Discrete Latent Perspective Learning for Segmentation and Detection

Deyi Ji [1 2] **Feng Zhao** [1] **Lanyun Zhu** [3] **Wenwei Jin** [2] **Hongtao Lu** [4] **Jieping Ye** [2]

Abstract

In this paper, we address the challenge of Perspective-Invariant Learning in machine learning and computer vision, which involves enabling a network to understand images from varying perspectives to achieve consistent semantic interpretation. While standard approaches rely on the labor-intensive collection of multi-view images or limited data augmentation techniques, we propose a novel framework, Discrete Latent Perspective Learning (DLPL), for latent multi-perspective fusion learning using conventional single-view images. DLPL comprises three main modules: Perspective Discrete Decomposition (PDD), Perspective Homography Transformation (PHT), and Perspective Invariant Attention (PIA), which work together to discretize visual features, transform perspectives, and fuse multi-perspective semantic information, respectively. DLPL is a universal perspective learning framework applicable to a variety of scenarios and vision tasks. Extensive experiments demonstrate that DLPL significantly enhances the network's capacity to depict images across diverse scenarios (daily photos, UAV, auto-driving) and tasks (detection, segmentation).

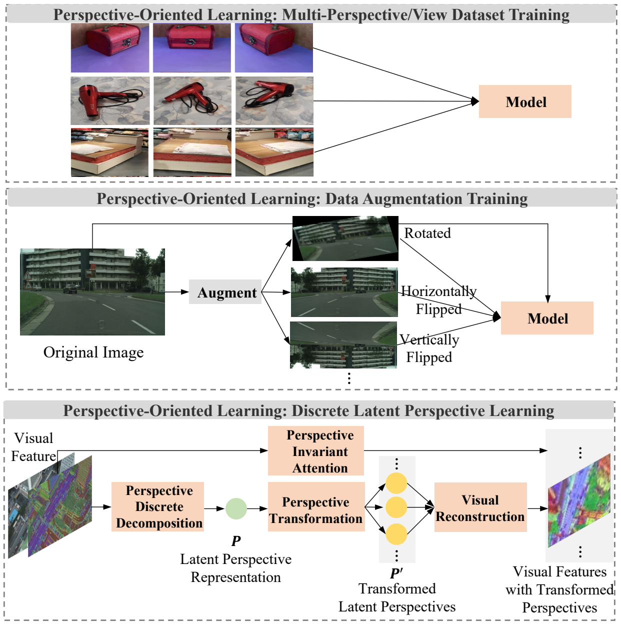
Perspective-Oriented Learning: Multi-Perspective/View Dataset Training
Model
Perspective-Oriented Learning: Data Augmentation Training
Original Image
Augment
Rotated
Horizontally Flipped
Vertically Flipped
Model
Perspective-Oriented Learning: Discrete Latent Perspective Learning
Visual Feature
Perspective Discrete Decomposition
P
Latent Perspective Representation
Perspective Invariant Attention
Perspective Transformation
P′
Transformed Latent Perspectives
Visual Reconstruction
Visual Features with Transformed Perspectives

Figure 1. The comparison of perspective-oriented learning methods: Perspective Augmentation, Multi-View Data Training, and the proposed Discrete Latent Perspective Learning.

1. Introduction

According to the theory of photography (Elkins, 2006; Bengio et al., 2013), image perspective/viewpoint refers to the perception of depth and spatial relationships between objects within an image. In the fields of machine learning and computer vision, current research often employs a variety of methods to enhance networks' **Perspective-Invariant Learning** (YU2, 2018; LO1, 1997), which involves enabling the network to learn scene information from different viewpoints and ensuring consistency in semantic learning across these varying perspectives, thus improving the portrayal of semantic information within images[1]. As shown in Figure 1, the most commonly used approaches include: (1) collecting multi-view images of a given scene to conduct multi-view training (Yu et al., 2023), and (2) utilizing data augmentation techniques (Cubuk et al., 2019), such as rotation and flipping, to increase the diversity of perspectives.

However, in the former case, the acquisition of multi-view images is significantly more challenging than that of single-view images, since the latter can rely on fixed cameras with a wide range of sources (e.g., Internet, vehicle-mounted cameras, surveillance cameras), whereas multi-view im-

[1]University of Science and Technology of China [2]Alibaba Group [3]Singapore University of Technology and Design [4]Dept. of CSE, MOE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University. Correspondence to: Feng Zhao <fzhao956@ustc.edu.cn>, Jieping Ye <yejieping.ye@alibaba-inc.com>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

[1] In this paper, the terms "perspective" and "viewpoint" are used interchangeably, the former is derived from the theory of photography, while the latter is more commonly employed in general discourse.

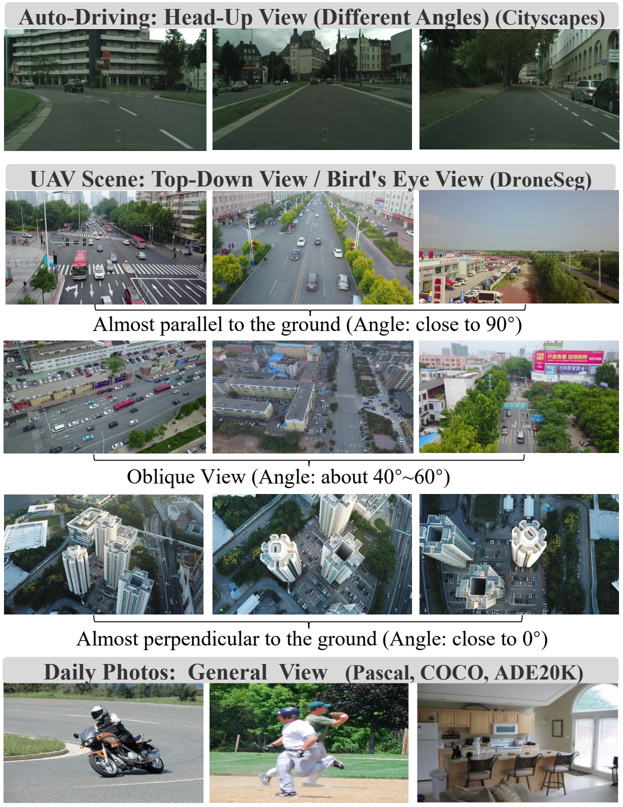
Auto-Driving: Head-Up View (Different Angles) (Cityscapes)
UAV Scene: Top-Down View / Bird's Eye View (DroneSeg)
Almost parallel to the ground (Angle: close to 90°)
Oblique View (Angle: about 40°~60°)
Almost perpendicular to the ground (Angle: close to 0°)
Daily Photos: General View (Pascal, COCO, ADE20K)

Figure 2. The variation in image perspectives in three typical scenarios: auto-driving, UAV flight, and daily photos. UAV images exhibit more diverse and dynamic viewpoints due to variations in flight heights, angles, and other factors. However, these datasets lack multi-view images for a single scene, preventing existing networks from effective perspective-invariant learning of the images.

ages are heavily dependent on manual collection, requiring multi-directional shooting of a scene, which is both time-consuming and labor-intensive. In practice, the costs associated with this method are difficult to justify, and the organization and annotation of multi-view images are also more expensive due to the need to ensure consistency in labeling. In the latter case, data augmentation methods often employ hand-crafted rotation operations to augment image perspectives, aiming to enhance training diversity. However, these operations yield limited performance improvements as they generate similar and rigid results, failing to capture the practical perspective changes encountered in real-world applications, such as Unmanned Aerial Vehicle (UAV) flights, auto-driving and daily photos, as shown in Figure 2.

In this paper, we propose to perform latent multi-perspective fusion of Invariant Learning within the discrete feature space, based on conventional single-view images. To achieve this, we present a systematic Discrete Latent Perspective Learning (DLPL) framework, comprising Perspective Representation, Transformation, and Multi-Perspective Fusion. Firstly, since perspective lacks a clear formulation in current research, we begin with Perspective Discrete Decomposition (PDD) to discretize the visual feature[2] to decompose the latent perspective representation. Following this, in line with photography theory, we treat the perspective as a spatial projection of the image, and transform it to generate new descriptive perspectives with the Perspective Homography Transformation (PHT) module. These new perspectives correspond to alternatives spatial projection of the image. Finally, we integrate both the perspectives for fusion learning, with Perspective Invariant Attention (PIA), to obtain perspective-invariant semantic information of the image, ultimately enhancing the model's ability to depict the image. It is important to note that DLPL is a universal learning framework for image perspectives and applicable broadly to different scenarios (daily photos, UAVs, Auto-driving) and various vision tasks (classification, detection, segmentation).

The overall contributions can be summarized as follows:

- DLPL is a general perspective-invariant learning framework that enables multi-perspective fusion learning within latent feature space for single-view images.
- Specifically, DLPL systematically includes key processes such as Perspective Representation (PDD), Perspective Transformation (PHT), and Multi-Perspective fusion (Multi-Level PIA).
- Experiments demonstrate that DLPL achieves impressive results across a wide range of scenarios (daily photos, UAV, Auto-driving) and different vision tasks (detection and segmentation).

2. Related Work

2.1. Scene Understanding

In existing visual datasets, changes in the perspective of images are very common and pose numerous challenges to image understanding. Among these, high-altitude scenes, such as those captured by UAVs or satellites, feature particularly rich variations in viewpoints. Initial works on core

[2] In this paper, we define "visual feature" as characteristics extracted directly from input images using conventional CNNs or ViTs, primarily containing visual information and representing the image's semantics. This aligns with the understanding in existing works (Caron et al., 2020; 2018; Jing & Tian, 2020). "Perspective feature/representation" refers to features related to the image's perspective, extracted from visual features using our designed Perspective Discrete Decomposition (PDD) module, mainly representing the image's perspective information. Additionally, visual features can be reconstructed from perspective representations through Visual Reconstruction (VR) module.

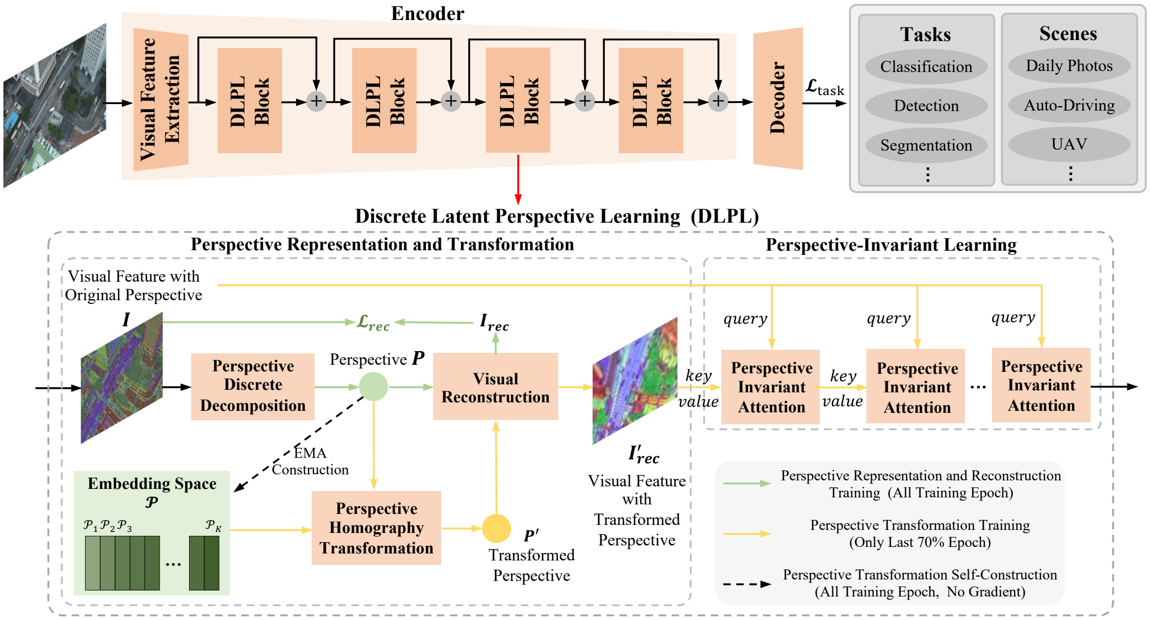
Encoder
Visual Feature Extraction
DLPL Block
DLPL Block
DLPL Block
DLPL Block
Decoder
$\mathcal{L}_{\text{task}}$
Tasks
Classification
Detection
Segmentation
Scenes
Daily Photos
Auto-Driving
UAV
Discrete Latent Perspective Learning (DLPL)
Perspective Representation and Transformation
Perspective-Invariant Learning
Visual Feature with Original Perspective
I
$\mathcal{L}_{rec}$
I_{rec}
Perspective Discrete Decomposition
Perspective P
Visual Reconstruction
EMA Construction
Embedding Space $\mathcal{P}$
$\mathcal{P}_1 \mathcal{P}_2 \mathcal{P}_3$
$\mathcal{P}_K$
Perspective Homography Transformation
P'
Transformed Perspective
I'_{rec}
Visual Feature with Transformed Perspective
query
key
value
Perspective Invariant Attention
Perspective Invariant Attention
Perspective Invariant Attention
Perspective Representation and Reconstruction Training (All Training Epoch)
Perspective Transformation Training (Only Last 70% Epoch)
Perspective Transformation Self-Construction (All Training Epoch, No Gradient)

Figure 3. The architecture of DLPL framework within a Vision Transformer (ViT) setup, which is divided into an encoder with a Visual Feature Extraction module and four DLPL blocks, and a task-specific decoder. The DLPL block include: (1) Perspective Discrete Decomposition (PDD) module decomposes input visual features $\mathbf{I}$ into a latent representation $\mathbf{P}$; (2) Visual Reconstruction (VR) module reconstructs visual features from $\mathbf{P}$ with a loss function $\mathcal{L}_{rec}$; (3) Perspective Embedding Space $\mathcal{P}$ constructed through EMA, based on $\mathbf{P}$; (4) Perspective Transformation leverages Homography for perspective variation, followed by VR to obtain reconstructed visual features $\mathbf{I}'_{rec}$ of the transformed perspective $\mathbf{P}'$. Finally, Perspective-Invariant Learning fuses the features of the original and transformed perspectives using Perspective-Invariant Attention (PIA) to learn semantic information invariant to perspective changes.

high-altitude vision tasks like object detection and semantic segmentation (Fu et al., 2022; Ji et al., 2023b;a; Chen et al., 2024b) often adapt network architectures from methods designed for natural scenes to establish baseline results. In contrast, the variations in perspectives within auto-driving (Cordts et al., 2016) and daily photo (Zhou et al., 2017) scenarios are not as pronounced. A plethora of methods have been proposed, including early CNN-based approaches (Chen et al., 2018a; Yuan et al., 2020; Zhu et al., 2021a; Ji et al., 2021; 2022; Hu et al., 2020; Ji et al., 2024a;b; Wang et al., 2021b;a; 2023; Feng et al., 2018; Ji et al., 2019) and recent methodologies based on Vision Transformers (ViTs) (Xie et al., 2021; Zheng et al., 2021; Yu et al., 2022; Carion et al., 2020; Zhu et al., 2023b;c; 2024a), as well as emerging studies on large vision models and multimodal models (Kirillov et al., 2023; Zhu et al., 2023a; Chen et al., 2023b; Zhu et al., 2024b). Nevertheless, economically and effectively addressing the challenges posed by perspective changes remains a difficult issue to be resolved.

2.2. Perspective-Oriented Learning

Current work on Perspective Learning primarily focuses on data augmentation techniques based on single-view images (Cordts et al., 2016; Zhou et al., 2017) and joint training with multi-view datasets (Yu et al., 2023; Chen et al., 2023a). A substantial body of research on data augmentation techniques (Mumuni & Mumuni, 2022; Cubuk et al., 2019; Chen et al., 2024a) has been conducted in existing works, including earlier strategies based on manual rules and later studies that are based on trained learning approaches. The construction of multi-view datasets is a more recent area of research, which involves capturing scenes and targets from multiple orientations. The creation of such datasets often demands significant human effort, and in tasks requiring fine-grained annotation like segmentation, there is the challenge of maintaining consistency across multiple orientations, that is, ensuring that the fine-grained annotations for the same target from different perspectives have consistent semantic meanings.

3. Discrete Latent Perspective Learning

3.1. Top-Down Overview

DLPL is a flexible and systematic framework for enhancing perspective-invariant learning capabilities in both CNNs and ViT networks. Specifically, as shown in Figure 3, taking a general ViT architecture as an example, DLPL can be implemented in the form of Transformer Blocks. The encoder includes a basic Visual Feature Extraction module and four DLPL Blocks. The decoder, on the other hand,

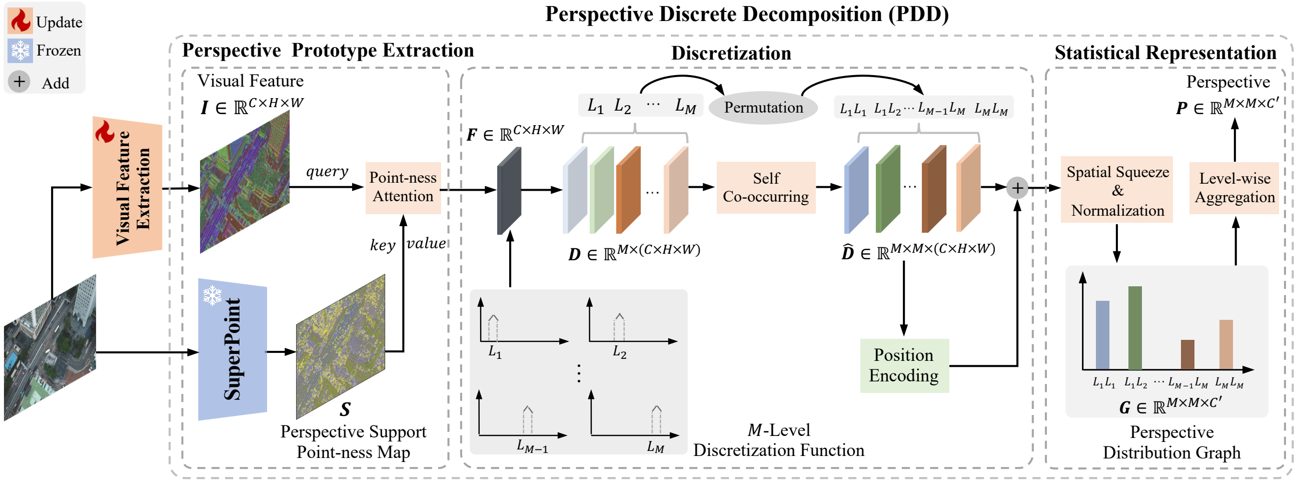
Update
Frozen
Add
Perspective Discrete Decomposition (PDD)
Perspective Prototype Extraction
Discretization
Statistical Representation
Visual Feature Extraction
Visual Feature
$I \in \mathbb{R}^{C\times H\times W}$
query
key
value
Point-ness Attention
SuperPoint
S
Perspective Support Point-ness Map
$F \in \mathbb{R}^{C\times H\times W}$
L_1 L_2 ··· L_M
Permutation
L_1L_1 L_1L_2 ··· $L_{M-1}L_M$ L_ML_M
Self Co-occurring
$D \in \mathbb{R}^{M\times(C\times H\times W)}$
$\hat{D} \in \mathbb{R}^{M\times M\times(C\times H\times W)}$
Position Encoding
L_1
L_2
L_{M-1}
L_M
M-Level Discretization Function
Spatial Squeeze & Normalization
Level-wise Aggregation
Perspective
$P \in \mathbb{R}^{M\times M\times C'}$
L_1L_1 L_1L_2 ··· $L_{M-1}L_M$ L_ML_M
$G \in \mathbb{R}^{M\times M\times C'}$
Perspective Distribution Graph

Figure 4. The illustration of PDD. It takes visual features $\mathbf{I}$ as input and processes through three key steps: (1) Extraction of perspective prototypes from a point-ness map $\mathbf{S}$ using spatially structured supporting points; (2) Discretization of the continuous perspective distribution into M levels within the latent feature space; (3) Construction of a statistical perspective graph $\mathbf{G}$ by encoding the intensity and positional relationships of discretized levels. This process yields the latent perspective feature map $\mathbf{P}$.

chooses the appropriate structure based on the specific task type (classification/segmentation/detection).

Extracting Original Perspective Representation (Section 3.2). Within each DLPL Block, given the input visual feature $\mathbf{I}$ (which contains the original image perspective), we extract the original perspective representation from $\mathbf{I}$, by the Perspective Discrete Decomposition (PDD) module.

Reconstructing Visual Feature (Section 3.3). In order to ensure that the visual feature can be reconstructed from the perspective representation, we design a Visual Reconstruction (VR) module. We constrain $\mathbf{I}$ and the reconstructed visual feature $\mathbf{I}_{rec}$ to be equivalent through a loss function $\mathcal{L}_{rec}$. This is a self-training process.

Perspective Space Construction (Section 3.4). Concurrently, during training, based on the perspective $\mathbf{P}$ of each image, we can incrementally build the Perspective Embedding Space $\mathcal{P}$ for the entire dataset. To maintain training stability, we update $\mathcal{P}$ using an Exponential Moving Average (EMA) method without backpropagating gradients.

Perspective Transformation (Section 3.5). As the VR's reconstruction ability and $\mathcal{P}$ become stable, we leverage the spatial Homography properties of image perspective, combined with the distribution of $\mathcal{P}$, to transform $\mathbf{P}$ and obtain another perspective $\mathbf{P}'$, of the image. Then, through VR, we obtain the corresponding visual feature $\mathbf{I}'_{rec}$ of $\mathbf{P}'$.

Perspective-Invariant Learning (Section 3.6). We fuse the visual feature $\mathbf{I}$ (with original perspective $\mathbf{P}$) and $\mathbf{I}'_{rec}$ (with transformed perspective $\mathbf{P}'$) using multi-level Perspective-Invariant Attention (PIA) to learn the network's semantic expression of image information under different perspectives.

In training, Visual Reconstruction and Perspective Space Construction accompany the entire training process. In contrast, Perspective Transformation training begins after the first two processes stabilize, at the 30% of the entire training epoch. Perspective-Invariant Learning also occurs throughout the entire training process. During the initial 30% training epoch, since $\mathbf{P}'$ is not ready, PIA degenerates to Self-Attention learning solely based on $\mathbf{P}$.

3.2. Perspective Discrete Decomposition

Since perspective lacks a clear formulation in current research, we begin with the PDD module. Drawing from visual features extracted by the previous module, we construct prototypes of image perspectives based on key supporting vectors of the perspective. These are then discretized within the latent feature space and statistically analyzed to decompose the implicit perspective distribution map $\mathbf{P}$, as shown in Figure 4.

Formally, PDD receives the visual feature $\mathbf{I}^{C\times H\times W}$ as input and produces a perspective feature $\mathbf{P} \in \mathbb{R}^{M\times M\times C'}$, where C, C' are the feature channels, H, W are the spatial dimensions, M is the discretization levels. Generally, PDD exploits the explicit well-formulated distribution of structured supporting points which build up the spatial structured prototype to tell the implicit perspective information. In concrete, PDD consists of three procedures:

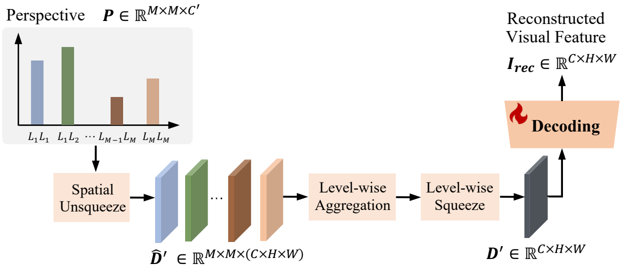
Perspective $P \in \mathbb{R}^{M\times M\times C'}$
L_1L_1 L_1L_2 $\cdots$ $L_{M-1}L_M$ L_ML_M
Spatial Unsqueeze
$\hat{D}' \in \mathbb{R}^{M\times M\times(C\times H\times W)}$
Level-wise Aggregation
Level-wise Squeeze
$D' \in \mathbb{R}^{C\times H\times W}$
Decoding
Reconstructed Visual Feature
$I_{rec} \in \mathbb{R}^{C\times H\times W}$

Figure 5. The illustration of Visual Reconstruction (VR). It is essentially the inverse of the PDD process, which reconstructs the visual feature from perspective representation.

Perspective Prototype Extraction. Besides $\mathbf{I}$, we utilize a pretrained SuperPoint (DeTone et al., 2018) network to extract point-ness map $\mathbf{S}$ of the original image, which serve as the supporting structure of image perspective. The theoretical basis behind this motivation stems from related research such as multi-view visual understanding(Ma et al., 2021; Dong et al., 2019; Yağmur & Ates, 2023), which utilizes the structured spatial information of keypoints as support points to characterize the spatial structure of the image's perspective during actual photography (Zeng et al., 2018; Liu & Li, 2023; Zhang & Hanson, 1996; Li et al., 2020). Next, we apply point-ness attention to $\mathbf{I}$ and $\mathbf{S}$ to obtain a basic perspective prototype,

$$\mathbf{F} = \mathrm{Softmax}(\frac{\mathbf{I} \times \mathbf{S}^\top}{\sqrt{C}}) \times \mathbf{S}, \tag{1}$$

where $\top$ is the matrix transposition operation. $\mathbf{F} \in \mathbb{R}^{C\times H\times W}$ is the perspective-structured visual feature.

Discretization. In order to effectively represent perspective within deep neural networks, we firstly perform Discretization on it. Inspired by (Zhu et al., 2021a; Ji et al., 2022), PDD defines M levels by equally dividing $\mathbf{F}$ in the spatial dimension. Specifically the m-th ($m \in [1, M]$) level $\mathbf{L}_m \in \mathbb{R}^C$ is computed as:

$$\mathbf{L}_m = \frac{m \cdot \max(\mathbf{F}) + (M-m)\cdot \min(\mathbf{F})}{M}. \tag{2}$$

Next, based on $\mathbf{L}$, we use M discretization functions to discretize F into an embedding $\mathbf{D} \in \mathbb{R}^{M\times C\times H\times W}$, where each $\mathbf{F}_{i,j} (i \in [1, H], j \in [1, W])$ is discretized into $\mathbf{D}_{i,j} \in \mathbb{R}^{M\times C}$. Specifically,

$$\mathbf{D}_{m,i,j} = \begin{cases} \mathbf{e} - |\mathbf{L}_m - \mathbf{F}_{i,j}| & \\ & \text{if } |\mathbf{L}_m - \mathbf{F}_{i,j}| < \dfrac{\mathbf{L}_{m+1} - \mathbf{L}_m}{2} \\ \mathbf{0} & \text{else} \end{cases}, \tag{3}$$

where $\mathbf{e} \in \mathbb{R}^C, \mathbf{0} \in \mathbb{R}^C$ denote the Unit vector and Zero vector respectively. Note a spireshaped function is used instead of a binary one to ensure that the operation is differentiable. $\mathbf{D}$ can be viewed as the form of a histogram for perspective distribution. According to the theory of photography, the perspective is related to spatial relationship between objects/pixels within the image. Correspondingly, we encode the spatial pixel co-occurring relationships by:

$$\begin{aligned} \hat{\mathbf{D}}_{i,j} = \mathbf{D}_{i,j}\mathbf{D}_{i,j}^\top, \quad \hat{\mathbf{L}} = \{\hat{\mathbf{L}}_{\hat{m}}\}, \\ \hat{m} = [m_1, m_2], m_1 \in [1, M_1], m_2 \in [1, M_2], \end{aligned} \tag{4}$$

where $M_1 = M_2 = M$, $\hat{\mathbf{q}} \in \mathbb{R}^{M\times M\times C\times H\times W}$ is spatial co-occurring perspective histogram, and $\hat{\mathbf{L}}$ is the corresponding co-occurring level. $\top$ indicates matrix transposition.

Statistical Representation. Further, we construct the perspective graph $\mathbf{G} \in \mathbb{R}^{M\times M\times C'}$ by squeezing the spatial dimensions in form of counting the intensity of each level, then encoding with position corresponding relationship (Zhu et al., 2021a; Ji et al., 2022):

$$\begin{aligned} \mathbf{G}_{\hat{m}} =& \Phi(\frac{\sum_{i=1}^{H}\sum_{j=1}^{W}\hat{\mathbf{D}}_{m,i,j}}{\sum_{m_1=1}^{M_1}\sum_{m_2=1}^{M_2}\sum_{i=1}^{H}\sum_{j=1}^{W}\hat{\mathbf{D}}_{m,n,i,j}} \\ &+ \sum_{i=1}^{H}\sum_{j=1}^{W}\hat{\mathbf{D}}_{m_1,m_2,i,j}\mathrm{Pos\_Enc}(i,j)), \end{aligned} \tag{5}$$

where $\mathrm{Pos\_Enc}$ indicates the position encoding operation. Cat is the concatenation operation. Φ is MLP layer for dimension normalization. As shown in Figure 4, $\mathbf{G}$ can be viewed as a Perspective Distribution Graph, where the levels $\{L_1L_1, L_1L_2, ..., L_{M-1}L_M, L_ML_M\}$ are nodes. Finally, we correlate the perspective graph over all nodes to generate the overall perspective representation $\mathbf{P} \in \mathbb{R}^{M\times M\times C'}$:

$$\mathbf{P} = \underset{\hat{m}}{\mathrm{MA}}(\mathbf{G}_{\hat{m}}), \tag{6}$$

where $\underset{\hat{m}}{\mathrm{MA}}(\cdot)$ indicate performing multi-head attention over all $\hat{m}$ levels. By decomposing in form of discrete graph in latent embedding space, $\mathbf{P}$ is formulated as pointness-structure-aware thus able to represent the overall perspective information of the image.

3.3. Visual Reconstruction

As illustrated in Section 3.1 and Figure 5, we exploit the VR to reconstruct visual feature $\mathbf{I}_{rec}$ from the perspective $\mathbf{P}$. It is essentially the inverse of the PDD process. Given the input perspective feature $\mathbf{P}$, we first restore the spatial dimension using an unsqueeze operation to obtain $\hat{\mathbf{D}}' \in \mathbb{R}^{M\times M\times(C\times H\times W)}$. Furthermore, we perform level-wise aggregation to mutually enhance the restored spatial information in form of multi-head attention. Afterward, we

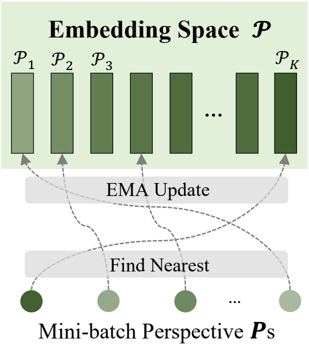

Figure 6. The illustration of the construction of Perspective Embedding Space $\mathcal{P}$, where K is the size of $\mathcal{P}$ and $\mathcal{P}_k$ is the k-th embedding in $\mathcal{P}$ ($k \in [1, K]$). In training, we update $\mathcal{P}$ as a function of the exponential moving average (EMA) of the incoming mini-batch perspective $\mathbf{P}$s from PDD.

embed the discretization levels into the visual feature using a level-wise squeeze method to obtain $\mathbf{D}' \in \mathbb{R}^{C \times H \times W}$, and finally, a transformer decoder is used to reconstruct the final visual feature $\mathbf{I}_{rec}$. All the unsqueeze and squeeze processes can be accomplished using simple multi-layer MLPs.

In order to enable the reconstruction ability (Ji et al., 2023b), we use a reconstruction loss to supervise the decoding process to constrain the structure affinity between $\mathbf{I}$ and $\mathbf{I}_{rec}$,

$$\mathcal{L}_{rec} = ||\mathbf{I}^{\top}\mathbf{I} - \mathbf{I}_{rec}^{\top}\mathbf{I}_{rec}||_2. \tag{7}$$

3.4. Perspective Embedding Space

Based on the decomposed perspective representation $\mathbf{P}$, we define a learnable latent perspective embedding space $\mathcal{P} \in \mathbb{R}^{K \times (M \times M \times C')} = \{\mathcal{P}_k \in \mathbb{R}^{M \times M \times C'}\}(k \in [1, K])$, where K is the size of the latent space. As shown in Figure 6, in training, we update $\mathcal{P}$ as function of exponential moving average (EMA) of the coming mini-batch perspective representations $\mathbf{P}$s from PDD:

$$\text{For each } \ \mathbf{P} \ \text{ in minibatch } \ \mathbf{P}\text{s},$$
$$\mathcal{P}_k := \begin{cases} \alpha\mathcal{P}_k + (1-\alpha)\mathbf{P} & \\ \quad \text{if} \quad \arg\min_k ||\mathcal{P}_k - \mathbf{P}||_2 = k \\ \mathcal{P}_k \qquad\qquad \text{else} \end{cases}, \tag{8}$$

where α is the moving weight and we find $\alpha = 0.9$ to work well in practice.

Then, we can obtain the overall perspective distribution of the whole dataset in form of Gaussian Mixed Model (GMM) over $\mathcal{P}$, as,

$$p = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathcal{P}_k, \mathbf{\Sigma}_k), \quad \sum_{k=1}^{K} \pi_k = 1, \ \ 0 \leq \pi_k \leq 1, \tag{9}$$

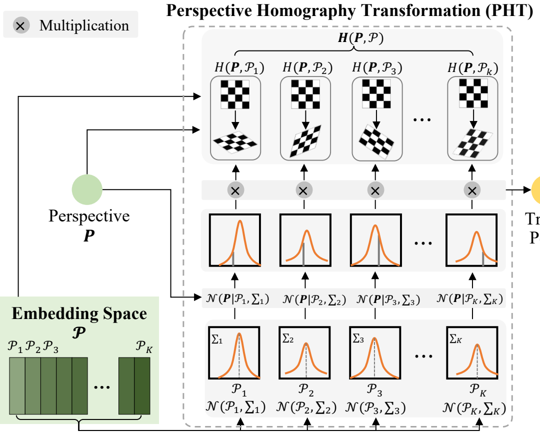
Multiplication
Perspective Homography Transformation (PHT)
$H(\boldsymbol{P},\mathcal{P})$
$H(\boldsymbol{P},\mathcal{P}_1)$
$H(\boldsymbol{P},\mathcal{P}_2)$
$H(\boldsymbol{P},\mathcal{P}_3)$
$H(\boldsymbol{P},\mathcal{P}_K)$
...
Perspective $\boldsymbol{P}$
$\boldsymbol{P}'$
Transformed Perspective
$\mathcal{N}(\boldsymbol{P}|\mathcal{P}_1,\Sigma_1)$
$\mathcal{N}(\boldsymbol{P}|\mathcal{P}_2,\Sigma_2)$
$\mathcal{N}(\boldsymbol{P}|\mathcal{P}_3,\Sigma_3)$
$\mathcal{N}(\boldsymbol{P}|\mathcal{P}_K,\Sigma_K)$
Embedding Space $\mathcal{P}$
$\mathcal{P}_1$ $\mathcal{P}_2$ $\mathcal{P}_3$
$\mathcal{P}_K$
Σ_1
Σ_2
Σ_3
Σ_K
$\mathcal{P}_1$
$\mathcal{P}_2$
$\mathcal{P}_3$
$\mathcal{P}_K$
$\mathcal{N}(\mathcal{P}_1,\Sigma_1)$
$\mathcal{N}(\mathcal{P}_2,\Sigma_2)$
$\mathcal{N}(\mathcal{P}_3,\Sigma_3)$
$\mathcal{N}(\mathcal{P}_K,\Sigma_K)$

Figure 7. The illustration of PHT. Based on the characteristics of projection transformation, we transform the original perspective $\mathbf{P}$ to generate another descriptive perspective $\mathbf{P}'$, which corresponds to an alternative spatial projection of the image.

where $\mathcal{N}(\cdot)$ indicates the Gaussian Distribution, the k-th component of GMM has the center of $\mathcal{P}^k$ with the variance of $\mathbf{\Sigma}_k$, and π_k is the mixture coefficient.

3.5. Perspective Homography Transformation

In line with photography theory (Elkins, 2006), we treat the perspective $\mathbf{P}$ as a spatial projection of the image (Malis & Vargas Villanueva, 2007). Based on the characteristics of projection transformation, we transform the perspective to generate another descriptive perspective $\mathbf{P}'$, which corresponds to an alternative spatial projection of the image. Based on Homography concept (Kriegman, 2007), the original perspective $\mathbf{P}$ and the transformed perspective $\mathbf{P}'$ form a Homography, and this transformation can be realized with a Homography matrix.

As shown in Figure 7, in concrete, we generate a close semantic-related transformation $\mathbf{P}'$ of the original perspective $\mathbf{P}$, by leveraging the all the perspective distribution and performing Homography transformation:

$$\begin{aligned} \mathbf{P}' = \mathbf{H}(\mathbf{P}, \mathcal{P})\mathbf{P} &= \sum_{k=1}^{K} H(\mathbf{P}, \mathcal{P}_k)p(\mathbf{P}) \\ &= \sum_{k=1}^{K} \pi_n \mathcal{N}(\mathbf{P}|\mathcal{P}_k, \mathbf{\Sigma}_k) H(\mathbf{P}, \mathcal{P}_k), \end{aligned} \tag{10}$$

where $\mathbf{H}(\mathbf{P}, \mathcal{P})$ is the Homography Matrix of the original perspective $\mathbf{P}$ over all the perspective embedding space $\mathcal{P}$, and calculated by the sum of Homography Matrix of $\mathbf{P}$ over each $\mathcal{P}_k$ in the space, that is, $\sum_{k=1}^{K} H(\mathbf{P}, \mathcal{P}_k)$. $H(\mathbf{P}, \mathcal{P}_k)$ indicates the estimation of Homography matrix between $\mathbf{P}$ and $\mathcal{P}_k$. Mathematically, Homography matrix can represent

Table 1. The scenes, datasets and task settings in our experiments.

Scenes	Semantic Segmentation	Object Detection
Auto-Driving	Cityscapes (Cordts et al., 2016)	-
UAV	UDD6 (Chen et al., 2018b) iSAID (Waqas Zamir et al., 2019) UAVid (Lyu et al., 2020) Aeroscapes (Nigam et al., 2018) DroneSeg	VisDrone (Zhu et al., 2021b) UAVDT (Wu et al., 2021)
Daily Photos	ADE20K (Zhou et al., 2017)	-

Table 2. UAV Scene Segmentation: Comparison with state-of-the-arts on UDD6, iSAID, UAVid, Aeroscapes and our proposed DroneSeg datasets.

Method	Param.	mIoU (%)				
		UDD6	iSAID	UAVid	Aeroscapes	DroneSeg
Deeplab	63M	71.84	59.20	56.82	51.40	38.69
OCR_W48	70M	73.37	62.73	63.10	58.19	43.10
PSPNet	67M	72.95	60.30	58.20	57.98	37.03
FarSeg	-	-	63.70	-	-	-
PFNet	-	-	66.90	-	-	-
SCO	-	-	69.10	-	-	-
SETR	318.3M	68.00	62.77	58.52	50.34	48.23
UperNet	234.0M	73.13	66.45	61.91	64.32	53.34
PoolFormer	77.1M	74.54	65.55	61.73	62.27	53.94
SegFormer	84.7M	74.28	67.19	62.01	66.40	55.33
DLPL (SETR)	321.1M	**72.62**	**65.45**	**60.23**	**55.02**	**52.94**
DLPL (UperNet)	236.8M	**76.00**	**68.81**	**63.27**	**66.25**	**56.10**
DLPL (PoolFormer)	79.9M	**75.98**	**68.57**	**64.02**	**65.03**	**56.44**
DLPL (SegFormer)	87.5M	**76.88**	**70.60**	**65.30**	**68.22**	**59.10**

the coordinates transformation from one perspective plane to another. In our formulation, $\mathbf{P}$ and $\mathcal{P}_k$ act as two perspective planes, thus $H(\mathbf{P}, \mathcal{P}_k)$ can characterize the perspective transformation from $\mathbf{P}$ to $\mathcal{P}_k$.

Finally, we can obtain the corresponding visual feature $\mathbf{I}'_{rec}$ with the transformed perspective $\mathbf{P}'$, by applying the Visual Reconstruction (VR) module to $\mathbf{P}'$.

3.6. Perspective-Invariant Learning

By the PHT and VR, we obtain the semantic-related perspective-transformed visual feature $\mathbf{I}'_{rec}$ for the original visual feature $\mathbf{I}$. As seen that $\mathbf{I}$ and $\mathbf{I}'_{rec}$ contain closely identical scene context and semantic information yet differ in perspective. Next, they are aggregated by the multi-level Perspective Invariant Attention (PIA) to leverage the relationship between the $\mathbf{I}$ (with original perspective $\mathbf{P}$) and $\mathbf{I}'_{rec}$ (with transformed perspective $\mathbf{P}'$). Formally, PIA is formulated as a cross-attention manner (Chen et al., 2021):

$$\mathrm{PIA}(\mathbf{I}, \mathbf{I}'_{rec}) = \mathrm{Softmax}(\frac{\mathbf{I} \times \mathbf{I}^{'\top}_{rec}}{\sqrt{C}}) \times \mathbf{I}^{'}_{rec}. \tag{11}$$

Table 3. UAV Scene Detection: Comparison with state-of-the-arts on VisDrone and UAVDT datasets.

Method	VisDrone-*dev*			UAVDT-*test*		
	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}
Dynamic R-CNN	13.70	24.70	13.50	57.60	65.30	64.6
RetinaNet	18.94	31.67	20.25	33.95	-	-
FRCNN(PVT2-B0)	19.40	31.80	20.80	60.80	71.50	69.8
RefineDet	19.89	37.27	20.18	-	-	-
DetNet	20.07	37.54	21.26	-	-	-
FPN	22.06	39.57	22.50	49.05	-	-
Light-RCNN	22.08	39.56	23.24	-	-	-
FRCNN(ResNet-50)	22.40	37.20	23.70	55.70	69.80	66.90
CornerNet	23.43	41.18	25.02	-	-	-
Cascade-RCNN	25.20	40.10	26.80	61.50	73.50	71.70
DETR	24.32	40.67	25.89	61.10	73.41	71.20
Deformable DETR	26.28	43.02	27.90	62.99	76.32	73.78
Sparse DETR	26.98	44.30	28.70	63.21	76.81	74.22
DLPL (DETR)	**26.45**	**43.28**	**27.95**	**63.36**	**76.80**	**74.00**
DLPL (Deformable DETR)	**28.00**	**45.89**	**29.60**	**65.07**	**78.67**	**76.01**
DLPL (Sparse DETR)	**28.75**	**46.38**	**30.55**	**65.57**	**79.02**	**76.92**

The fundamental concept is to drive the model to jointly learn perspective-invariant semantic information in the image from multiple distinct viewpoints. Each layer of PIA generates a visual feature containing a new perspective. We cascade multiple layers of PIA and, through repeated perspective-invariant fusion, ultimately enhance the model's robustness to perspective in practical applications (e.g. auto-driving or UAV flights).

3.7. Optimization

The overall loss $\mathcal{L}$ is intuitive: the combination of the main task-specific loss $\mathcal{L}_{task}$ and the reconstruction loss $\mathcal{L}_{rec}$:

$$\mathcal{L} = \mathcal{L}_{task} + \lambda \mathcal{L}_{rec}, \tag{12}$$

where λ is the loss weight and set to 0.4.

4. Experiments

4.1. Scenes, Datasets, Tasks and Implementation Details

DLPL is a general and applicable to a broad range of scenes and tasks. In our experiments, we focus on three types of scenarios (auto-driving, UAV, daily photos) and two fine-grained tasks (segmentation and detection). Furthermore, we concentrate on the more challenging task of segmentation, especially within the UAV scenario due to its rich variety of scene changes. Specifically, given the current lack of a large-scale UAV segmentation benchmark dataset within the field, we propose DroneSeg[3], the largest-scale

[3] We release the proposed DroneSeg dataset at https://github.com/jankyee/DroneSeg.

Table 4. Auto-Driving and Daily photos Segmentation: Comparison with state-of-the-arts on Cityscapes and ADE20K datasets.

Method	**Backbone**	**Pram.**	**mIoU**(%)	
			Cityscapes val	**ADE20K** val
FCN	ResNet-101	68.6M	76.6	41.4
EncNet	ResNet-101	55.1M	76.9	44.7
PSPNet	ResNet-101	68.1M	78.5	44.4
CCNet	ResNet-101	68.9M	80.2	45.2
DeeplabV3+	ResNet-101	62.7M	80.9	44.1
OCRNet	HRNet-W48	70.5M	81.1	45.6
SETR	ViT-Large	318.3M	82.2	50.2
Segmenter	ViT-Large	307.0M	80.7	52.2
SegFormer	MiT-B4	64.1M	83.8	51.1
SegFormer	MiT-B5	84.7M	84.0	51.8
DLPL (SETR)	ViT-Large	321.1M	**83.9**	**53.1**
DLPL (Segmenter)	ViT-Large	309.8M	**82.3**	**53.5**
DLPL (SegFormer)	MiT-B4	66.9M	**85.0**	**53.5**
DLPL (SegFormer)	MiT-B5	87.5M	**85.1**	**53.7**

Table 5. The comparison of DLPL with other perspective-oriented learning methods (data augmentation). Perspective-Vertical and Perspective-Horizontal means adjusting perspectives in vertical and horizontal directions, and are implemented by the default "torchvision.transforms" interfaces in PyTorch (Paszke et al., 2019). For each naive augmentation, we try multiple magnitudes and find the best performance, for fair comparisons.

Perspective-Oriented Learning Method		mIoU (%)
Baseline (SegFormer w/o data augmentation)		52.03
Naive Augmentation	+ Random Rotate	52.94
	+ Random Scale	53.09
	+ Random Perspective-Vertical	53.56
	+ Random Perspective-Horizontal	53.42
	+ Random Combination	55.33
AutoAugmentation (Cubuk et al., 2019)		55.80
DLPL (SegFormer w/o data augmentation)		58.67
DLPL (SegFormer + Random Combination)		59.10
DLPL (SegFormer + AutoAugmentation)		59.09

and semantically richest fine-grained annotated UAV segmentation dataset, to date.

In all experiments, we adopt the MMSegmentation (MMSegmentation, 2020) and MMDetection (Chen et al., 2019) toolboxes as codebases and follow the default basic configurations. In DLPL block, we employ the pretrained and SuperPoint parameters in (DeTone et al., 2018) and froze them in PDD. In our experiments, we incorporate DLPL with various representative ViT segmentors and detectors.

4.2. Comparison with State-of-the-Arts

4.2.1. UAV Scene

Compared to the other two types of scenarios, UAVs present more challenging perspective understanding due to the rich variety of perspective changes caused by real-time movement and deviation during flight. Therefore, segmentation and detection in UAV scenarios become the focus of our experiments. As shown in Tables 2 and 3, we apply DLPL to the currently advanced ViT networks and achieve significant performance improvements in segmentation and detection with less than 3M additional parameters. Specifically, on our newly proposed DroneSeg dataset, which presents the level of challenge to date, we achieve the highest improvement of nearly 5% over four types of plain ViT. In the UAV detection scenario, where objects are relatively sparse, we also achieve an improvement of nearly 3% over various plain ViT detectors. This consistent performance enhancement directly confirms the effectiveness, generality, and robustness of our proposed DLPL.

4.2.2. Auto-Driving and Daily Photos

These two scenes typically also encompass various shooting perspectives. To further validate the universality of DLPL, we conduct experimental verifications within these two types of scenes as well. Specifically, we focused primarily on the segmentation task whose performance is more evidently affected by perspective changes. As shown in Table 4, ViT detectors integrated with DLPL achieve highest performance increases of 3% compared to the plain ViT detectors.

4.3. Ablation Study

We perform ablation studies on semantic segmentation on DroneSeg *test* set, SegFormer is used as baseline network.

4.3.1. The Efficacy of DLPL

As discussed in Section 2, as one type of perspective-oriented learning, we compare DLPL with the various ordinary perspective augmentation techniques. We notice that in all the scenes, the perspective changes are almost always related to the flying height and angles, which can be reflected on the scales, rotations, perspective-vertical and perspective-horizontal augmentations. So, we use the combination of these data augmentation methods for the comparison with DLPL. Extensively, we also include the auto-augmentation techniques (Cubuk et al., 2019) to further support our claims. As shown in Table 5, DLPL shows more potentials than both naive and auto augmentations. We also show that DLPL is still adaptive to the ordinary data augmentation to obtain further improvements.

Table 6. The significance of PDD by comparing it with ordinary perspective representation.

Perspective Representation	mIoU (%)
Ordinary Transformer Network	57.07
Directly using **F**	55.91
PDD	59.10

Table 7. The effectiveness of PHT by comparing with ordinary perspective transformation methods.

Perspective Transformation	mIoU (%)
KNN	56.57
Momentum Update	56.01
Naive GMM	56.94
PHT	59.10

4.3.2. The Significance of PDD

For the first time, PDD provides heuristic inspiration for latent perspective representation. We demonstrate the significance of PDD by comparing with other ordinary perspective representation techniques: (1) directly forcing the perspective learning using an ordinary Transformer network, (2) representing the perspective by directly using $\mathbf{F}$, followed by MLPs. The former lacks of in-depth characterization, and the latter may result in coarse-grained learning. As shown in Table 6, PDD shows undisputed brilliant performance, confirming its rationality and effectiveness.

4.3.3. Effectiveness of PHT

PHT is for perspective transformation over the original perspective $\mathbf{P}$ and perspective embedding space $\mathcal{P}$, in the form of Homography. In order to validate the effectiveness of such choice, we also implement some other perspective transformation methods for comparisons, including K-Nearest Neighbor (KNN), Momentum Update and naive GMM. Results in Table 7 demonstrate that PHT achieves the highest performance.

4.3.4. Visualization

To provide a clearer understanding of perspective transformation, we visualize and compare the feature maps before and after applying DLPL, as depicted in Figure 8. It shows that after passing through DLPL, the perspective reflected in visual feature maps are obviously transformed. For more in-depth insights, we select representative transformation intensities: weak (row 1), medium (row 2), and strong (row 3). These intensities accurately capture the various stages of gradual perspective changes during network training, which is conducive to network training stability and also in line with the practical scene perspective changes such as UAV flights. As the training deepens, the network can eventually learn rich perspective diversity.

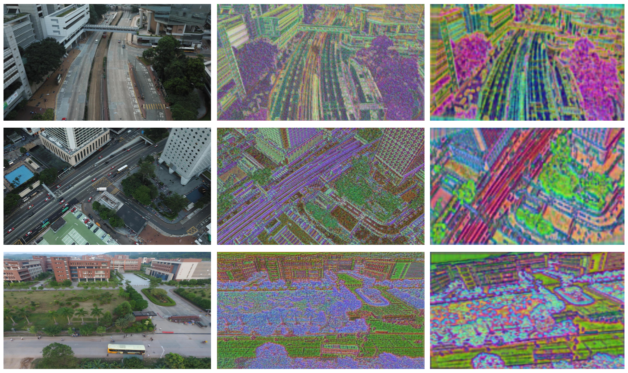
(a) Image
(b) Feature with Original Perspective
(c) Feature with Transformed Perspective

Figure 8. Visualization comparison on feature maps before and after DLPL. We select three representative transformation intensities: weak (row 1), medium (row 2), and strong (row 3). We can find the overall views are obviously transformed after DLPL.

5. Conclusion

We introduce a Discrete Latent Perspective Learning (DLPL) framework that enhances Perspective-Invariant Learning using single-view images. The framework integrates Perspective Discrete Decomposition (PDD), Perspective Homography Transformation (PHT), and Multi-Perspective Fusion via Perspective Invariant Attention (PIA), overcoming the limitations of multi-view data collection and perspective augmentations. Extensive experiments have proven that DLPL can enhance the robustness of neural networks to perspective changes, achieving notable improvements in a wide range of scenes and tasks.

Acknowledgements

This work was supported by the Anhui Provincial Natural Science Foundation under Grant 2108085UD12, the JKW Research Funds under Grant 20-163-14-LZ-001-004-01, the National Key R&D Program of China under Grant 2020AAA0103902, NSFC (No. 62176155), Shanghai Municipal Science and Technology Major Project, China (2021SHZDZX0102).

We acknowledge the support of GPU cluster built by MCC Lab of Information Science and Technology Institution, USTC.

We would like to express our gratitude to Qi Zhu for his brilliant suggestions regarding the writings and figures of this paper, as well as the insightful comments from the Reviewers, Area Chairs, and Program Chairs.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Perspective-transformation-invariant generalized hough transform for perspective planar shape detection and matching. *Pattern Recognition*, 30(3):383–396, 1997. ISSN 0031-3203.

A novel perspective invariant feature transform for rgb-d images. *Computer Vision and Image Understanding*, 167: 109–120, 2018. ISSN 1077-3142.

Bengio, Y., Courville, A., and Vincent, P. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35 (8):1798–1828, 2013.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pp. 213–229, 2020.

Caron, M., Bojanowski, P., Joulin, A., and Douze, M. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 132–149, 2018.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020.

Chen, C., Fan, Q., and Panda, R. Crossvit: Cross-attention multi-scale vision transformer for image classification. *CoRR*, abs/2103.14899, 2021.

Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *European Conference on Computer Vision*, pp. 801–818, 2018a.

Chen, T., Fu, C., Zhu, L., Mao, P., Zhang, J., Zang, Y., and Sun, L. Deep3dsketch: 3d modeling from free-hand sketches with view-and structural-aware adversarial training. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1–5, 2023a.

Chen, T., Zhu, L., Ding, C., Cao, R., Zhang, S., Wang, Y., Li, Z., Sun, L., Mao, P., and Zang, Y. Sam fails to segment anything?–sam-adapter: Adapting sam in underperformed scenes: Camouflage, shadow, and more. *arXiv preprint arXiv:2304.09148*, 2023b.

Chen, T., Ding, C., Zhu, L., Zang, Y., Liao, Y., Li, Z., and Sun, L. Reality3dsketch: Rapid 3d modeling of objects from single freehand sketches. *IEEE Transactions on Multimedia*, 26:4859–4870, 2024a.

Chen, T., Yu, C., Li, J., Zhang, J., Zhu, L., Ji, D., Zhang, Y., Zang, Y., Li, Z., and Sun, L. Reasoning3d – grounding and reasoning in 3d: Fine-grained zero-shot open-vocabulary 3d reasoning part segmentation via large vision-language models. *arXiv preprint arXiv:2405.19326*, 2024b.

Chen, Y., Wang, Y., Lu, P., Chen, Y., and Wang, G. Large-scale structure from motion with semantic constraints of aerial images. In *Chinese Conference on Pattern Recognition and Computer Vision*, pp. 347–359, 2018b.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3213–3223, 2016.

Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., and Le, Q. V. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 113–123, 2019.

DeTone, D., Malisiewicz, T., and Rabinovich, A. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 224–236, 2018.

Dong, J., Jiang, W., Huang, Q., Bao, H., and Zhou, X. Fast and robust multi-person 3d pose estimation from multiple views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7792–7801, 2019.

Elkins, J. Photography theory (art seminar). Dec 2006.

Feng, W., Ji, D., Wang, Y., Chang, S., Ren, H., and Gan, W. Challenges on large scale surveillance video analysis. In *IEEE Conference on Computer Vision and Pattern Recognition workshops*, pp. 69–76, 2018.

Fu, X., Zhang, S., Chen, T., Lu, Y., Zhu, L., Zhou, X., Geiger, A., and Liao, Y. Panoptic nerf: 3d-to-2d label transfer for panoptic urban scene segmentation. In *2022 International Conference on 3D Vision*, pp. 1–11, 2022.

Hu, H., Ji, D., Gan, W., Bai, S., Wu, W., and Yan, J. Class-wise dynamic graph convolution for semantic segmentation. In *European Conference on Computer Vision*, pp. 1–17, 2020.

Ji, D., Lu, H., and Zhang, T. End to end multi-scale convolutional neural network for crowd counting. In *Eleventh International Conference on Machine Vision*, volume 11041, pp. 761–766, 2019.

Ji, D., Wang, H., Hu, H., Gan, W., Wu, W., and Yan, J. Context-aware graph convolution network for target re-identification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 1646–1654, 2021.

Ji, D., Wang, H., Tao, M., Huang, J., Hua, X.-S., and Lu, H. Structural and statistical texture knowledge distillation for semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16876–16885, 2022.

Ji, D., Zhao, F., and Lu, H. Guided patch-grouping wavelet transformer with spatial congruence for ultra-high resolution segmentation. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pp. 920–928, 8 2023a.

Ji, D., Zhao, F., Lu, H., Tao, M., and Ye, J. Ultra-high resolution segmentation with ultra-rich context: A novel benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23621–23630, June 2023b.

Ji, D., Gao, S., Tao, M., Lu, H., and Zhao, F. Changenet: Multi-temporal asymmetric change detection dataset. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2725–2729, 2024a.

Ji, D., Gao, S., Zhu, L., Zhu, Q., Zhao, Y., Xu, P., Lu, H., Ye, J., and Zhao, F. View-centric multi-object tracking with homographic matching in moving uav. *arXiv preprint arXiv:2403.10830*, 2024b.

Jing, L. and Tian, Y. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(11): 4037–4058, 2020.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4015–4026, 2023.

Kriegman, D. Homography estimation. *Computer Vision I, CSE 252A, Winter*, 2007.

Li, Y., Pei, W., and He, Z. Srhen: stepwise-refining homography estimation network via parsing geometric correspondences in deep latent space. In *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 3063–3071, 2020.

Liu, J. and Li, X. Geometrized transformer for self-supervised homography estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9556–9565, 2023.

Lyu, Y., Vosselman, G., Xia, G.-S., Yilmaz, A., and Yang, M. Y. Uavid: A semantic segmentation dataset for uav imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 165:108 – 119, 2020.

Ma, L., Chen, K., Liu, J., and Zhang, J. Homography-driven plane feature matching and pose estimation. In *2021 6th IEEE International Conference on Advanced Robotics and Mechatronics*, pp. 791–796, 2021.

Malis, E. and Vargas Villanueva, M. Deeper understanding of the homography decomposition for vision-based control. 2007.

MMSegmentation. Mmsegmentation: Openmmlab semantic segmentation toolbox and benchmark. *Availabe online: https://github. com/open-mmlab/mmsegmentation*, 2020.

Mumuni, A. and Mumuni, F. Data augmentation: A comprehensive survey of modern approaches. *Array*, 16:100258, 2022. ISSN 2590-0056.

Nigam, I., Huang, C., and Ramanan, D. Ensemble knowledge transfer for semantic segmentation. In *2018 IEEE Winter Conference on Applications of Computer Vision*, pp. 1499–1508, 2018.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32, 2019.

Wang, H., Jiao, L., Liu, F., Li, L., Liu, X., Ji, D., and Gan, W. Ipgn: Interactiveness proposal graph network for human-object interaction detection. *IEEE Transactions on Image Processing*, 30:6583–6593, 2021a.

Wang, H., Jiao, L., Liu, F., Li, L., Liu, X., Ji, D., and Gan, W. Learning social spatio-temporal relation graph in the wild and a video benchmark. *IEEE Transactions on Neural Networks and Learning Systems*, 34(6):2951–2964, 2021b.

Wang, Y., Cheng, J., Chen, Y., Shao, S., Zhu, L., Wu, Z., Liu, T., and Zhu, H. Fvp: Fourier visual prompting for source-free unsupervised domain adaptation of medical image segmentation. *IEEE Transactions on Medical Imaging*, 2023.

Waqas Zamir, S., Arora, A., Gupta, A., Khan, S., Sun, G., Shahbaz Khan, F., Zhu, F., Shao, L., Xia, G.-S., and Bai, X. isaid: A large-scale dataset for instance segmentation in aerial images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 28–37, 2019.

Wu, X., Li, W., Hong, D., Tao, R., and Du, Q. Deep learning for unmanned aerial vehicle-based object detection and tracking: A survey. *IEEE Geoscience and Remote Sensing Magazine*, 10(1):91–124, 2021.

Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., and Luo, P. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34:12077–12090, 2021.

Yağmur, İ. C. and Ates, H. F. Improved homographic adaptation for keypoint generation in cross-spectral registration of thermal and optical imagery. In *Image and Signal Processing for Remote Sensing XXIX*, volume 12733, pp. 67–73, 2023.

Yu, W., Luo, M., Zhou, P., Si, C., Zhou, Y., Wang, X., Feng, J., and Yan, S. Metaformer is actually what you need for vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10819–10829, 2022.

Yu, X., Xu, M., Zhang, Y., Liu, H., Ye, C., Wu, Y., Yan, Z., Zhu, C., Xiong, Z., Liang, T., Chen, G., Cui, S., and Han, X. Mvimgnet: A large-scale dataset of multi-view images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9150–9161, June 2023.

Yuan, Y., Chen, X., and Wang, J. Object-contextual representations for semantic segmentation. In *European Conference on Computer Vision*, pp. 173–190, 2020.

Zeng, R., Denman, S., Sridharan, S., and Fookes, C. Rethinking planar homography estimation using perspective fields. In *Asian Conference on Computer Vision*, pp. 571–586, 2018.

Zhang, Z. and Hanson, A. R. 3d reconstruction based on homography mapping. *Proc. ARPA96*, pp. 1007–1012, 1996.

Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P. H., and Zhang, L. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., and Torralba, A. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 633–641, 2017.

Zhu, L., Ji, D., Zhu, S., Gan, W., Wu, W., and Yan, J. Learning statistical texture for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12537–12546, June 2021a.

Zhu, L., Chen, T., Ji, D., Ye, J., and Liu, J. Llafs: When large-language models meet few-shot segmentation. *arXiv preprint arXiv:2311.16926*, 2023a.

Zhu, L., Chen, T., Yin, J., See, S., and Liu, J. Continual semantic segmentation with automatic memory sample selection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3082–3092, 2023b.

Zhu, L., Chen, T., Yin, J., See, S., and Liu, J. Learning gabor texture features for fine-grained recognition. In *IEEE/CVF International Conference on Computer Vision*, pp. 1621–1631, 2023c.

Zhu, L., Chen, T., Yin, J., See, S., and Liu, J. Addressing background context bias in few-shot segmentation through iterative modulation. In *IEEE/CVF International Conference on Computer Vision*, 2024a.

Zhu, L., Ji, D., Chen, T., Xu, P., Ye, J., and Liu, J. Ibd: Alleviating hallucinations in large vision-language models via image-biased decoding. *arXiv preprint arXiv:2402.18476*, 2024b.

Zhu, P., Wen, L., Du, D., Bian, X., Fan, H., Hu, Q., and Ling, H. Detection and tracking meet drones challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7380–7399, 2021b.