ViT-P3DE*: Vision Transformer Based Multi-Camera Instance Association with Pseudo 3D Position Embeddings

Minseok Seo, **Hyuk-Jae Lee** and **Xuan Truong Nguyen**
Inter-University Semiconductor Research Center (ISRC)
Department of Electrical and Computer Engineering, Seoul National University
{sms0121, hjlee, truongnx}@capp.snu.ac.kr

Abstract

Multi-camera instance association, which identifies identical objects among multiple objects in multi-view images, is challenging due to several harsh constraints. To tackle this problem, most studies have employed CNNs as feature extractors but often fail under such harsh constraints. Inspired by Vision Transformer (ViT), we first develop a pure ViT-based framework for robust feature extraction through self-attention and residual connection. We then propose two novel methods to achieve robust feature learning. First, we introduce learnable pseudo 3D position embeddings (P3DEs) that represent the 3D location of an object in the world coordinate system, which is independent of the harsh constraints. To generate P3DEs, we encode the camera ID and the object's 2D position in the image using embedding tables. We then build a framework that trains P3DEs to represent an object's 3D position in a weakly supervised manner. Second, we also utilize joint patch generation (JPG). During patch generation, JPG considers an object and its surroundings as a single input patch to reinforce the relationship information between two features. Ultimately, experimental results demonstrate that both ViT-P3DE and ViT-P3DE with JPG achieve state-of-the-art performance and significantly outperform existing works, especially when dealing with extremely harsh constraints.

1 Introduction

Multi-camera instance association (MCIA) aims to identify the identical objects among objects in images of the same scene captured by different cameras. For example, given two images of the same scene caught by different cameras, as shown in Fig. 1, MCIA generates object patches by cropping images according to the bounding boxes of each object. MCIA then compares object patches in one image with those in the other image to associate the identical objects between images. While MCIA is being used in real-world applications such as automatic check-out systems in supermarkets, it is accompanied by several problems, such as major differences in the appearance of the same objects, severe object occlusion, and the presence of multiple objects that look similar or identical in a scene, as shown in Fig. 1.

Figure 1: Constraints of MCIA that presented in MessyTable. (Red box) Presence of multiple similar-or-identical looking objects in a scene. (Green box) Major appearance differences among identical objects due to view variations. (Blue box) Severe object occlusion caused by the high density of the scene.

To tackle these problems of MCIA, many approaches based on CNNs have been studied [Simo-Serra *et al.*, 2015; Han *et al.*, 2015; Zagoruyko and Komodakis, 2015; Cai *et al.*, 2020]. These methods leverage CNNs to extract robust features, which are the key to resolving the problems of MCIA. However, CNN-based methods often struggle to associate the target object with a complex background composed of multiple objects. CNNs might extract features from the non-target object because CNNs tend to focus on the most discriminative local feature with fixed receptive fields of convolution. As shown in Fig. 2-(b), CNNs mainly extract the features of the snack instead of the banana in the case of III and IV.

Recently, Vision Transformer (ViT) has yielded comparable performance with CNN-based methods in computer vision tasks [Dosovitskiy *et al.*, 2021; Liu *et al.*, 2021; Yun *et al.*, 2022; He *et al.*, 2022]. The self-attention and residual connection make ViT a promising solution for MCIA. (1) ViTs emphasize the target object in the complex background and extract the global information of the target object using the dynamic receptive field of self-attention. (2) ViTs also

globally extract the features of non-target objects and build the relationship between non-target objects and the target object. This relationship is organized around the target object by strong residual connections [Raghu *et al.*, 2021]. As shown in Fig. 2-(c), Gradient-weighted Class Activation Mappings (Grad-CAM) [Selvaraju *et al.*, 2017] of ViTs exhibit similarities to the segmentation results due to ViTs' capability to globally recognize multiple objects, including the target object. Inspired by these abilities of ViT, we develop a pure ViT-based baseline MCIA framework.

However, features extracted from ViT alone are insufficient to handle cases of multiple similar-looking objects with a clean background because ViT cannot utilize relationship information and may fail to differentiate the objects with only visual information. To solve this issue, we propose two methods to enhance the robustness of the ViT-baseline framework. First, we propose Psuedo 3D position Embeddings (P3DEs) based on the following observations: a) utilizing non-visual cues enhances MCIA performance [Cai *et al.*, 2020] and b) ViT learns desired characteristics through certain auxiliary embeddings [Naseer *et al.*, 2021]. P3DEs represent the approximate location of an object in the world coordinate system (WCS), which is independent of the constraints of MCIA. We generate these embeddings from the camera ID and the position of the object in the image. We then model the relationship between the object's 2D position in the image and the object's 3D position in the WCS in a weakly supervised manner by jointly training P3DEs with existing input embeddings. Through encoders in ViT, P3DEs are trained to represent the object's 3D position. By using P3DEs, our method yields significant performance improvement.

Second, we utilize Joint Patch Generation (JPG), simple yet effective patch generation method, with which ViT uses an object and its surroundings as a single input patch. This trick allows a ViT-baseline framework to learn in-depth relationship between the two types of information by enlarging receptive fields. Moreover, we avoid the computation overhead resulting from the expanded object patch by resizing. Ultimately, JPG further enhances the performance of the ViT-baseline framework. We evaluate the performance of our methods on MessyTable, which presents the harshest constraints among MCIA datasets. At the same time, we estimate the robustness of these methods in terms of each of those constraints. Our contributions are summarized as follows:

1. *ViT-based baseline framework for MCIA*: We develop a pure ViT-baseline MCIA framework that respectively enhances AP and IPAA-100 by 14.8% and 4.8% compared to a SOTA CNN-based framework.
2. *Pseudo 3D Position Embeddings*: We propose auxiliary embeddings whose features are independent of the constraints of MCIA. By integrating these embeddings into the ViT-baseline framework, the performance improves AP by 7.1% and IPAA-100 by 11.1%.
3. *Joint Patch generation*: We build effective patch generation method for ViT to reinforce relationship information while avoiding computation overhead. Experimental results show that this trick further improves AP and IPAA-100 by 5.0% and 5.4%, respectively.

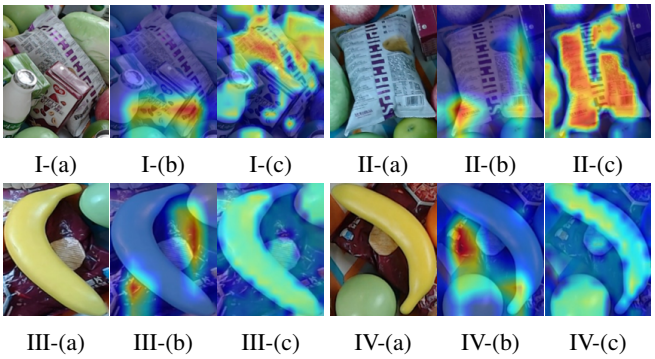

Figure 2: Grad-CAM comparison between CNN-based and ViT-based methods on MCIA: (a) Original images, (b) CNN-based methods, (c): ViT-based methods. The target object of I, II is the snack and that of III, IV is the banana.

2 Background and Related Works

2.1 Multi-Camera Instance Association

Extracting robust features is the key to solving the constraints of MCIA. Instead of using hand-crafted features, such as SIFT [Luo *et al.*, 2021], generating features with CNN networks has recently become more favorable. Many works propose Siamese-type CNN models [Koch *et al.*, 2015] to estimate the similarity between the positive and negative patch pairs. DeepDesec [Simo-Serra *et al.*, 2015] extracts discriminant representations from a Siamese network and measures the L2 distance between vectors of patches. MatchNet [Han *et al.*, 2015] proposes a cascaded Siamese network and metric networks in place of the L2 distance. DeepCompare [Zagoruyko and Komodakis, 2015] learns the similarity function from raw image pixels and demonstrates the effectiveness of multi-resolution patches in MCIA.

TripletNet [Cai *et al.*, 2020] initially uses the triplet network architecture [Schroff *et al.*, 2015] with a CNN feature extractor and utilizes the L2 distance. By exploiting the triplet network architecture, TripletNet yields better performance than any other Siamese network based framework. To address the limitation of TripletNet in distinguishing between objects with similar appearances, ASNet [Cai *et al.*, 2020] utilizes surrounding information through the implementation of separate patch generation (SPG). This method involves the creation of two patches - one for the appearance of the target object and the other for only surrounding objects. Two TripletNets are trained within the ASNet framework, each utilizing one of these patches. By considering the features of neighboring objects, ASNet achieves state-of-the-art results.

2.2 Challenges in MCIA

Challenging Constraints

In contrast to single-camera datasets, MCIA datasets, such as MPII Multi-Kinect (MPII MK) [Susanto *et al.*, 2012], EPFL Multi-View Multi-Class (EPFL MVMC) [Roig *et al.*, 2011], WILDTRACK [Chavdarova *et al.*, 2018], and MessyTable [Cai *et al.*, 2020], introduce new kinds of constraints. In particular, MessyTable reproduces a real-world setting to present many practical and challenging constraints that lead to object

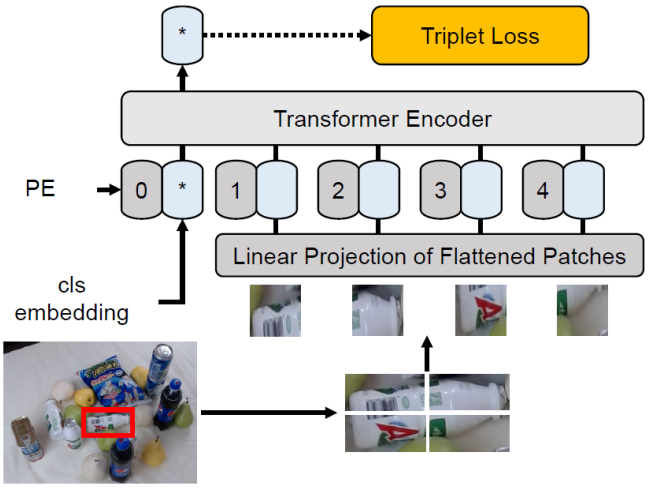

Figure 3: ViT-based baseline MCIA framework

misidentification, as shown in Fig. 1: a) Presence of multiple similar-or-identical looking objects in a scene, b) major appearance differences among identical objects due to dynamic view variations, and c) heavy occlusion in object by the high density of a scene. Due to these constraints, existing works often fail to associate the identical objects.

Non-visual Information

To solve the difficulties caused by the aforementioned constraints, many works utilize non-visual cues such as geometric information with deep learning networks. Since determining the accurate 3D position of an object from positions in multi-view images is error-prone due to partially visible instances, a previous work has proposed alternative methods. Epipolar soft constraint (ESC) [Cai *et al.*, 2020], a geometric cue taken from epipolar geometry, was proposed to improve performance. However, ESC needs all camera parameters and is not effective when the objects of comparison are in close proximity to one another. Meanwhile, TransReID [He *et al.*, 2021] proposes side information embedding (SIE), which uses the camera ID and viewpoint to learn robust features. Unfortunately, if two objects in the same image (e.g. identical camera ID) are visually similar or identical, SIE fails to associate them with objects in other images. Furthermore, several studies [Chen *et al.*, 2022; Wang *et al.*, 2022] of person re-identification propose pose information considering various human poses resulting from different joint configurations. However, this information has limited usefulness for the instance association. Instances possess a fixed structure, and as a result, they cannot be distinguished based on structural information.

3 Methodology

3.1 ViT-based Baseline MCIA Framework

Inspired by the effectiveness of the triplet network and the features of ViT in the MCIA framework, we build a ViT-based baseline framework for MCIA, as presented in Fig. 3. Given an image and a bounding box of the object, we crop the image corresponding to the bounding box to generate the object patch and resize it to a fixed height and width. Then, the resized patch is divided into N patches and each patch acts as one token embedding, as shown in Fig. 3. The class (cls) embedding (e_{cls}) is concatenated before the embedding of patches (E_{patch}). The position embedding (PE;E_{pos}) is added to the cls embedding and patch embeddings according to the order. Ultimately, the input embedding of the transformer (Z_{in}) is described as follows:

$$Z_{in} = [e_{cls}; E_{patch}] + E_{pos} \quad (1)$$

To exploit global representations, we use cls embedding, which encodes global features [He *et al.*, 2021], when training ViT with the triplet loss. Given the anchor, positive and negative patches, the triplet loss in the ViT-baseline framework is expressed as:

$$L_b = max(\|f_{a,cls} - f_{p,cls}\|_2 - \|f_{a,cls} - f_{n,cls}\|_2 + \alpha, 0) \quad (2)$$

where α is a margin between positive and negative pairs and f_{cls} is the cls embedding in the output of the transformer.

3.2 Pseudo 3D Position Embedding

Unfortunately, features extracted from f_{cls} are insufficient to solve the problem of scenes that present extremely harsh constraints because the features are somewhat dependent on the constraints. For example, ViT-baseline easily fails to associate objects whose appearances are similar to each other with a clean background since the visual information and relationship information yielded by ViT become ineffective. To solve this problem, we propose learnable pseudo 3D position embeddings (P3DEs) that represent features, which are unique to each object and independent of harsh constraints.

Implementation

P3DEs consist of three embeddings, $P3DE_x, P3DE_y$, and $P3DE_z$, that represent the x, y, and z-coordinates of the object's 3D position in the WCS. As shown in Fig. 4, we construct learnable embedding tables T_x, T_y, and T_z to generate P3DEs. To encode the depth of an object, we construct a camera embedding table T_z that uses the camera ID as a key. We also build two location embedding tables T_x and T_y that use the camera ID and the middle point of the object's bounding box in the image as keys. The size of each embedding table is expressed by Eq. 3, where N is the total number of cameras, D the embedding dimension size, H' the scaled height, and W' the scaled width. Conceptually, the numbers of indexes of T_x and T_y are determined by $N * W$ and $N * H$, respectively. We reduce the size of the indexes of T_x and T_y with a small grid ratio G, e.g., $0 < G < 1$, to train all elements in those tables several times. We present the scaling formula in Eq. 4. Given embedding tables T_x, T_y, and T_z, where (x', y') and cam_id respectively denote the midpoint of the object's bounding box and the camera ID, each embedding of those tables is represented according to Eq. 5.

$$T_x \in \mathbb{R}^{N*W'*D}, T_y \in \mathbb{R}^{N*H'*D}, T_z \in \mathbb{R}^{N*D} \quad (3)$$

$$H' = \lfloor H * G \rfloor, W' = \lfloor W * G \rfloor \quad (4)$$

$$\begin{aligned} P3DE_x &= T_x[cam\_id * W' + \lfloor x' * G \rfloor] \\ P3DE_y &= T_y[cam\_id * H' + \lfloor y' * G \rfloor] \\ P3DE_z &= T_z[cam\_id] \end{aligned} \quad (5)$$

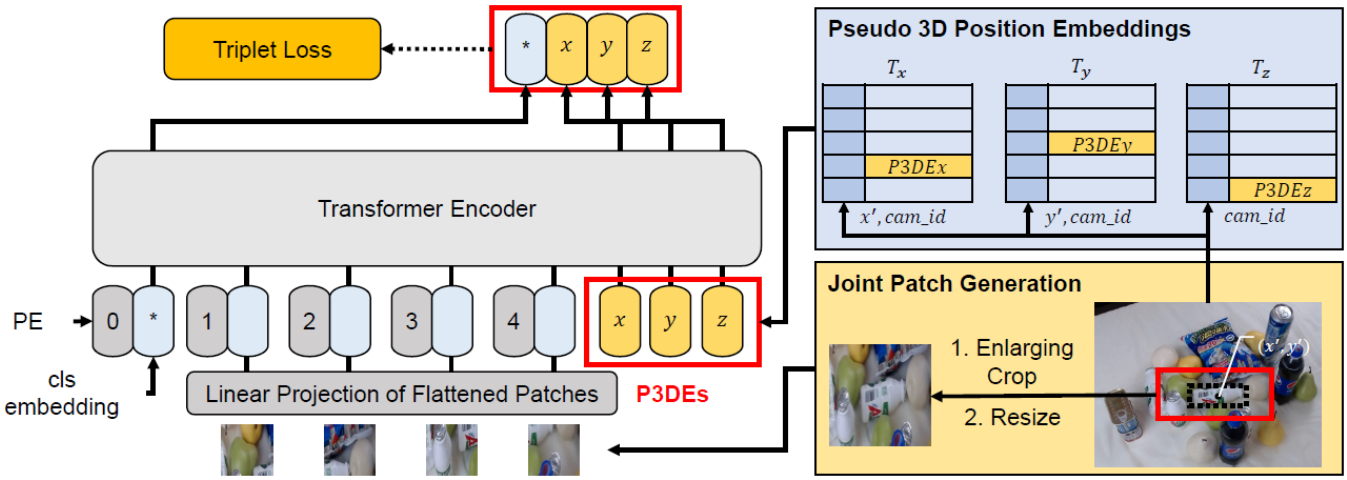

Figure 4: Architecture of ViT-baseline with pseudo 3D position embeddings (P3DEs) and joint patch generation (JPG) (ViT-P3DE*). P3DEs are generated from each embedding table using the camera ID of the image and the middle point of the bounding box of the object as keys. These embeddings are jointly trained with existing input embeddings and are added to cls embedding before computing loss. JPG generates a single patch encompassing the target object and surrounding objects by enlarging the crop. It minimizes computational cost through the resizing of the patch.

Remarks on P3DE

To compute an object's 3D position in the WCS, the accurate camera parameters and multiple positions of the objects from different viewpoints are necessary. However, measuring camera parameters requires strong assumptions and these parameters vary according to the camera settings [Bogdan *et al.*, 2018]. Therefore, we use the camera ID and location of the object, which are usually available in MCIA, when generating P3DEs. Through this point, P3DEs deal with the various camera settings without the information on camera parameters.

Weakly Supervised 3D Localization

As shown in Fig. 4, we integrate P3DEs into the ViT-baseline to model the relationship between an object's 2D position in the image and the 3D position of an object in the WCS. We concatenate the P3DEs after the input embeddings to jointly train P3DEs and input embeddings. Moreover, we explicitly add f_{P3DEs} to a f_{cls} with a regularization parameter λ, where f_{P3DEs} denotes the P3DEs in the outputs of the transformer. Accordingly, the final feature (f_t) and triplet loss function (L_{p3de}) of ViT-baseline with P3DE are formulated as Eq. 6 and 7, respectively. Through these processes, our framework learns the localization of an object's 3D location in the WCS in a weakly supervised manner, which is a learning scheme that utilizes only coarse-grained labels [Zhou, 2018], in addition to learning MCIA in a supervised manner. We train our framework in a weakly supervised manner for the 3D position because MCIA usually provides only information on whether objects are identical (coarse-grained labels) and does not offer any information on the 3D location of objects (fine-grained labels). As it is hard to learn accurate information with weak supervision, P3DEs represent the pseudo 3D position of the object.

$$f_t = f_{cls} + \lambda * (f_{P3DE_x} + f_{P3DE_y} + f_{P3DE_z})/3 \quad (6)$$

$$L_{p3de} = max(\|f_{a,t} - f_{p,t}\|_2 - \|f_{a,t} - f_{n,t}\|_2 + \alpha, 0) \quad (7)$$

3.3 Joint Patch Generation

As in the case of SPG in ASNet, utilizing information about surrounding objects is beneficial for differentiating between objects with similar appearances. However, the SPG method is not effective when used in conjunction with the ViT, as it removes the global relationship information among objects presented in the patch by masking the target object when generating the patch for surrounding objects.

In this paper, we utilize joint patch generation (JPG), with which ViT effectively considers the surrounding information of an object. Transformer achieves great performance improvement in the NLP field by learning the relationships between tokens [Vaswani *et al.*, 2017]. Paying attention to this observation, we exploit JPG that allows ViT to learn not only the global features of an object's appearance and its surrounding information but also the in-depth relationship between the two types of information. As shown in Fig. 4, JPG generates an object patch by cropping an image with a bounding box that has been enlarged by a square of zoom-out ratio (ZO^2) in order to include the surrounding information. To mitigate the computational overhead caused by the expanded patch, JPG employs resizing. The fixed height and width used in this resizing process are equivalent to those of the ViT-baseline. Through these processes, JPG achieves a considerable performance improvement while preserving computational efficiency. Moreover, JPG avoids any model parameters overhead by using just one patch, in constrast to SPG, which always incurs such overhead by using two patches and doubling the number of feature extractors.

	MPII MK	EPFL MVMC	WILDTRACK	MessyTable
Cameras	4	6	7	9
Settings	2	1	1	567
Classes	9	3	-	120
Scenes	33	240	400	5,579
Instances	6∼10	5∼9	13∼40	6∼73

Table 1: Configurations of MCIA datasets

4 Experiments

4.1 Experimental Settings

Dataset

We summarize configurations of MCIA datasets in Table 1. Considering that WILDTRACK is the dataset for multi-camera people detection and tracking, we do not consider the number of classes in this case. We use MessyTable as our target dataset among the MCIA datasets because this dataset includes the harshest constraints. First, this dataset presents major appearance differences among identical objects by utilizing 567 different camera settings with nine cameras while other datasets use only one or two camera settings. Secondly, MessyTable includes many seriously occluded objects by locating the large number of objects, up to 73, on the table where the space is limited. Lastly, with 120 classes of objects that are classified into 42 groups in terms of appearance similarity, most of the scenes in this dataset include multiple objects that appear similar or identical. For these reasons, we evaluate the effectiveness of our methods on MessyTable.

Evaluation Metric

We evaluate our framework with the following evaluation metrics: class-agnostic average precision (AP), a false positive rate at 95% recall (FPR-95), and image pair association accuracy (IPAA). IPAA, newly introduced by Cai et al. (2020), evaluates the association results at the image-pair level, while AP and FPR-95 estimate the result at the object-pair level. IPAA-X means the percentage of image pairs, which satisfy the condition that at least X% of the object pairs in images are associated correctly. IPAA is usually a stricter evaluation metric than AP and FPR-95. We mainly note IPAA-100 among IPAA-X in this paper.

Implementation

We use the latest MCIA frameworks, TripletNet and ASNet, on MessyTable as baselines. Given scenes captured by nine cameras, we conduct experiments using all 72 camera pairs for the training data to measure peak performance and prevent performance variation due to camera pair sampling. We observe that the AP of ASNet can vary by more than 6% depending on the sampled eight camera pairs. To ensure a fair comparison, we reproduce the results of TripletNet and ASNet under our experimental setting based on their publicly available code [Cai *et al.*, 2020] and present both our reproduced results and the originally reported results [Cai *et al.*, 2020] trained with eight camera pairs (marked with an $\star$) in Table 2. It is observed that training with fewer camera pairs may not achieve peak performance (i.e., a 3.3 AP decrease for ASNet) compared to the all-pairs setting.

Framework	Backbone	Param.	AP ↑	FPR-95 ↓	IPAA-100 ↑
TripletNet [Cai *et al.*, 2020]	ResNet-18$\star$	11M	46.7	20.6	16.8
	ResNet-18	11M	53.4	15.3	19.9
	ResNet-50	24M	56.1	12.0	23.9
	ResNet-101	43M	57.5	11.0	24.9
ASNet [Cai *et al.*, 2020]	ResNet-18$\star$	22M	52.4	20.9	17.0
	ResNet-18	22M	55.7	14.4	23.1
	ResNet-50	47M	56.7	14.9	22.7
	ResNet-101	85M	59.1	12.7	25.6
ViT-baseline	ViT-T	6M	70.5	7.9	27.9
	ViT-S	22M	73.3	6.8	29.7
	ViT-B	87M	74.7	6.3	32.1
	DeiT-T	6M	71.7	7.3	28.8
	DeiT-S	22M	73.8	6.5	29.2
	DeiT-B	87M	75.2	6.1	32.7
ViT-P3DE*	ViT-T	6M	82.8	4.9	43.5
	ViT-S	22M	85.1	4.2	46.2
	ViT-B	87M	86.6	3.6	48.3
	DeiT-T	6M	82.6	5.0	43.4
	DeiT-S	23M	85.3	4.0	47.1
	DeiT-B	87M	87.4	3.4	49.2
ASNet+ESC [Cai *et al.*, 2020]	ResNet-18	22M	60.2	11.8	26.8
	ResNet-50	47M	60.4	12.5	24.4
	ResNet-101	85M	62.8	10.7	27.7
ViT-P3DE*+ESC	ViT-T	6M	87.8	3.0	46.7
	ViT-S	22M	88.6	2.7	47.8
	ViT-B	87M	89.5	2.4	48.1

Table 2: Performances of the SOTA frameworks and our proposed frameworks. $\star$: Results with a previous work configuration that uses eight camera pairs in training.

We train the frameworks with one batch size that contains 64 triplet pairs on a single A100 GPU. Unless otherwise noted, we resize an object patch to 224$*$224 with a bilinear interpolation. We utilize the Adam optimizer [Kingma and Ba, 2015] with an initialized learning rate of 1e-4. The weights of the frameworks are initialized with the pretrained weights on ImageNet 1K [Deng *et al.*, 2009]. We build other settings, such as the triplet loss margin α, to be identical to those of TripletNet and ASNet. We set the grid ratio G as one over sixty, the regularization parameter λ as 0.1, and the zoom-out ratio ZO as 2 experimentally. Ablation studies of these parameters are given in the Appendix.

4.2 Results of the Proposed Method

We evaluate our proposed frameworks, ViT-baseline and ViT-baseline with P3DEs and JPG (ViT-P3DE*), and state-of-the-art CNN-based frameworks, TripletNet and ASNet, on MCIA. We measure the performance of TripletNet and ASNet using representative CNN backbones, ResNet families [He *et al.*, 2016]. We use the ViT [Dosovitskiy *et al.*, 2021] and DeiT [Touvron *et al.*, 2021] families as the backbone for the ViT-baseline and ViT-P3DE*. Note that ViT-Tiny (ViT-T) and ViT-Small (ViT-S) are DeiT-Tiny (DeiT-T) and DeiT-Small (DeiT-S) without a distillation embedding, respectively.

As shown in Table 2, although the number of model parameters of the ViT-baseline with the ViT-T backbone is far smaller than that of the TripletNet and ASNet with all CNN backbones, the performance of the ViT-baseline is signifi-

	Method	AP ↑	FPR-95 ↓	IPAA-100 ↑
ViT-Baseline	Base.	70.5	7.9	27.9
Additional Embedding	Base.+3E	70.6	7.7	28.3
	Base.+SIE	71.2	7.6	28.6
	DeiT	71.7	7.3	28.8
	ViT-P3DE	77.6	5.6	39.0
Patch Generation	Base.+SPG	70.1	8.9	26.4
	Base.+JPG	75.5	7.9	33.3

Table 3: Ablation study of ViT-P3DE*

cantly better than that of the TripletNet and ASNet with all CNN backbones. In particular, our ViT-baseline with ViT-T improves 11.4% AP, 4.8% FPR-95, and 2.3% IPAA-100 compared to ASNet with ResNet-101. These results demonstrate the significance of globally recognizing objects in MCIA, in contrast to only extracting local features of an object.

By utilizing our proposed methods, P3DEs and JPG, on the ViT-baseline with ViT-T backbone, ViT-P3DE* attains a 12.3% AP, 3% FPR-95, and 15.6% IPAA-100 improvement and achieves the state-of-the-art performance. The enhancement of FPR-95 is relatively modest compared to the other metrics because the number of negative samples greatly surpasses that of positive samples in the MessyTable. Although IPAA-100 is a stricter evaluation metric than AP, ViT-P3DE* yields a larger improvement in IPAA-100 than AP when compared to the ViT-baseline. This is because ViT-baseline struggles to associate instance-pairs that are affected by extremely harsh constraints. On the other hand, ViT-P3DE* addresses these problems by utilizing inherent features of the target object and the in-depth relationship information between objects. As in the case of the ASNet, ViT-P3DE* also has an ability to exploit benefits of ESC [Cai *et al.*, 2020] if all camera parameters are provided.

4.3 Ablation Study of ViT-P3DE*

We evaluate the performance gains of our methods on the ViT-baseline with ViT-T backbone and compare ours with the methods of previous works.

Additional Embedding

We compare P3DEs with additional three embeddings (Base.+3E), distillation embedding (DeiT) [Touvron *et al.*, 2021], and side information embedding (SIE) [He *et al.*, 2021]. Additional three embeddings in Base.+3E are independently trained without encoding information and SIE only uses the camera ID since the MessyTable has no annotated viewpoint. As shown in Table 3, P3DEs achieve 7.1% AP, 2.3% FPR-95, and 11.1% IPAA-100 improvement while other methods attain less than 1.5% enhancement for all metrics. These results demonstrate the importance of encoding information in the embeddings.

Patch Generation

We compare JPG with SPG [Cai *et al.*, 2020]. As presented in Table 3, using SPG drops the performance. This is because the masking part in the patch of surrounding objects leads to a larger L2 distance between surrounding features compared to the ViT-baseline. On the other hand, JPG reduces the L2 distance by reinforcing the relationship information and achieves 5.0% AP and 5.4% IPAA-100 improvement.

AP	ASNet	ViT	ViT+JPG	ViT+P3DE	ViT-P3DE*
0°∼15°	81.5	91.2	92.2	93.4	94.6

Table 4: APs of instances-pairs with 0°∼15° angle difference

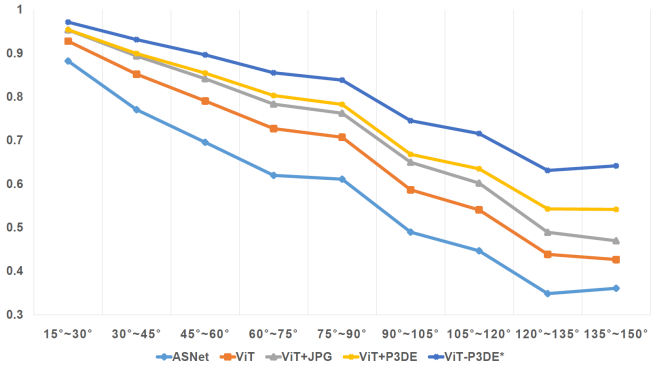

Figure 5: Normalized APs of instance-pairs with each angle difference

4.4 Robustness of ViT-baseline and ViT-P3DE*

We evaluate the ASNet and our works in terms of each of the three constraints: appearance differences among identical objects, object occlusion, and the presence of similar objects in one scene. Because the effectiveness of the frameworks in addressing these constraints is similar when the architecture is identical, we show the results of the frameworks with the ResNet-18 and ViT-T backbones. ASNet denotes ASNet with the ResNet-18 backbone and ViT refers to the ViT-baseline with the ViT-T backbone. To evaluate the robustness of the frameworks with respect to a specific constraint, we compare the results of instance-pairs impacted by different degrees of that constraint. We first classify instance-pairs into groups according to the severity of the constraint. We then normalize the results of each group by the base group with the best result to empirically observe performance change trends between individual groups and the base group. While there may be cases where constraints affect one another, our approach provides valuable insights into performance variations across diverse subsets. We mainly present the normalized APs (normAP) in this section.

Appearance Differences among Identical Objects

To observe the frameworks' robustness to this constraint, we classify instance-pairs into ten groups according to their angle differences. In MessyTable, the angle variation of each camera is up to 150°. We divide this range into ten groups, each group covering 15° angle differences, as shown in Fig. 5. We normalize the AP of each group with the AP of instance-pairs that are of angle differences from 0° to 15°. As shown in Fig. 5, the normAP of ViT-baseline is on average 8.6% better than that of ASNet in all angle differences. Moreover, the

AP	ASNet	ViT	ViT+JPG	ViT+P3DE	ViT-P3DE*
SPS	82.4	92.7	92.0	93.8	94.1
SPD	53.7	68.6	73.9	76.2	81.7
SBD	51.6	66.2	72.1	74.5	80.4
SPD/SPS	0.652	0.740	0.803	0.812	0.868
SBD/SPS	0.626	0.714	0.784	0.794	0.854

Table 5: APs of SPS, SPD, and SBD, along with normalized APs

performance degradation of ViT-baseline due to increasing viewpoint variation is less than that of ASNet, as the features from ViT are robust to the scene bias [Choi *et al.*, 2019].

We find that P3DEs successfully represent the features, which are independent of viewpoint variations. P3DEs improve the normAP in all angle differences. Moreover, as the angle difference increases, the gap between the graph of ViT-baseline and that of ViT-baseline with P3DEs becomes larger, from 2.6% at 15°∼30° to 11.5% at 135°∼150°. This is because ViT-baseline with P3DEs recognizes the contextual cues of the input and accordingly uses more features of P3DEs through attention mechanism. Also, JPG makes the ViT-baseline more robust by reinforcing the relationship between an object and its surroundings. Through these methods, ViT-P3DE* achieves significant enhancement in the robustness to appearance differences among identical object. The normAP of ViT-P3DE* is from 4.4% to 21.5% better than that of ViT-baseline. Moreover, as angle differences increase, ViT-P3DE* improves the performance more. From this point, we believe that our method works well in a real-world environment, where constraints are usually extremely harsh.

Presence of Similar Objects in One Scene

We classify instance-pairs into two types: superclass-duplication (SPD) and superclass-single (SPS). Note that, objects labeled as the same superclass appear similar or identical to each other. Among the instance-pairs between two images, an instance-pair that includes object A in one image is classified as SPD if there are more than one object with the same superclass as A in the other image. Conversely, if there is only one or no object with the same superclass as A in the other image, the instance-pair that contains object A is classified as SPS. We normalize the AP of SPD with that of SPS. As shown in Table 5, the normAP of ASNet is lower than that of ViT-baseline even with surrounding information. This is because ASNet often focuses on the non-target object. ViT-baseline resolves this problem by considering all objects within the patch.

P3DEs are significantly effective on this constraint because it represents a unique feature of an object. ViT-baseline with P3DE achieves a normAP improvement by 7.2% compared to the ViT-baseline. Meanwhile, JPG achieves 6.3% better normAP than the ViT-baseline. It is worth noting that the significant improvement in the AP of SPD through using JPG contributes to the better normAP. This highlights the vital role that the relationship information between an object and its surroundings plays in the SPD instance pairs. With these effective methods, ViT-P3DE* yields 12.8% improvement in normAP over the ViT-baseline.

AP	ASNet	ViT	ViT+JPG	ViT+P3DE	ViT-P3DE*
SPS in *easy*	89.4	96.6	95.8	97.0	97.0

Table 6: APs of SPS in *easy* scene

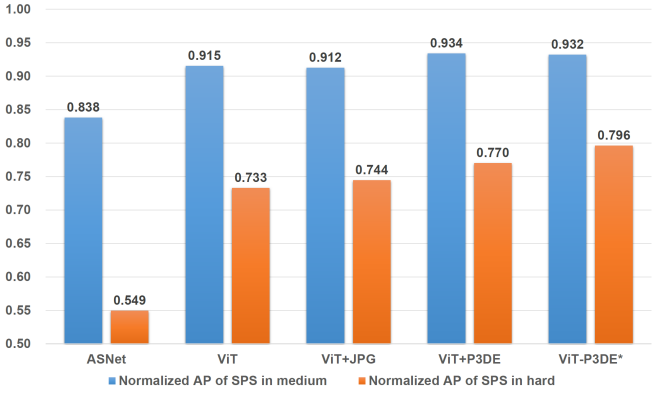

Figure 6: Normalized APs of SPS in *medium* and *hard* scenes

Presence of Identical Objects in One Scene

Among SPD, we call instance-pairs that include the object, which is labeled as the same subclass as that of more than one objects in the image of other camera, as subclass-duplication (SBD). Objects labeled as the identical superclass look similar or identical to each other while objects tagged as the same subclass appear identical to each other. In other words, it is more difficult to identify identical objects among SBD compared to SPD. We examine the proposed frameworks' robustness to the constraint, which is the presence of identical objects in one scene. To measure this, we normalize the AP of SBD with that of SPS. We present the results in Table 5.

While ViT-baseline and ASNet's normAP differences between SPD and SBD are 2.6% AP, that of ViT-P3DE* is 1.4% AP. This is because ViT-P3DE* utilizes features that resolve the misidentification among an accurate object and different objects that appear identical. The results of ViT+JPG and ViT+P3DE demonstrate that the features captured by JPG and P3DEs are robust to this constraint. Ultimately, our framework achieves the best robustness to scenes that include multiple objects that look identical.

Object Occlusion

To examine the effects of object occlusion, we utilize scenes in MessyTable that have been classified into *easy, medium,* and *hard* in terms of the degree of occlusion and the number of objects that appear similar or identical. A scene that features extreme occlusion and a large number of similar instances is classified as *hard*, while less complicated scenes are classified as *medium* or *easy*. To consider the effect of occlusion on the frameworks' performances, we evaluate only the objects that correspond to SPS in *easy, medium,* and *hard* scenes, respectively. We normalize the AP of SPS in *medium* and *hard* scenes with the that of SPS in *easy* scenes. As shown in Fig. 6, ViT-baseline's normAPs of SPS in *medium* and *hard* scenes respectively are 7.7% and 18.4% higher than those of ASNet, respectively. ViT-baseline framework is re-

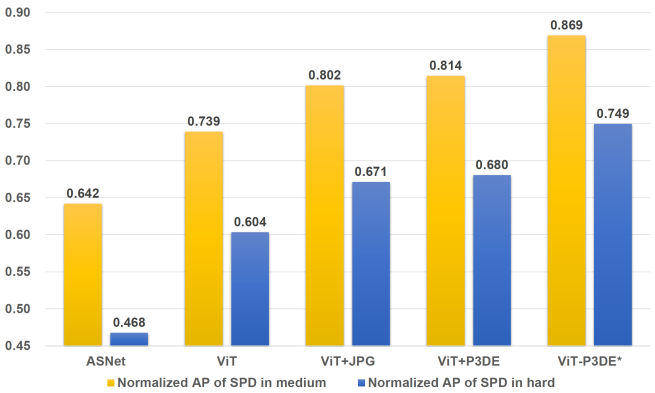

Figure 7: Normalized APs of SPD in *medium* and *hard* scenes

silient to severe occlusion because ViT captures fine-grained features from the small unoccluded part of objects through early long-range receptive fields.

With features that remain uninfluenced despite object occlusion, P3DEs resolve severe occlusion. By capturing the occlusion of the input patches and accordingly utilizing encoded information in P3DEs, ViT-baseline with P3DEs improves by 1.9% and 3.7% normAP in *medium* and *hard* scenes, respectively compared to the ViT-baseline. Meanwhile, JPG only improves the performance on *hard* scenes (-0.8% in *easy*, -1.0% in *medium*, +0.5% in *hard*). This is because using larger receptive fields often causes misidentification to instance-pairs in *easy* and *medium* scenes, which present a considerably lesser degree of occlusion than *hard* scenes. In *easy* and *medium* scenes, the portion of the surrounding objects included in the input patch is quite small even if the objects are positioned densely. Therefore, when JPG is applied, the fraction of neighboring objects increases and the increased fraction can hinder instance association. On the other hand, the fraction of surrounding instances in the input patch is already high in hard scenes. As a result, the strength of JPG is manifested only in hard scenes. P3DEs in ViT-P3DE* negates the negative effects of JPG on *easy* and *medium* scenes by prioritizing the attention on the target object. Ultimately, ViT-P3DE* achieves 1.7% and 6.3% improvements on normAP in *medium* and *hard* scenes, respectively compared to ViT-baseline.

Object Occlusion & Presence of Similar Objects in One Scene

We inspect the frameworks' robustness to instance-pairs that are influenced by two constraints, object occlusion and the presence of similar objects in one scene, at the same time. We first respectively evaluate SPS in *easy* scenes and SPD in *medium* and *hard* scenes, which are classified in terms of the degree of occlusion and the number of objects. Then, we normalize the AP of SPD in *medium* and *hard* scenes with that of SPS in *easy* scenes.

To explain the results clearly, we use the word normAP-(SPS or SPD)-(*med* or *hard*) to represent the normalized AP of SPS or SPD in *medium* or *hard* scenes. As in Fig. 7, ViT-baseline respectively achieves 7.7% and 18.4% improvement compared to that of ASNet in terms of normAP-SPS-*med* and normAP-SPS-*hard*. Meanwhile, ViT-baseline improves normAP-SPD-*med* and normAP-SPD-*hard* by 9.7% and 13.6%, respectively compared to that of ASNet. Since the combination of two harsh constraints makes the association more difficult, the performance improvement of normAP-SPD-*hard* is less than that of normAP-SPS-*hard*. On the other hand, ViT-P3DE* achieves far better performance improvements in both norm-AP-SPD-*med* (13.0%) and norm-AP-SPD-*hard* (14.5%) than those of norm-AP-SPS-*med* (1.7%) and norm-AP-SPS-*hard* (6.3%), respectively compared to ViT-baseline. ViT-P3DE* yields these great results by simultaneously solving both problems caused by object occlusion and the presence of multiple objects that appear similar. Based on these results, we believe that our framework is the key to resolving problems in real-world task.

5 Conclusion

To the best of our knowledge, this is one of the first works that apply ViT to MCIA domain. We also propose pseudo 3D position embeddings (P3DEs) and joint patch generation (JPG) that enhance the robustness of ViT's features under the challenging constraints of MCIA. As a result, our final framework ViT-P3DE* achieves state-of-the-art performance. We believe that our proposed methods are helpful for vision tasks addressing problems that are unresolvable with just visual information. In future work, we plan to examine the applicability of our framework to various ViT variants.

A Ablation Study of P3DEs

To build the framework that learns the object's pseudo 3D position, we jointly train P3DEs with existing input embeddings and add P3DEs to cls embedding with regularization parameter λ when computing loss. We inspect the effectiveness of each method with ViT-baseline framework that exploits the ViT-T backbone. JT denotes the method that jointly trains P3DEs and input embeddings. Add denotes the method that adds P3DEs to cls embedding.

As shown in Table 7, JT occupies a large portion of ViT-P3DE's performance improvement. While Add enhances 1.1% AP, 0.3% FPR-95, and 1.1% IPAA-100, JT improves 6%, 2%, 10% in AP, FPR-95, and IPAA-100 respectively. Through these results, we demonstrate that ViT almost fully utilizes P3DEs with a self-attention mechanism.

B Ablation Studies of Hyperparameters

B.1 Grid Ratio

Grid ratio G is one of the key parameters because the grid ratio determines whether all elements of the embedding table

Method	JT	Add	AP ↑	FPR-95 ↓	IPAA-100 ↑
ViT-baseline	✗	✗	0.705	0.079	0.279
	✓	✗	0.765	0.059	0.379
ViT-P3DE	✓	✓	0.776	0.056	0.390

Table 7: Ablation study of P3DEs

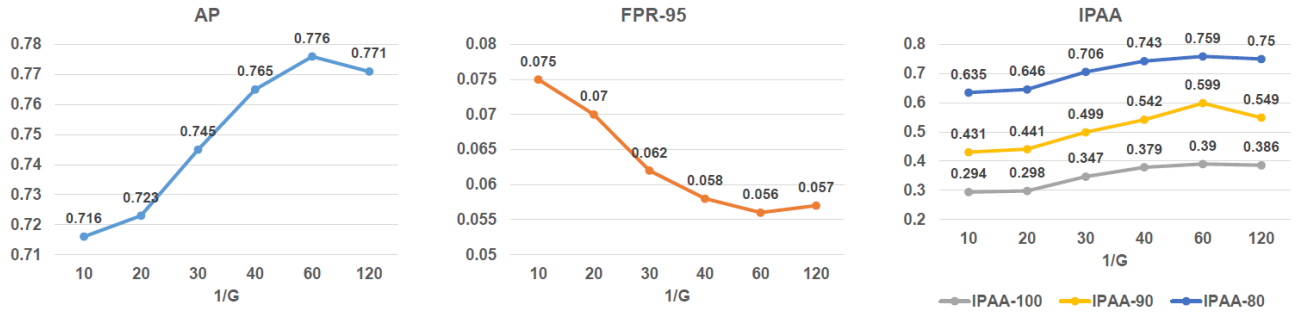

Figure 8: Ablation study of G

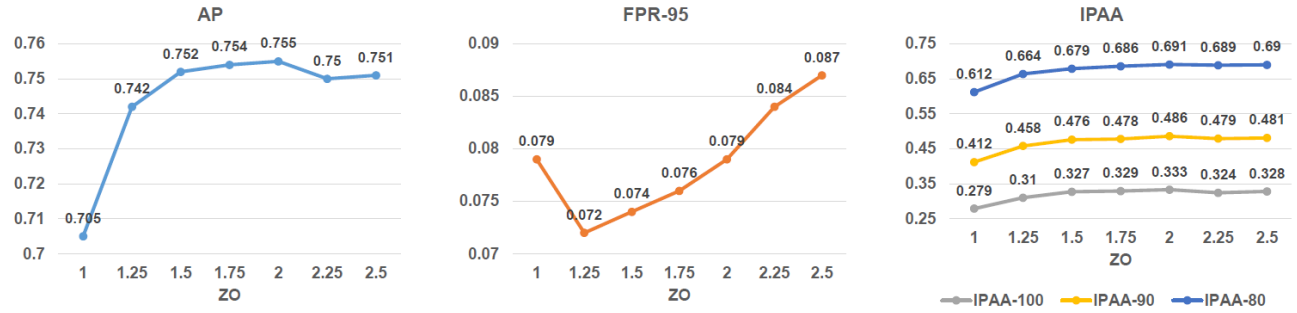

Figure 9: Ablation study of ZO

are sufficiently trained. Moreover, it controls the number of pixels that the embedding represents. To select the best G, we set the $1/G$ as common divisors of image height and width, which respectively are 1080 and 1920 in the MessyTable dataset, and inspect the performance of ViT-P3DE in each case. We set the regularization parameter λ as 0.1. As shown in Fig. 8, when $1/G$ is between 10 and 20, the performances of ViT-P3DE are quite low because many elements in the embedding table are not trained enough. As $1/G$ gradually increases to 60, the elements of the embedding table become sufficiently trained and then the performance is improved. However, when $1/G$ becomes large, such as 120, the performance improvement by P3DEs is less than that of the case that $1/G$ is 60 because one embedding represents the number of pixels that is more than necessary. Through these results, we find that the embedding table must be at least a certain size to ensure that the elements of the embedding table are sufficiently trained but the embedding table should not be too large to represent the position features of each pixel. Therefore, we set the grid ratio G of ViT-P3DE as one over sixty.

B.2 Zoom-Out Ratio

Determining zoomout-ratio (ZO) is important. This is because if a value of ZO is too large, the performance improvement of JPG drops due to putting noise in the instance-pairs that can be associated only with the object's appearance. To maximize the performance gain of JPG, we conduct experiments with varying ZO from 1 to 2.5. As shown in Fig. 9, using even a little bit of surrounding features significantly improves performance because it allows ViT to reinforce the relationship information between an object and its surroundings. Until ZO is 2, both AP and IPAA improve as ZO increases. When ZO exceeds 2, AP and IPAA are similar to or lower than those of the framework with ZO of 2. However, FPR-95 becomes worse as ZO increases except in the case that ZO is between 1 and 1.25 by additionally introducing surrounding information. We select the value of ZO as 2 with which ViT-baseline+JPG achieves maximum performance improvement in terms of AP and IPAA and yields acceptable FPR-95.

B.3 Regularization Parameter

We add P3DEs to cls embedding right before computing loss to fully use P3DEs. We examine how much more features of P3DEs are necessary to increase performance. We conduct an experiment with ViT-P3DE that uses varying regularization parameters λ with grid ratio of 10. As shown in Table 8, performance is not that sensitive to λ because cls embedding includes sufficient features of P3DEs through an attention mechanism. Among values of λ in Table 8, we choose λ as 0.1 that increases the performance of ViT-P3DE.

λ	AP ↑	FPR-95 ↓	IPAA-100 ↑	IPAA-90 ↑	IPAA-80 ↑
0	0.713	0.076	0.289	0.428	0.629
0.05	0.708	0.078	0.286	0.422	0.623
0.1	**0.716**	**0.075**	0.294	**0.431**	**0.635**
0.2	0.713	0.076	0.291	0.428	0.629
0.3	0.711	**0.075**	**0.295**	0.429	0.631

Table 8: Ablation study of λ

Acknowledgements

This work was supported in part by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.RS-2023-00228255,PIM-NPU Based Processing System Software Developments for Hyper-scale Artificial Neural Network Processing) and in part by the MSIT, Korea, under the ITRC(Information Technology Research Center) support program(IITP-2023-2020-0-01461) supervised by the IITP.

References

[Bogdan *et al.*, 2018] Oleksandr Bogdan, Viktor Eckstein, Francois Rameau, and Jean-Charles Bazin. Deepcalib: A deep learning approach for automatic intrinsic calibration of wide field-of-view cameras. In *Proceedings of the 15th ACM SIGGRAPH European Conference on Visual Media Production*, pages 1–10, 2018.

[Cai *et al.*, 2020] Zhongang Cai, Junzhe Zhang, Daxuan Ren, Cunjun Yu, Haiyu Zhao, Shuai Yi, Chai Kiat Yeo, and Chen Change Loy. Messytable: Instance association in multiple camera views. In *European Conference on Computer Vision*, pages 1–16. Springer, 2020.

[Chavdarova *et al.*, 2018] Tatjana Chavdarova, Pierre Baqué, Stéphane Bouquet, Andrii Maksai, Cijo Jose, Timur Bagautdinov, Louis Lettry, Pascal Fua, Luc Van Gool, and François Fleuret. Wildtrack: A multi-camera hd dataset for dense unscripted pedestrian detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5030–5039, 2018.

[Chen *et al.*, 2022] Cuiqun Chen, Mang Ye, Meibin Qi, Jingjing Wu, Jianguo Jiang, and Chia-Wen Lin. Structure-aware positional transformer for visible-infrared person re-identification. *IEEE Transactions on Image Processing*, 31:2352–2364, 2022.

[Choi *et al.*, 2019] Jinwoo Choi, Chen Gao, Joseph CE Messou, and Jia-Bin Huang. Why can't i dance in the mall? learning to mitigate scene bias in action recognition. *Advances in Neural Information Processing Systems*, 32, 2019.

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[Dosovitskiy *et al.*, 2021] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations*, 2021.

[Han *et al.*, 2015] Xufeng Han, Thomas Leung, Yangqing Jia, Rahul Sukthankar, and Alexander C. Berg. Matchnet: Unifying feature and metric learning for patch-based matching. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3279–3286, 2015.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[He *et al.*, 2021] Shuting He, Hao Luo, Pichao Wang, Fan Wang, Hao Li, and Wei Jiang. Transreid: Transformer-based object re-identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15013–15022, 2021.

[He *et al.*, 2022] Ju He, Jie-Neng Chen, Shuai Liu, Adam Kortylewski, Cheng Yang, Yutong Bai, and Changhu Wang. Transfg: A transformer architecture for fine-grained recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 852–860, 2022.

[Kingma and Ba, 2015] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations*, 2015.

[Koch *et al.*, 2015] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2, page 0. Lille, 2015.

[Liu *et al.*, 2021] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[Luo *et al.*, 2021] Wenhan Luo, Junliang Xing, Anton Milan, Xiaoqin Zhang, Wei Liu, and Tae-Kyun Kim. Multiple object tracking: A literature review. *Artificial Intelligence*, 293:103448, 2021.

[Naseer *et al.*, 2021] Muzammal Naseer, Kanchana Ranasinghe, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Intriguing properties of vision transformers. In *Advances in Neural Information Processing Systems 34*, pages 23296–23308, 2021.

[Raghu *et al.*, 2021] Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? *Advances in Neural Information Processing Systems*, 34, 2021.

[Roig *et al.*, 2011] Gemma Roig, Xavier Boix, Horesh Ben Shitrit, and Pascal Fua. Conditional random fields for multi-camera object detection. In *2011 International Conference on Computer Vision*, pages 563–570. IEEE, 2011.

[Schroff *et al.*, 2015] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[Selvaraju *et al.*, 2017] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[Simo-Serra *et al.*, 2015] Edgar Simo-Serra, Eduard Trulls, Luis Ferraz, Iasonas Kokkinos, Pascal Fua, and Francesc Moreno-Noguer. Discriminative learning of deep convolutional feature point descriptors. In *Proceedings of the IEEE international conference on computer vision*, pages 118–126, 2015.

[Susanto *et al.*, 2012] Wandi Susanto, Marcus Rohrbach, and Bernt Schiele. 3d object detection with multiple kinects. In *European Conference on Computer Vision*, pages 93–102. Springer, 2012.

[Touvron *et al.*, 2021] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[Wang *et al.*, 2022] Tao Wang, Hong Liu, Pinhao Song, Tianyu Guo, and Wei Shi. Pose-guided feature disentangling for occluded person re-identification based on transformer. In *Thirty-Sixth AAAI Conference on Artificial Intelligence*, pages 2540–2549, 2022.

[Yun *et al.*, 2022] Ilwi Yun, Hyuk-Jae Lee, and Chae Eun Rhee. Improving 360 monocular depth estimation via non-local dense prediction transformer and joint supervised and self-supervised learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 3224–3233, 2022.

[Zagoruyko and Komodakis, 2015] Sergey Zagoruyko and Nikos Komodakis. Learning to compare image patches via convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4353–4361, 2015.

[Zhou, 2018] Zhi-Hua Zhou. A brief introduction to weakly supervised learning. *National science review*, 5(1):44–53, 2018.