ is set to 0.05, demanding more precise predictions compared to $\sigma = 0.02$. NME is defined similarly to HRNet V2 (Wang et al., 2020), where the normalization distance refers to the longest side of the ground-truth bounding box.

4.2 Implementation Details

In our setup, the default Vision_Encoder is ResNet50 (He et al., 2016), pre-trained on the ImageNet dataset (Deng et al., 2009) by default unless otherwise specified. The Self_Attn. module consists of three layers, each featuring a multi-head self-attention mechanism and a feed-forward neural network (FFN). This self-attention component is equipped with four attention heads and an embedding dimension of 512, with a dropout rate set at 0.1. The FFN includes two fully connected layers, an embedding dimension of 512, and 2048 feedforward channels. We employ ReLU as the activation function and maintain a dropout rate of 0.1. The Cross_Attn. component also comprises three layers. Each layer incorporates a multi-head self-attention mechanism, a multi-head cross-attention mechanism, and an FFN. The FFN configuration mirrors that of the Self_Attn. For text encoding, we default to using CLIP (Radford et al., 2021)'s Text_Encoder, pre-trained alongside the ViT-B/32 Vision_Encoder on image-text paired data, unless an alternative specification is provided.

The objects of interest are extracted using their bounding boxes and resized to dimensions of 256 × 256. To bolster the model's generalization capabilities, data augmentation techniques such as random scaling (varying from −15 to 15%) and random rotation (varying from −15° to 15°) are applied. Training is carried out across 4 GPUs, each with a batch size of 64, for a total of 210 epochs.

4.3 Results for OVKD

4.3.1 Setting A: Diverse Keypoint Categories

Table 1 presents the performance comparison between the baseline framework and KDSM on the MP-78 dataset for this setting. The table highlights that KDSM consistently surpasses the baseline in all five dataset splits. The quantitative comparison of the results shows a significant performance improvement when using the KDSM framework. The Mean (PCK@0.2) score across all five splits increases from 42.93 for the baseline to 88.23 for the KDSM framework, resulting in a remarkable enhancement of 45.30 points. Similarly, the Mean (PCK@0.05) score increases from 11.93 for the baseline to 62.35 for the KDSM framework, indicating a substantial enhancement of 50.42 points. In addition, the Mean NME decreases from 29.72 for the baseline to 7.93 for the KDSM framework, showcasing an improvement of 21.79 points. This indicates that the KDSM approach is more effective at handling the "Diverse Keypoint Categories" setting in the zero-shot fashion. The superior performance of the KDSM framework on the "Diverse Keypoint Categories" setting can be attributed to its capacity to better align and match semantic information from text prompts with local visual features, as well as its ability to effectively transfer knowledge to unseen (*animal species*, *keypoint category*) pairs.

4.3.2 Setting B: Varied Animal Species

Table 2[1] displays the performance comparison between the baseline framework and KDSM on the MP-78 dataset for the "Varied Animal Species" setting under a zero-shot setting. Additionally, it compares the results with class-agnostic keypoint detection methods under 1-shot and 5-shot settings.

The KDSM framework significantly outperforms the baseline in the zero-shot setting, demonstrating its effectiveness in handling unseen animal species without category-specific training. The enhanced performance of the KDSM framework is due to the efficient knowledge transfer from seen to unseen {*animal species*}. Besides, recent research (Shi et al., 2023; Xu et al., 2022) has developed few-shot species class-agnostic keypoint detection techniques that can identify keypoints across various animal species without category-specific training. However, these techniques typically rely on support images with annotations during both the training and testing phases. In contrast, our OVKD approach using the KDSM framework does not require support images by leveraging text prompts {*animal species*} and {*keypoint category*} for semantic guidance.

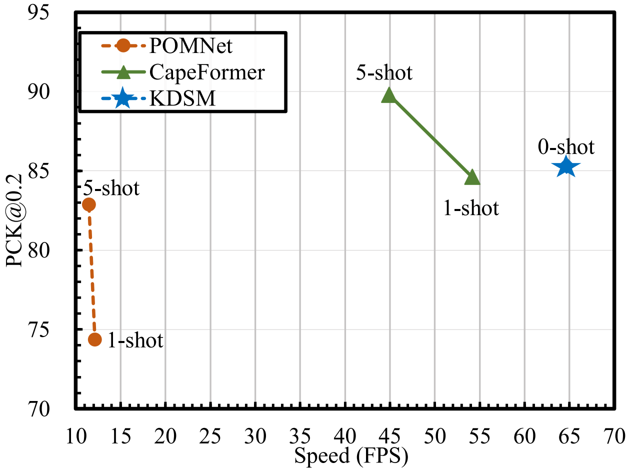


Fig. 4 Comparison of the trade-off between PCK@0.2 and Speed for Setting B. The speed is measured using Frames Per Second (FPS) on a single NVIDIA A100-SXM-80GB card. The test is conducted using an average of 1000 images for one species

OVKD and few-shot species class-agnostic keypoint detection represent distinct methodological approaches, making direct comparisons challenging, so we primarily benchmark against our baseline. However, we also highlight the performance gap contrast with few-shot species class-agnostic keypoint detection methods at a macro level. Our method demonstrates comparable results to these few-shot species class-agnostic keypoint detection approaches and outperforms the state-of-the-art 1-shot solution, CapeFormer (Shi et al., 2023), across all three metrics. This emphasizes the effectiveness of our approach. Furthermore, our zero-shot OVKD even surpasses the 5-shot setting of FS-ULUS (Lu &Koniusz, 2022), MAML (Finn et al., 2017), Fine-tune (Nakamura &Harada, 2019), and POMNet (Xu et al., 2022) across all three metrics. It should be noted that methods like POMNet and CapeFormer have limitations during training as they cannot access images of new categories and rely on support images during testing. Hence, it is reasonable for our zero-shot method to exhibit superior performance compared to few-shot solutions. In particular, when considering the PCK@0.05 metric, we outperform the state-of-the-art CapeFormer by 9.25 points (56.20 vs. 46.95). It is worth noting that PCK@0.05 requires more precise predictions of keypoint locations compared to the less stringent PCK@0.2 metric. By evaluating keypoint detection performance using different metrics such as PCK and NME, we provide a comprehensive analysis of our method's performance.

[1] We refer to the method "*F*ew-*s*hot keypoint detection with *u*ncertainty *l*earning for *u*nseen *s*pecies" as FS-ULUS.

Table 1 Comparisons with the baseline framework on the MP-78 dataset for Setting A with PCK@0.2, PCK@0.05 and NME

Metric	Framework	Split1	Split2	Split3	Split4	Split5	Mean metric
PCK@0.2 ↑	Baseline	42.02	44.00	42.55	43.80	42.26	42.93
	KDSM	**87.93**	**88.50**	**87.64**	**88.28**	**88.82**	**88.23**
	Baseline	11.08	11.44	10.35	14.98	11.80	11.93
PCK@0.05 ↑	KDSM	**62.80**	**63.11**	**61.91**	**62.00**	**61.91**	**62.35**
	Baseline	29.96	29.18	30.54	29.31	29.60	29.72
NME ↓	KDSM	**8.20**	**7.55**	**8.23**	**7.81**	**7.84**	**7.93**

Best results are indicated in bold
↑ indicates higher is better, while ↓ indicates lower is better

Table 2 Comparisons on MP-78 dataset for Setting B

Metric	Framework	Shot setting	Split1	Split2	Split3	Split4	Split5	Mean metric
PCK@0.2 ↑	MAML (Finn et al., 2017)	5-shot	76.37	75.53	71.15	69.46	67.55	72.01
	Fine-tune (Nakamura &Harada, 2019)	5-shot	77.81	76.51	72.55	71.09	69.85	73.56
	FS-ULUS (Lu &Koniusz, 2022)	5-shot	78.34	79.67	76.89	81.52	75.23	78.33
	POMNet (Xu et al., 2022)	5-shot	81.25	86.44	81.01	86.93	78.68	82.86
	CapeFormer (Shi et al., 2023)	5-shot	**91.01**	**90.95**	**87.90**	**91.90**	**87.23**	**89.80**
	MAML (Finn et al., 2017)	1-shot	75.11	74.31	69.80	68.22	67.44	70.98
	Fine-tune (Nakamura &Harada, 2019)	1-shot	76.65	76.41	71.37	69.97	69.36	72.75
	FS-ULUS (Lu &Koniusz, 2022)	1-shot	73.69	70.65	63.97	71.14	63.65	68.62
	POMNet (Xu et al., 2022)	1-shot	73.07	77.89	71.79	78.76	70.26	74.35
	CapeFormer (Shi et al., 2023)	1-shot	85.41	88.39	83.53	85.74	80.04	84.62
	Baseline	Zero-shot	56.06	55.36	54.35	53.07	50.66	53.90
	KDSM	Zero-shot	**85.48**	**89.45**	**84.29**	**86.25**	**81.17**	**85.33**
PCK@0.05 ↑	CapeFormer (Shi et al., 2023)	5-shot	**46.90**	**51.90**	**44.45**	**52.30**	**39.21**	**46.95**
	CapeFormer (Shi et al., 2023)	1-shot	40.59	44.13	35.59	42.34	33.00	39.13
	Baseline	Zero-shot	32.40	32.20	29.37	30.67	27.13	30.35
	KDSM	Zero-shot	**60.26**	**61.17**	**55.08**	**55.96**	**48.53**	**56.20**
	CapeFormer (Shi et al., 2023)	5-shot	**8.63**	**7.81**	**9.85**	**8.02**	**10.15**	**8.89**
NME ↓	CapeFormer (Shi et al., 2023)	1-shot	10.84	9.58	11.77	10.65	13.16	11.20
	Baseline	Zero-shot	23.78	25.21	25.62	25.92	26.30	25.37
	KDSM	Zero-shot	**9.71**	**8.04**	**10.96**	**9.58**	**12.16**	**10.09**

KDSM notably demonstrates comparable performance on par with other few-shot species class-agnostic keypoint detection approaches. We use different colors to show the **best**and **second-best** results respectively

4.3.3 Inference Speed

In Fig. 4, we compare the trade-off between PCK@0.2 and Inference Speed (Frames Per Second) with state-of-the-art few-shot solutions, namely POMNet (Xu et al., 2022) and CapeFormer (Shi et al., 2023). The speed is reported as an average of 1000 test images. As shown in the figure, it is evident that our KDSM method surpasses POMNet (Xu et al., 2022) in both average speed and accuracy. Furthermore, our approach exhibits a significant speed advantage compared to CapeFormer (Shi et al., 2023). These findings highlight the promising prospects of our method for practical applications.

4.3.4 Long-tail Animal Species

AnimalWeb (Khan et al., 2020) is a long-tail keypoint detection dataset consisting of 350 different animal species. The number of annotated images per species ranges from 1 to 239, reflecting the varying difficulty in data collection and substantial species imbalance. We prepare the data of uncommon animal species from AnimalWeb by first excluding the categories that are already present in MP-78. Then, we sort the remaining species on the AnimalWeb dataset based on the number of annotated samples. 280 species with relatively smaller number of samples are evenly divided into four partitions, denoted as W-SP1, W-SP2, W-SP3, and W-SP4, and each partition contains 70 species. Notably, W-SP1 consists

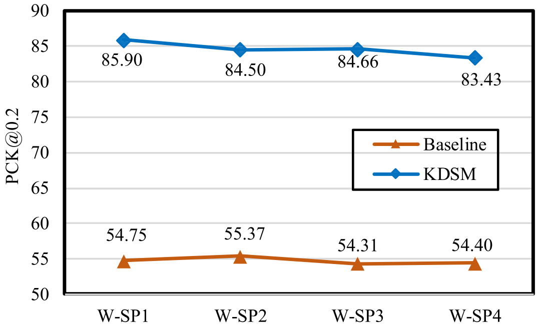


Fig. 5 Comparisons of the performance (PCK@0.2) between the baseline and KSDM on long-tail species for the AnimalWeb dataset

of the most common species, while W-SP4 represents the long-tail species with the fewest annotations. We evaluate the baseline and KDSM in each partition using the five models trained in setting B, and report the average PCK@0.2 across the five models as the final result for each method, respectively. As shown in Fig. 5, even on the long-tail species set W-SP4, KDSM achieves a PCK@0.2 score of 83.43, which is comparable with the relatively common species set W-SP1. This demonstrates the robustness of KDSM in handling long-tail categories. We also observe that KDSM gets a slightly lower result for W-SP2 compared to W-SP3 (84.50 vs. 84.66), which is expected due to inherent differences between species, including variations in pose and other factors, indicating that the detection accuracy is not solely determined by species prevalence. Furthermore, similar to the results shown in Table 2, there is a noticeable performance gap between the baseline and KDSM, indicating the superiority of KDSM in the OVKD task.

4.4 Ablation Study

In this section, we do some ablation experiments about the hyperparameter settings of the loss function, domain distribution matrix matching, VKRA module, Vision_Encoder and Text_Encoder. The default setting is $\alpha = 1e^{-3}$ and $\beta = 1$.

4.4.1 Discussion of the Loss Function of KDSM

We explore various hyperparameter configurations in this section. Table 3 illustrates how these settings impact KDSM's performance in the OVKD Setting A evaluation. We observe that as the value of α is reduced from 1 to 10^{-10}, while maintaining β at a constant 1, the Mean (PCK@0.2) shows an increasing trend. The optimal performance is attained at $\alpha = 10^{-6}$, resulting in a Mean (PCK@0.2) of 88.23. Conversely, further reducing α below 10^{-6}, or setting it to 0, leads to a decrease in Mean (PCK@0.2), suggesting an ideal range for α's value. Notably, when α is set to 0, the Mean (PCK@0.2) falls sharply to 31.15, underscoring the significance of domain distribution matrix matching.

4.4.2 Domain Distribution Matrix Matching

Table 4 demonstrates a significant improvement in Mean (PCK@0.2) scores with the inclusion of DDMM. In setting A, the Mean (PCK@0.2) is enhanced from 42.93 to 65.89, while in setting B, Mean (PCK@0.2) is elevated from 53.90 to 73.59. This substantial increase attests to DDMM's effectiveness in promoting knowledge transfer between seen and unseen keypoint categories. Moreover, the uniform improvement across all dataset splits underscores the robustness and adaptability of our proposed method, emphasizing its suitability for diverse real-world applications.

4.4.3 Vision-Keypoint relational Awareness Module

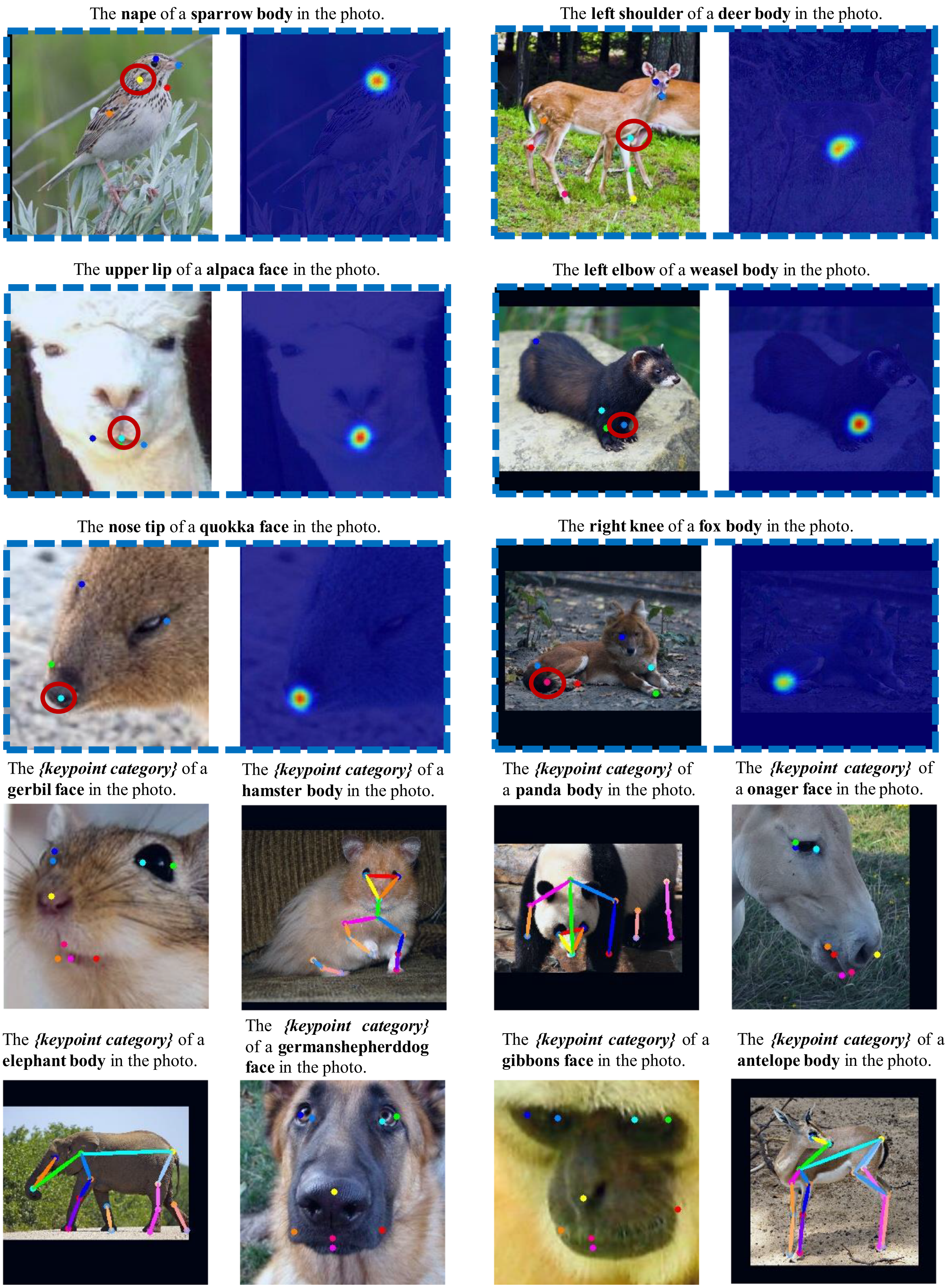
Table 4 shows that integrating the baseline framework with both DDMM and VKRA Module leads to a notable increase in Mean (PCK@0.2) scores. Specifically, the Mean (PCK@0.2) rises from 42.93 in the baseline without these components to 76.30 in setting A and from 53.90 to 83.74 in setting B when incorporating both DDMM and VKRA modules. This improvement underscores the critical necessity of the VKRA module in our methodology, as it adeptly discerns the semantic connections between visual features and text prompts, thereby enhancing generalization capabilities for unseen keypoint categories.

4.4.4 Attention Layers in Vision-Keypoint relational Awareness Module

Our study also delves into the optimal number of self-attention and cross-attention layers within the VKRA module. The findings, as depicted in Table 5, indicate that augmenting the number of self-attention blocks from 1 to 3 leads to a marked improvement in performance (compare row 1 with row 3). However, adding a fourth self-attention block doesn't contribute substantially to further gains (compare row 3 with row 4). A similar pattern is observed with the number of cross-attention blocks, leading us to implement three cross-attention blocks in our final configuration.

4.4.5 Discussion on the Choice of Vision Encoder

Following previous research (Ni et al., 2022), we deviate from using the frozen CLIP visual encoder and instead train a task-specific visual encoder, but we still leverage the language model's knowledge (that is why we can achieve OVKD). The results of deploying various Vision Encoders such as MobileNet V2 (Sandler et al., 2018), EfficientNet-B0 and B3 (Tan &Le, 2019), as well as ResNet50 (He et



The **nape** of a **sparrow body** in the photo.

The **left shoulder** of a **deer body** in the photo.

The **upper lip** of a **alpaca face** in the photo.

The **left elbow** of a **weasel body** in the photo.

The **nose tip** of a **quokka face** in the photo.

The **right knee** of a **fox body** in the photo.

The *{keypoint category}* of a **gerbil face** in the photo.

The *{keypoint category}* of a **hamster body** in the photo.

The *{keypoint category}* of a **panda body** in the photo.

The *{keypoint category}* of a **onager face** in the photo.

The *{keypoint category}* of a **elephant body** in the photo.

The *{keypoint category}* of a **germanshepherddog face** in the photo.

The *{keypoint category}* of a **gibbons face** in the photo.

The *{keypoint category}* of a **antelope body** in the photo.

Fig. 6 Visual results of KDSM on the test sets of two experiment settings of OVKD. The first three rows show the heatmaps for Setting A, and the last two rows show the results for Setting B. KDSM achieves satisfactory results in both two settings. Due to space limitations, we use {*keypoint category*} to represent the keypoint categories

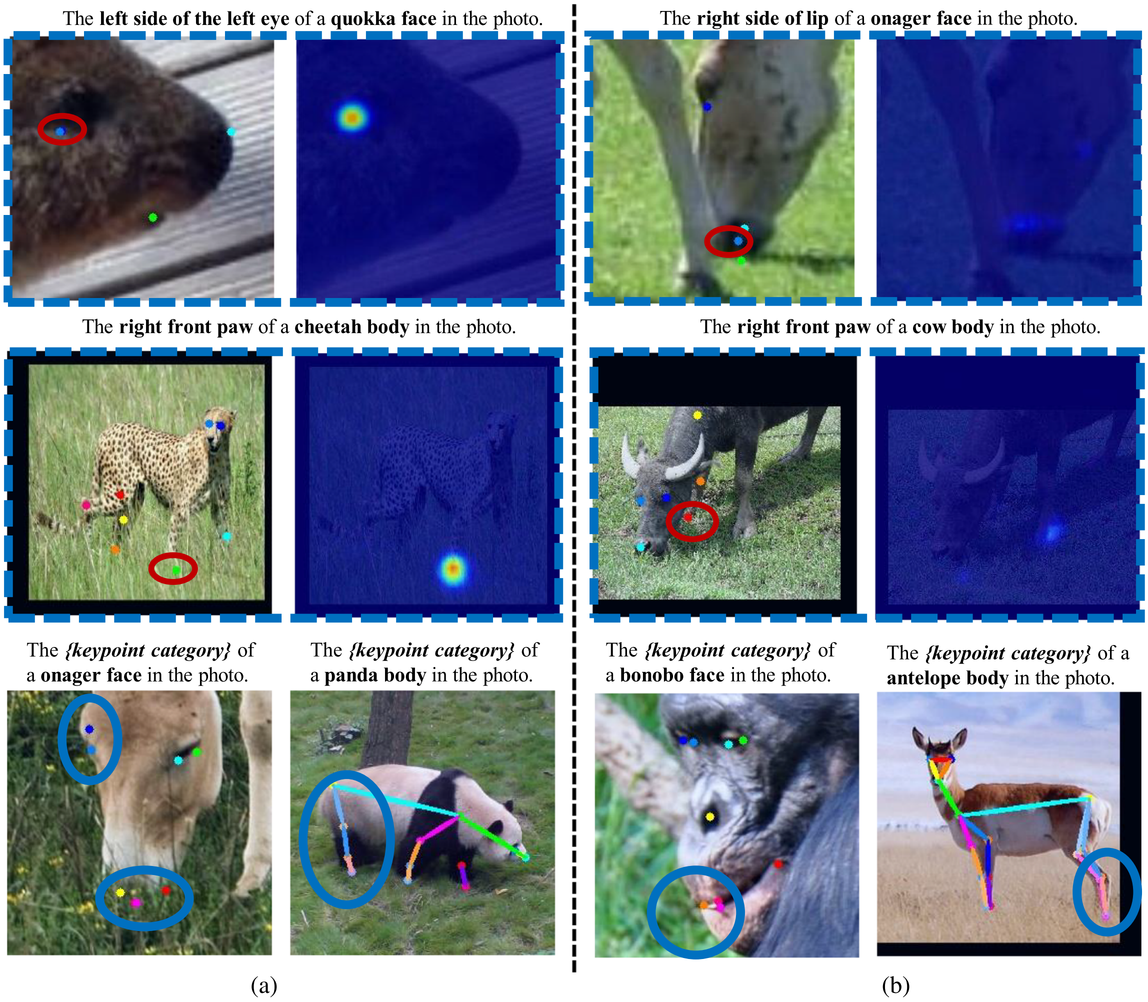

The **left side of the left eye** of a **quokka face** in the photo.
The **right side of lip** of a **onager face** in the photo.
The **right front paw** of a **cheetah body** in the photo.
The **right front paw** of a **cow body** in the photo.
The ***{keypoint category}*** of a **onager face** in the photo.
The ***{keypoint category}*** of a **panda body** in the photo.
The ***{keypoint category}*** of a **bonobo face** in the photo.
The ***{keypoint category}*** of a **antelope body** in the photo.
(a)
(b)

Fig. 7 Visual results of challenging KDSM on the test sets of two experiment settings of OVKD. **a** Demonstrates that KDSM can handle challenging scenarios involving body occlusion, environmental occlusion, and complex poses. **b** Illustrates the failure cases of KDSM in challenging keypoint detection. The points circled in red represent the ground-truth keypoint locations corresponding to the heatmaps. The blue circles enclose the challenging regions of keypoint detection. Due to space limitations, we use {*keypoint category*} to represent the keypoint categories

Table 3 Impact of hyperparameter settings on the performance (PCK0.2) of KDSM in Setting A for the OVKD task

α	β	Split1	Split2	Split3	Split4	Split5	Mean (PCK@0.2 ↑)
1	1	13.57	13.22	13.26	12.80	13.78	13.32
$1 \times 1e^{-1}$	1	42.87	31.06	32.34	14.16	31.32	30.35
$1 \times 1e^{-3}$	1	79.02	71.35	76.68	79.79	74.67	76.30
$1 \times 1e^{-4}$	1	83.99	79.96	87.50	87.32	85.86	84.93
$1 \times 1e^{-6}$	1	87.93	88.50	87.64	88.28	88.82	**88.23**
1×10^{-7}	1	87.71	89.47	87.33	86.40	89.02	87.99
$1 \times 1e^{-8}$	1	30.50	30.02	30.54	30.36	28.77	30.04
$1 \times 1e^{-10}$	1	29.28	31.38	30.61	31.48	29.69	30.49
0	1	30.02	30.63	32.64	31.31	32.17	31.35

al., 2016) within the KDSM are detailed in Table 6. Even if the extremely lightweight models such as MobileNet V2, EfficientNet-B0 and B3, KDSM still achieves reasonable performance and outperforms the OVKD baseline method (42.93 PCK@0.2 for setting A and 53.90 PCK@0.2 for setting B). We choose ResNet in our implementation in order

Table 4 Ablation study of proposed components on MP-78 for OVKD

Baseline	DDMM	VKRA	Split1	Split2	Split3	Split4	Split5	Mean (PCK@0.2 ↑)
Setting A								
✔	✖	✖	42.02	44.00	42.55	43.80	42.26	42.93
✔	✔	✖	69.64	57.86	67.95	62.10	71.92	65.89
✔	✔	✔	79.02	71.35	76.68	79.79	74.67	76.30
Setting B								
✔	✖	✖	56.06	55.36	54.35	53.07	50.66	53.90
✔	✔	✖	72.96	77.66	76.63	78.26	62.43	73.59
✔	✔	✔	84.02	87.99	83.22	83.20	80.25	83.74

Experiments are conducted on both Setting A and Setting B. PCK@0.2 is used as the metric

Table 5 Performance (PCK0.2) comparison of different attention blocks in Setting A for the OVKD task

Self_Attention	Cross_Attention	Split1	Split2	Split3	Split4	Split5	Mean (PCK@0.2 ↑)
1	3	65.43	53.78	48.80	56.90	57.97	56.58
2	3	74.89	61.87	69.55	78.56	70.39	71.05
3	3	79.02	71.35	76.68	79.79	74.67	76.30
4	3	82.49	83.15	72.00	76.66	74.33	77.73
3	1	77.04	71.16	65.19	69.18	66.65	69.84
3	2	79.44	69.07	78.38	76.49	73.75	74.43
3	4	79.62	67.00	75.69	76.41	71.71	74.09

to ensure a fair comparison with state-of-the-art few-shot keypoint detectors, i.e., POMNet and CapeFormer that use ResNet50 as vision encoder. Besides, our attempt to utilize the CLIP pre-trained Vision Transformer (Dosovitskiy et al., 2020) (ViT-B/32) falls short of expectations. This could be attributed to the fact that OVKD necessitates precise joint localization and detailed region-level feature extraction to handle diverse pose variations, which contrasts with the global image-level features captured by the CLIP visual encoder.

4.4.6 Discussion on the Choice of Text Encoder

Table 6 compares different Text_Encoders that have been pre-trained in conjunction with distinct image encoders of CLIP (Radford et al., 2021). In setting A, the Mean (PCK@0.2) scores are 55.50, 76.30, and 78.27 for the Text_Encoders pre-trained with ResNet50, ViT-B/32, and ViT-B/16 image encoders, respectively. In setting B, the Mean (PCK@0.2) scores are 79.41, 83.74, and 84.31 for the Text_Encoders pre-trained with ResNet50, ViT-B/32, and ViT-B/16 image encoders, respectively. Notably, the Text_Encoder corresponding to ViT-B/16 image encoder achieves the highest performance. The performance disparity among the Text_Encoders indicates that using a more robust Text_Encoder, particularly one pre-trained with a more powerful image encoder, leads to improved results. Although we utilize the Text_Encoder pre-trained with ViT-B/32 image encoder in this study, this finding highlights the significant potential for enhancing our method's performance by integrating a stronger Text_Encoders.

4.4.7 Discussion of OVKD Task for Different Super-Categories

To assess KDSM's capability in managing various super-categories within the OVKD task, we segregated the MP-78 dataset into two distinct, non-overlapping super-categories: Face and Body. Table 7 demonstrates KDSM's differing performance in these categories. Specifically, it achieved Mean (PCK@0.2) scores of 81.89 for the Face category and 73.97 for the Body category, clearly showing a superior performance in the Face category. The comparatively lower score for the Body category likely stems from the more complex and varied body poses. Despite the strong results, there appears to be potential for further enhancement, particularly in the Body category's performance.

4.5 Qualitative Results

In Fig. 6, we showcase the performance of KDSM in two experimental scenarios of OVKD. The top three rows depict the heatmaps for novel keypoint categories in setting A, and the bottom two rows display the actual keypoint detection outcomes in setting B. These visualizations effectively highlight KDSM's capability to adeptly navigate the OVKD task in both experimental setups.

Table 6 Performance (PCK0.2) comparison of different Text_Encoder and different Vision_Encoder configurations

Text_Encoder	Vision_Encoder (FLOPs)	Split1	Split2	Split3	Split4	Split5	Mean (PCK@0.2 ↑)
Setting A							
Ours: CLIP-B/32	**ResNet50 (5.40G)**	79.02	71.35	76.68	79.79	74.67	76.30
CLIP-Res50	ResNet50 (5.40G)	60.60	50.34	65.61	60.07	40.89	55.50
CLIP-B/16	ResNet50 (5.40G)	82.39	72.58	73.69	80.89	81.82	78.27
CLIP-B/32	MobileNet V2 (0.42G)	58.17	51.76	60.24	64.00	66.58	60.15
CLIP-B/32	EfficientNet-B0 (0.53G)	55.22	43.37	44.30	52.53	39.28	46.94
CLIP-B/32	EfficientNet-B3 (1.33G)	57.45	46.43	54.95	57.54	45.87	52.45
CLIP-B/32	ViT-B/32 (CLIP) (3.83G)	52.54	49.25	59.23	50.30	45.17	51.30
Setting B							
Ours: CLIP-B/32	**ResNet50 (5.40G)**	84.02	87.99	83.22	83.20	80.25	83.74
CLIP-Res50	ResNet50 (5.40G)	77.22	80.70	78.81	80.85	79.46	79.41
CLIP-B/16	ResNet50 (5.40G)	83.49	89.19	83.84	83.96	81.06	84.31
CLIP-B/32	MobileNet V2 (0.42G)	73.03	65.65	60.42	59.35	55.68	62.83
CLIP-B/32	EfficientNet-B0 (0.53G)	78.25	73.49	75.93	80.16	73.26	76.22
CLIP-B/32	EfficientNet-B3 (1.33G)	80.28	87.14	79.86	82.07	75.18	80.71
CLIP-B/32	ViT-B/32 (CLIP) (3.83G)	70.74	73.90	60.68	73.75	61.55	68.12

FLOPs represents the computational complexity of the Vision_Encoder. (CLIP) represents the pre-trained image encoder from CLIP (Radford et al., 2021). KDSM's default configurations are indicated in bold

Table 7 Performance (PCK@0.2) of KDSM on different super-categories in Setting A for the OVKD task

Super-Category	Split1	Split2	Split3	Split4	Split5	Mean (PCK@0.2 ↑)
Face	85.05	77.56	83.52	87.31	76.01	81.89
Body	76.73	68.61	73.67	76.37	74.49	73.97
Face w/body	79.02	71.35	76.68	79.79	74.67	76.30

5 Future Work

Firstly, our research focuses on achieving OVKD, a new and promising research topic, with satisfactory performance on regular scenes. Further improvement in challenging scenarios (e.g., occlusion, lighting, and resolution) will be left for our future work. Unlike traditional methods that rely on manual annotation, OVKD offers valuable recognition to arbitrary keypoints without prior annotation. We include some results of our method's performance in occlusion scenarios in Fig. 7a, demonstrating its capability to handle certain occlusion cases effectively. However, we also present some instances where our method encounters challenges under occlusion, as seen in Fig. 7b, indicating areas for potential improvement.

Secondly, we notice certain issues with individual predicted heatmaps in Fig. 6, such as "The left shoulder of a deer body in the photo," exhibiting the problem of "anisotropic Gaussian distribution". In future work, we can try to find appropriate methods to address the "anisotropic Gaussian" issue in the OVKD task by adjusting the loss function like LUVLi (Kumar et al., 2020) and STAR Loss (Zhou et al., 2023).

Last but not least, we plan to explore a new research direction that employs a hybrid approach utilizing both textual and visual prompts in the future. This new direction can leverage visual prompts to detect keypoints in the absence of specific semantic information. For instance, the datasets, such as WFLW (Wayne et al., 2018) (98 annotated keypoint categories) and CatFLW (Martvel et al., 2023) (48 annotated keypoint categories), are annotated with a considerable number of non-semantic keypoint categories, which will be effectively addressed through this new research direction.

6 Conclusion

We address the challenges inherent in traditional image-based keypoint detection methods for animal (including human) body and facial keypoint detection by introducing the *O*pen-*V*ocabulary *K*eypoint *D*etection (OVKD) task. This task is designed to identify keypoints in images, regardless of whether the specific animal species and keypoint category have been encountered during training. Our novel framework, Open-Vocabulary *K*eypoint *D*etection with *S*emantic-feature *M*atching (KDSM), leverages the synergy of advanced language models to effectively bridge the gap between text and visual keypoint features. KDSM integrates innovative strategies such as *D*omain *D*istribution *M*atrix *M*atching (DDMM) and other special modules, such as the *V*ision-*K*eypoint *R*elational *A*wareness (VKRA) module, leading to significant performance enhancements. Specifically, we observed a 45.30-point improvement in detecting diverse keypoint categories and a 31.43-point improvement for varied animal species compared to the baseline framework. Notably, KDSM achieves comparable results with those of state-of-the-art few-shot species class-agnostic keypoint detection methods. The proposed approach lays the groundwork for future exploration and advancements in OVKD, driving further improvements in quantitative performance metrics.

Acknowledgements This work was supported in part by the National Science Foundation of China (Grant No. 62088102), in part by the National Key R&D Program of China (NO.2022ZD0160101).

Data availibility The dataset MP-100 for this study can be downloaded at: https://github.com/luminxu/Pose-for-Everything. Our reorganized and partitioned dataset MP-78 is released together with our source code.

Declarations

Conflict of interest The authors declare that they have no Conflict of interest.

References

Andriluka, M., Pishchulin, L., Gehler, P., & Schiele, B. (2014). 2D human pose estimation: New benchmark and state of the art analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3686–3693)

Bangalath, H., Maaz, M., Khattak, M. U., Khan, S. H., & Shahbaz Khan, F. (2022). Bridging the gap between object and image-level representations for open-vocabulary detection. *Advances in Neural Information Processing Systems, 35*, 33781–33794.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., & Agarwal, S. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems, 33*, 1877–1901.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020). End-to-end object detection with transformers. In *Computer vision—ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, proceedings, part I 16* (pp. 213–229)

Chen, L., Li, J., Dong, X., Zhang, P., He, C., Wang, J., Zhao, F., & Lin, D. (2023). Sharegpt4v: Improving large multi-modal models with better captions. arXiv preprint arXiv:2311.12793.

Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition* (pp. 248–255). IEEE

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2020). Image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929

Fang, H.-S., Xie, S., Tai, Y.-W., & Lu, C. (2017). RMPE: Regional multi-person pose estimation. In *ICCV* (pp. 2334–2343)

Feighelstein, M., Shimshoni, I., Finka, L. R., Luna, S. P. L., Mills, D. S., & Zamansky, A. (2022). Automated recognition of pain in cats. *Scientific Reports, 12*(1), 9575.

Finn, C., Abbeel, P., & Levine, S. (2017). Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning* (pp. 1126–1135). PMLR

Graving, J. M., Chae, D., Naik, H., Li, L., Koger, B., Costelloe, B. R., & Couzin, I. D. (2019). Deepposekit, a software toolkit for fast and robust animal pose estimation using deep learning. *Elife, 8*, e47994.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778)

Hu, S., Zheng, C., Zhou, Z., Chen, C., & Sukthankar, G. (2023). Lamp: Leveraging language prompts for multi-person pose estimation. In *2023 IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 3759–3766). IEEE

Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., Le, H., Sung, Y.-H., Li, Z., & Duerig, T. (2021). Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning* (pp. 4904–4916). PMLR

Khan, M. H., McDonagh, J., Khan, S., Shahabuddin, M., Arora, A., Khan, F. S., Shao, L., & Tzimiropoulos, G. (2020). Animalweb: A large-scale hierarchical dataset of annotated animal faces. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 6939–6948)

Koestinger, M., Wohlhart, P., Roth, P. M., & Bischof, H. (2011). Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *2011 IEEE international conference on computer vision workshops (ICCV workshops)* (pp. 2144–2151). IEEE

Kumar, A., Marks, T. K., Mou, W., Wang, Y., Jones, M., Cherian, A., Koike-Akino, T., Liu, X., & Feng, C. (2020). Luvli face alignment: Estimating landmarks' location, uncertainty, and visibility likelihood. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 8236–8246)

Labuguen, R., Matsumoto, J., Negrete, S. B., Nishimaru, H., Nishijo, H., Takada, M., Go, Y., Inoue, K., & Shibata, T. (2021). Macaquepose: A novel "in the wild" macaque monkey pose dataset for markerless motion capture. *Frontiers in Behavioral Neuroscience,14*, 581154

Li, B., Weinberger, K. Q., Belongie, S., Koltun, V., & Ranftl, R. (2022). Language-driven semantic segmentation. arXiv preprint arXiv:2201.03546

Li, D., Li, J., & Hoi, S. (2024). Blip-diffusion: Pre-trained subject representation for controllable text-to-image generation and editing. *Advances in Neural Information Processing Systems,36*

Lin, B., Tang, Z., Ye, Y., Cui, J., Zhu, B., Jin, P., Zhang, J., Ning, M., & Yuan, L. (2024). Moe-llava: Mixture of experts for large vision-language models. arXiv preprint arXiv:2401.15947

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *Computer vision—ECCV 2014: 13th European conference, Zurich, Switzerland, September 6–12, 2014, proceedings, part V 13* (pp. 740–755)

Lu, C., & Koniusz, P. (2022). Few-shot keypoint detection with uncertainty learning for unseen species. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 19416–19426)

Martvel, G., Farhat, N., Shimshoni, I., & Zamansky, A. (2023). Catflw: Cat facial landmarks in the wild dataset. arXiv preprint arXiv:2305.04232

Nakamura, A., & Harada, T. (2019). Revisiting fine-tuning for few-shot learning. arXiv preprint arXiv:1910.00216

Newell, A., Yang, K., & Deng, J. (2016). Stacked hourglass networks for human pose estimation. In *Computer vision—ECCV 2016: 14th European conference, Amsterdam, The Netherlands, October 11–14, 2016, proceedings, part VIII 14* (pp. 483–499)

Ni, B., Peng, H., Chen, M., Zhang, S., Meng, G., Fu, J., Xiang, S., & Ling, H. (2022). Expanding language-image pretrained models for general video recognition. In *Computer vision—ECCV 2022: 17th European conference, Tel Aviv, Israel, October 23–27, 2022, proceedings, part IV* (pp. 1–18)

Pan, Y., Yao, T., Li, Y., & Mei, T. (2020). X-linear attention networks for image captioning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10971–10980)

Patel, M., Gu, Y., Carstensen, L. C., Hasselmo, M. E., & Betke, M. (2023). Animal pose tracking: 3D multimodal dataset and token-based pose optimization. *International Journal of Computer Vision, 131*(2), 514–530.

Pereira, T. D., Aldarondo, D. E., Willmore, L., Kislin, M., Wang, S.S.-H., Murthy, M., & Shaevitz, J. W. (2019). Fast animal pose estimation using deep neural networks. *Nature Methods, 16*(1), 117–125.

Pessanha, F., Salah, A. A., van Loon, T. J. P. A. M., & Veltkamp, R. C. (2023). Facial image-based automatic assessment of equine pain. *IEEE Transactions on Affective Computing, 14*(3), 2064–2076.

Pourpanah, F., Abdar, M., Luo, Y., Zhou, X., Wang, R., Lim, C. P., Wang, X. Z., & Wu, Q. J. (2022). A review of generalized zero-shot learning methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*

Qian, R., Li, Y., Xu, Z., Yang, M.-H., Belongie, S., & Cui, Y. (2022). Multimodal open-vocabulary video classification via pre-trained vision and language models. arXiv preprint arXiv:2207.07646

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., & Krueger, G., Sutskever, I. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning* (pp. 8748–8763). PMLR.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B.. (2022). High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10684–10695).

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). Mobilenetv 2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4510–4520).

Shi, M., Huang, Z., Ma, X., Hu, X., & Cao, Z. (2023). Matching is not enough: A two-stage framework for category-agnostic pose estimation. In *IEEE/CVF conference on computer vision and pattern recognition, CVPR 2023, Vancouver, BC, Canada, June 17–24, 2023* (pp. 7308–7317). IEEE

Tan, M., & Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning* (pp. 6105–6114).

Tu, J., Wu, G., & Wang, L. (2023). Dual graph networks for pose estimation in crowded scenes. *International Journal of Computer Vision,* 1–21.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems,30*.

Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Yadong, M., Tan, M., Wang, X., et al. (2020). Deep high-resolution representation learning for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 43*(10), 3349–3364.

Wang, Y., Peng, C., & Liu, Y. (2018). Mask-pose cascaded CNN for 2D hand pose estimation from single color image. *IEEE Transactions on Circuits and Systems for Video Technology, 29*(11), 3258–3268.

Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., & Perona, P. (2010). Caltech-UCSD birds 200.

Weng, T., Xiao, J., Pan, H., & Jiang, H. (2023). PartCom: Part composition learning for 3d open-set recognition. *International Journal of Computer Vision,* 1–24.

Wu, W., Qian, C., Yang, S., Wang, Q., Cai, Y., & Zhou, Q. (2018). Look at boundary: A boundary-aware face alignment algorithm. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2129–2138).

Xiao, B., Wu, H., & Wei, Y. (2018). Simple baselines for human pose estimation and tracking. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 466–481).

Xu, L., Jin, S., Zeng, W., Liu, W., Qian, C., Ouyang, W., Luo, P., & Wang, X. (2022). Pose for everything: Towards category-agnostic pose estimation. In *Computer vision—ECCV 2022: 17th European conference, Tel Aviv, Israel, October 23–27, 2022, proceedings, part VI* (pp. 398–416)

Xu, M., Zhang, Z., Wei, F., Hu, H., & Bai, X. (2023). Side adapter network for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 2945–2954).

Xu, Y., Zhang, J., Zhang, Q., & Tao, D. (2024). Vitpose++: Vision transformer for generic body pose estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 46*(2), 1212–1230.

Yao, L., Han, J., Wen, Y., Liang, X., Xu, D., Zhang, W., Li, Z., Xu, C., & Xu, H. (2022). Detclip: Dictionary-enriched visual-concept paralleled pre-training for open-world detection. arXiv preprint arXiv:2209.09407

Yu, H., Xu, Y., Zhang, J., Zhao, W., Guan, Z., & Tao, D. (2021). Ap-10k: A benchmark for animal pose estimation in the wild. arXiv preprint arXiv:2108.12617

Zhang, H., Lai, S., Wang, Y., Da, Z., Dun, Y.,& Qian, X. (2023). Scgnet: Shifting and cascaded group network. *IEEE Transactions on Circuits and Systems for Video Technology*

Zhang, H., Dun, Y., Pei, Y., Lai, S., Liu, C., Zhang, K., & Qian, X. (2024). HF-HRNet: A simple hardware friendly high-resolution network. *IEEE Transactions on Circuits and Systems for Video Technology*. https://doi.org/10.1109/TCSVT.2024.3377365

Zhang, H., Shao, W., Liu, H., Ma, Y., Luo, P., Qiao, Y., & Zhang, K. (2024b). AVIbench: Towards evaluating the robustness of large vision-language model on adversarial visual-instructions. arXiv preprint arXiv:2403.09346

Zhou, Z., Li, H., Liu, H., Wang, N., Yu, G., & Ji, R. (2023). Star loss: Reducing semantic ambiguity in facial landmark detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 15475–15484).

Zhu, X., Zhang, R., He, B., Guo, Z., Zeng, Z., Qin, Z., Zhang, S., & Gao, P. (2023). Pointclip v2: Prompting clip and GPT for powerful 3d open-world learning. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 2639–2650).

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.