This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3460868

SCIPaD: Incorporating Spatial Clues into Unsupervised Pose-Depth Joint Learning

Yi Feng, Zizhan Guo, Qijun Chen, *Senior Member, IEEE*, and Rui Fan, *Senior Member, IEEE*

***Abstract*—Unsupervised monocular depth estimation frameworks have shown promising performance in autonomous driving. However, existing solutions primarily rely on a simple convolutional neural network for ego-motion recovery, which struggles to estimate precise camera poses in dynamic, complicated real-world scenarios. These inaccurately estimated camera poses can inevitably deteriorate the photometric reconstruction and mislead the depth estimation networks with wrong supervisory signals. In this article, we introduce SCIPaD, a novel approach that incorporates spatial clues for unsupervised depth-pose joint learning. Specifically, a confidence-aware feature flow estimator is proposed to acquire 2D feature positional translations and their associated confidence levels. Meanwhile, we introduce a positional clue aggregator, which integrates pseudo 3D point clouds from DepthNet and 2D feature flows into homogeneous positional representations. Finally, a hierarchical positional embedding injector is proposed to selectively inject spatial clues into semantic features for robust camera pose decoding. Extensive experiments and analyses demonstrate the superior performance of our model compared to other state-of-the-art methods. Remarkably, SCIPaD achieves a reduction of 22.2% in average translation error and 34.8% in average angular error for camera pose estimation task on the KITTI Odometry dataset. Our source code is available at mias.group/SCIPaD.**

***Index Terms*—monocular depth estimation, autonomous driving, convolutional neural network, camera pose estimation**

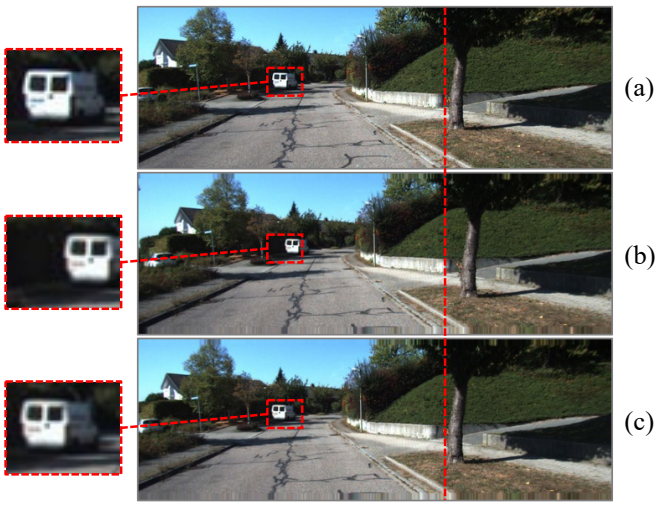

Fig. 1. Photometric reconstruction comparison between SQLdepth [13] and our proposed SCIPaD: (a) the original target frame; (b) the image reconstructed using SQLdepth; (c) the image reconstructed using SCIPaD. Red boxes and vertical lines are added for visual comparison. Our method demonstrates superior performance in camera pose and depth joint estimation.

I. Introduction

AUTONOMOUS vehicles are gradually becoming an integral part of our daily lives [1]. Monocular depth estimation plays a crucial role in the perception systems of autonomous vehicles, as it directly enables agents to perform scene parsing [2]–[4], self-localization [5], and scene reconstruction [6]–[8]. Early works [9], [10] solve the monocular depth estimation problem via supervised learning, which requires precise, extensive depth ground truth, typically acquired using additional cameras or LiDARs [11], [12]. However, it is time-consuming and labor-intensive to gather large-scale depth data from the real world, and the specific data distribution can restrict the generalizability of the network to new, unseen scenarios. To address these limitations, self-supervised methods have emerged as favorable alternatives, which generate supervisory signals from stereo pairs or monocular videos to jointly estimate depth and camera pose (also referred to as ego-motion), thereby eliminating the necessity for ground-truth depth acquisition.

In recent years, self-supervised monocular depth estimation has attracted considerable attention, with ongoing efforts aimed at addressing corner cases in complex and dynamic environments. These self-supervised methods typically utilize a photometric reconstruction loss as the supervisory signal, which can be affected by the precision of both depth predictions and camera pose estimations. Nevertheless, existing frameworks tend to prioritize depth estimation accuracy while often neglecting the accuracy of pose estimation, which is currently generated by a shallow convolutional neural network (CNN)-based PoseNet [14].

The adopted PoseNet architecture suffers from three key limitations: First, the acquisition of ego-motion is strongly related to positional clues and geometrical constraints. Classical visual odometry algorithms [15], [16] utilize the positions of matched keypoints to recover camera poses, leveraging epipolar geometry and perspective-n-point. However, current learning-based frameworks take concatenated RGB frames as input and use encoder-decoder structures for 6-

This research was supported by the Science and Technology Commission of Shanghai Municipal under Grant 22511104500, the National Natural Science Foundation of China under Grant 62233013, the Fundamental Research Funds for the Central Universities, and the Xiaomi Young Talents Program. (*Corresponding author: Rui Fan*)

Yi Feng, Zizhan Guo, Qijun Chen, and Rui Fan are with the Department of Control Science & Engineering, the College of Electronics & Information Engineering, Shanghai Research Institute for Intelligent Autonomous Systems, Shanghai Institute of Intelligent Science and Technology, the State Key Laboratory of Intelligent Autonomous Systems, and Frontiers Science Center for Intelligent Autonomous Systems, Tongji University, Shanghai 201804, China (e-mails: fengyi@ieee.org, {2052110, qjchen}@tongji.edu.cn, rui.fan@ieee.org).

DoF pose regression. Despite their computational efficiency, these structures lack geometrical reasoning and positional encoding abilities. Second, the depth estimation network (hereafter referred to as DepthNet) provides valuable priors of 3D spatial layout. Existing methods fail to utilize the inherent knowledge of spatial layout to enhance the interpretability and precision of ego-motion estimations, further leading to sub-optimal performance in depth-pose joint estimation. Third, the PoseNet backbone is typically pretrained on image classification datasets such as ImageNet [17], which excels in extracting semantic information rather than modeling geometrical correlations [18]. Moreover, these networks typically use the deepest semantic features for camera pose regression, while discarding the shallower ones which could potentially contain informative spatial clues and positional correlations.

To address the aforementioned issues, we first propose a confidence-aware feature flow estimator (**CAFFE**) to calculate and adjust dense feature correspondences with the consideration of pixel-wise confidence levels. This module explicitly extracts abundant positional clues regarding 2D feature translations, which provides strong constraints for ego-motion recovery. Additionally, we argue that the output of DepthNet offers valuable priors of 3D geometrical layout, which may enhance the consistency between camera pose and depth predictions. Therefore, we introduce a positional clue aggregator (**PCA**), which incorporates 2D feature correspondences from CAFFE and 3D spatial layout from DepthNet into a homogeneous positional embedding space. Finally, it is observed that the deeper layers of PoseNet encode richer semantic information, whereas the shallower layers emphasize precise spatial cues. Building upon this insight, we introduce a hierarchical positional embedding injector (**HPEI**), which selectively incorporates positional embeddings into semantic clues with learnable gates. As demonstrated in the experiments, the performance of our model is enhanced by the adaptive integration of both semantic and spatial information.

In conclusion, we propose **SCIPaD**, a **S**patial **C**lue-**I**ncorporated **P**ose **a**nd **D**epth joint learning framework, which integrates all aforementioned innovative components. It demonstrates superior performance in pose and depth joint estimation compared with other state-of-the-art (SoTA) methods, especially in ego-motion recovery. As illustrated in Fig. 1, we compare the photometric reconstruction performance between SQLdepth [13] and our proposed SCIPaD. It can be seen that current SoTA method often fails to produce precise camera poses, which further misleads the photometric consistency-based supervisory signals.

In summary, we make the following contributions:

1) We propose SCIPaD, a novel monocular depth-pose joint learning framework with spatial clues incorporated, achieving SoTA performance across multiple public datasets.
2) We introduce a CAFFE for calculating and reweighting dense feature correspondences, in which a novel 2D soft argmax function is utilized for differentiable dense feature matching.
3) We develop a PCA, which incorporates 2D feature flow and 3D spatial layout into a homogeneous representation of positional clues, ensuring comprehensive geometry encoding and proving highly effective in ego-motion estimation tasks.
4) We propose an HPEI, which selectively injects positional embeddings into semantic clues through a learnable gating mechanism, leading to improved performance in both depth and camera pose estimation.

The remainder of this article is structured as follows. Sect. II presents an overview of the SoTA monocular depth estimation and camera pose estimation methods. In Sect. III, we introduce the proposed SCIPaD framework for depth-pose joint learning. In Sect. IV, we present the experimental results across several public datasets. Sect. V provides a detailed discussion and concludes this article.

II. Related Work

Depth estimation is a fundamental task in computer vision and robotics [19], involving the use of RGB images to generate dense depth predictions. In recent years, monocular depth estimation has garnered significant attention due to its wide applications in autonomous driving [20], [21] and virtual reality [22], [23]. To date, deep learning-based approaches for this task generally fall into two categories: supervised and unsupervised [24].

A. Supervised Monocular Depth Estimation

The supervised learning approach for this task requires pixel-level depth ground truth during the training phase. Eigen *et al.* [9] first introduced a deep learning model with a coarse-to-fine architecture to predict depth maps. Subsequent research has focused on employing more complex network structures [10], [25], [26] or loss functions [27], [28] to enhance depth estimation performance.

Cao *et al.* [25] and Fu *et al.* [28] reformulated the depth regression problem as a classification task, aiming to predict depth ranges rather than exact depth values. Lee *et al.* [29] introduced multi-scale guidance layers to establish the connections between intermediate layer features and the final depth map. Bhat *et al.* [30] proposed AdaBins, which adapts bin sizes based on image content, improving the adaptability of depth prediction. Yang *et al.* [31] developed a Vision Transformer-based architecture to capture long-range correlations in depth estimation. Recently, Depth Anything [32] achieved impressive results by fully unleashing the potential of foundation models. It first reproduced a MiDaS-based [33] teacher model with DINOv2 [34] pretrained weights, and then used these teacher predictions as pseudo labels to train a student model on extensive unlabeled data (62M).

Despite the promising results of these approaches, the requirement for substantial amounts of ground-truth depth labels in supervised training can be costly and limits the widespread adoption of these methods.

B. Unsupervised Monocular Depth Estimation

In the absence of ground-truth depth data, an unsupervised framework aims to train depth estimation models using image

reconstruction as a supervisory signal. Garg *et al.* [35] approached depth estimation as a novel view synthesis problem, minimizing the photometric loss between an input left image and the synthesized right image. Building upon this, Godard *et al.* [36] enhanced accuracy by introducing a left-right disparity consistency loss.

In addition to using stereo pairs, unsupervised methods can learn depth estimation from monocular video frames. Zhou *et al.* [37] developed a network that jointly estimates depth maps and camera poses between sequential frames. To tackle challenges such as dynamic scenes and occlusions, other researchers have explored multi-task learning, which incorporates additional tasks like optical flow estimation [38], [39] and semantic segmentation [40], [41]. Furthermore, some researchers have introduced additional constraints, such as uncertainty estimation [42], [43], to improve the robustness and accuracy of the models. Godard *et al.* [6] proposed Monodepth2, which leverages a minimum reprojection loss to mitigate occlusion issues and an automasking loss to filter out moving objects with velocities similar to the camera. Watson *et al.* [44] introduced ManyDepth, which utilizes multiple frames at test time and leverages geometric constraints through cost volume construction, achieving superior performance.

Despite these advancements, all current unsupervised methods still rely on simple CNN-based PoseNet architectures for camera pose estimation, which often exhibit limited generalizabilities and consequently result in suboptimal depth reconstruction performance.

C. Camera Pose Estimation

Structure from motion (SfM) is widely recognized as a benchmark technique for 3D reconstruction and camera trajectory recovery from videos and image collections. Numerous studies [45]–[48] have endeavored to integrate neural networks into the SfM pipeline, fully leveraging the geometric priors learned from training data. Jau *et al.* [46] designed an end-to-end network for 6-DoF camera pose estimation. However, their method relies on highly accurate ground-truth camera poses, which are often unavailable or difficult to acquire. Li *et al.* [45] proposed UnDeepVO, an unsupervised visual odometry framework, capable of recovering absolute scales using monocular videos and stereo image pairs. Tang *et al.* [49] improved camera pose estimation performance by combining appearance and geometric matching through a differentiable SfM module. Bian *et al.* [50] introduced a geometry consistency loss to enforce the scale-consistent depth learning.

Nonetheless, the pose estimation networks of these unsupervised SoTA methods heavily rely on a basic CNN-based architecture like PoseNet [14]. As shown in Fig. 2, the model processes channel-wise concatenated monocular video frames, which are then passed through several convolutional layers for channel reduction, followed by an average pooling layer to produce a tensor of shape 1×6. This tensor, representing a combination of three Euler angles and three translational components, lacks interpretability for geometric modeling and robustness in scenarios involving moving objects.

To fully exploit the epipolar constraints, we explicitly encode the positional priors and incorporate depth information into camera pose estimation. Our system design and experimental findings provide novel insights that can significantly enhance the performance of both monocular depth estimation and camera pose estimation, providing valuable guidance for researchers in the field.

III. Methodology

A. Architecture Overview

Previous studies such as [13], [51], [52] adopt a CNN-based PoseNet [14] for camera pose estimation. Although the network architecture is effective and lightweight, it struggles to handle dynamic objects in the scene, which can significantly impair the performance of camera pose estimation, and further lead to failures in maintaining photometric consistency constraints. Moreover, the network backbone (usually ResNet-18 [53]) is pretrained for image classification, which excels at extracting semantic cues from images. However, camera pose estimation requires the utilization of spatial and geometrical information, including 2D positional translations and 3D point cloud layouts, which have not been considered in the current SoTA frameworks. SCIPaD is thus proposed to solve the problems aforementioned.

As illustrated in Fig. 2, for the input reference and target frames $\boldsymbol{I}^r, \boldsymbol{I}^t \in \mathbb{R}^{3\times H\times W}$, meaningful semantic information is first extracted from two separate branches: one branch processes the channel-wise concatenated $\boldsymbol{I}^r$ and $\boldsymbol{I}^t$ to explore the implicit correlations between the semantics and estimated camera pose, producing a hierarchical semantic feature set $\mathcal{F}^s = \{\boldsymbol{F}^s_1, ..., \boldsymbol{F}^s_k\}$, where $\boldsymbol{F}^s_k \in \mathbb{R}^{c\times \frac{H}{2^k}\times \frac{W}{2^k}}$ represents the semantic features at the k-th stage. The other branch, which remains frozen, processes the batch-wise concatenated frames and separates them to produce translation-equivariant feature sets $\mathcal{F}^r = \{\boldsymbol{F}^r_1, ..., \boldsymbol{F}^r_k\}$ and $\mathcal{F}^t = \{\boldsymbol{F}^t_1, ..., \boldsymbol{F}^t_k\}$ of $\boldsymbol{I}^r$ and $\boldsymbol{I}^t$, respectively. Subsequently, confidence-aware feature flow is acquired by calculating feature affinity using a differentiable 2D soft argmax function, which is then integrated with the 3D point cloud data obtained from DepthNet to derive positional features $\mathcal{F}^p = \{\boldsymbol{F}^p_1, ..., \boldsymbol{F}^p_k\}$. Finally, the embedded positional features are hierarchically injected into the semantic features for 6-DoF camera pose regression.

The following subsections detail the confidence-aware feature flow estimator, the spatial clue aggregator, and the hierarchical positional embedding injector within the SCIPaD framework.

B. Confidence-Aware Feature Flow Estimator

As illustrated in Fig. 3, we take features $\boldsymbol{F}^t_i, \boldsymbol{F}^r_i \in \mathbb{R}^{c\times h\times w}$ from the target and reference frames at stage i as an example, where c is the feature channel dimension, and $(h, w) = \frac{(H,W)}{2^i}$ is the exponentially decreased feature resolution. We assume that the temporal intervals between successive frames are sufficiently short to ensure a relative constancy in object dimensions and to preserve the brightness constancy assumption [54] across corresponding points. Meanwhile, CNN-based models use a set of convolution kernels with shared weights for feature extraction, which exhibits a strong inductive bias of translation equivariance. Leveraging this property, we aim to

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3460868

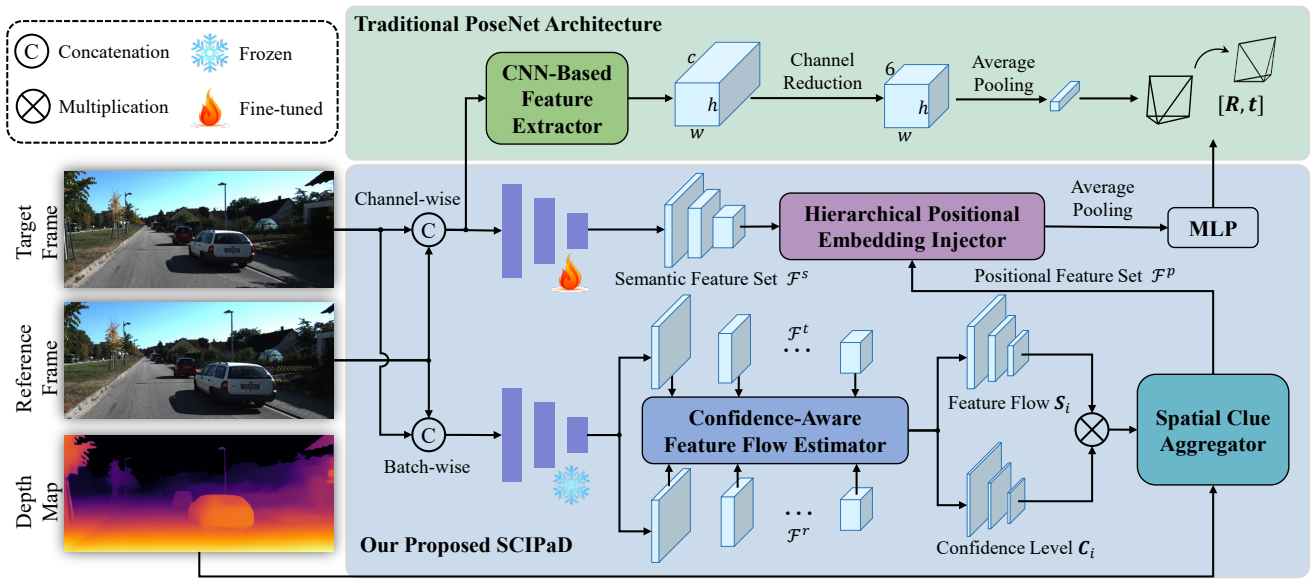

Fig. 2. An illustration of our proposed SCIPaD framework. Compared with the traditional PoseNet [14] architecture, it comprises three main parts: (1) a confidence-aware feature flow estimator, (2) a spatial clue aggregator, and (3) a hierarchical positional embedding injector.

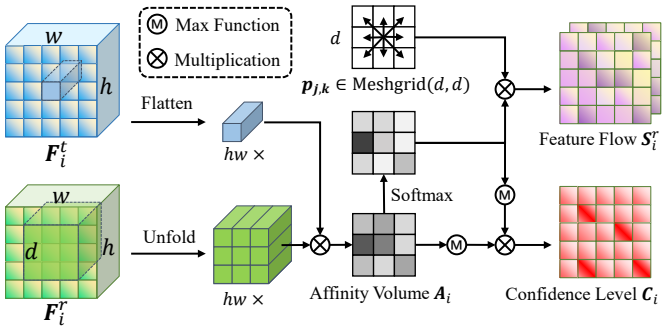

Fig. 3. An illustration of our proposed confidence-aware feature flow estimator. It produces feature flow $\boldsymbol{S}_i^r$ and its confidence $\boldsymbol{C}_i$ through affinity volume construction and a differentiable 2D soft argmax function.

find feature correspondences between the reference and target frames for explicit geometrical encoding.

Unlike previous work [55], which primarily emphasizes feature flow generation across consecutive frames, our proposed CAFFE also produces pixel-wise confidence levels for reweighting the feature flow. We first normalize $\boldsymbol{F}_i^r$ to have a unit length along the channel dimension, and then extract sliding local blocks from the normalized features with window size d, resulting in the unfolded reference features $\tilde{\boldsymbol{F}}_i^r \in \mathbb{R}^{h\times w\times c\times d^2}$ as follows:

$$\tilde{\boldsymbol{F}}_i^r = \text{Reshape}(\text{Unfold}(\frac{\boldsymbol{F}_i^r}{||\boldsymbol{F}_i^r||_2})). \tag{1}$$

Similarly, the target features $\boldsymbol{F}_i^t$ are processed to form $\tilde{\boldsymbol{F}}_i^t \in \mathbb{R}^{h\times w\times 1\times c}$. Subsequently, the cross-frame feature affinity $\boldsymbol{A}_i \in \mathbb{R}^{h\times w\times d\times d}$ in stage i is calculated as follows:

$$\boldsymbol{A}_i = \text{Reshape}(\tilde{\boldsymbol{F}}_i^t \tilde{\boldsymbol{F}}_i^r). \tag{2}$$

The affinity volumes capture correspondences and their confidence levels between features from the two input frames. Specifically, a higher affinity value indicates a stronger resemblance between a pixel in the target frame and another pixel within the selected window of the reference frame, while a lower value suggests a mismatch or lower confidence in the correspondence. Hence, in order to determine the relative feature position displacements, *i.e.*, feature flow $\boldsymbol{S}_i^r \in \mathbb{R}^{h\times w\times 2}$, a straightforward way to localize the matched features is taking the position arguments of the maxima as follows:

$$\boldsymbol{S}_i^r = \arg\max_{\boldsymbol{p}\in\mathcal{W}} \boldsymbol{A}_i(:,\boldsymbol{p}), \tag{3}$$

where $\mathcal{W} = \{\boldsymbol{p} = [j,k]^\top \mid j,k \in [0,d] \cap \mathbb{Z}\}$ represents the set of pixels in the specified window partition. However, the argmax function is non-differentiable and generates discrete outputs, which prevents the network from backpropagation and introduces quantization errors. To address this issue, we draw inspiration from the smooth approximation proposed in [56], and introduce a 2D soft argmax as a substitute for the original argmax function:

$$\boldsymbol{S}_i^r = \sum_{\boldsymbol{p}\in\mathcal{W}} \boldsymbol{p} \frac{\exp(\boldsymbol{A}_i(:,\boldsymbol{p}))}{\sum_{\boldsymbol{p}} \exp(\boldsymbol{A}_i(:,\boldsymbol{p}))}. \tag{4}$$

In this way, the position with the maximum likelihood is calculated using a probability-weighted sum of the position enumerations $\boldsymbol{p}$, where the probabilities are normalized through the softmax of the affinity values. This 2D soft argmax approach enhances feature matching with sub-pixel accuracy, facilitating the flow of gradients from pose estimation back through the point coordinates.

Another crucial piece of information conveyed by $\boldsymbol{A}_i$ is the confidence level $\boldsymbol{C}_i \in \mathbb{R}^{h\times w\times 1}$, which indicates the quality of the calculated feature flow. We argue that $\boldsymbol{C}_i$ depends on two factors:

- **Magnitude of affinity values**. If all the affinity values are relatively small, it suggests a lack of strong feature

correspondences within the specified window. For example, if a moving object occupies the entire window and occludes the original matched pixel, this can result in smaller affinity values in the entire window.

- **Distribution of affinity values**. If the largest affinity values are closely clustered, it suggests the presence of texture-less areas or keypoints that are difficult to discriminate.

To avoid these aforementioned issues and lower their impact on matched correspondences, we formulate the feature matching confidence level $\boldsymbol{C}_i$ as follows:

$$\boldsymbol{C}_i = \max_{\boldsymbol{p}\in\mathcal{W}} \boldsymbol{A}_i(:,\boldsymbol{p}) \cdot \max_{\boldsymbol{p}\in\mathcal{W}} \frac{\exp(\boldsymbol{A}_i(:,\boldsymbol{p}))}{\sum_{\boldsymbol{p}} \exp(\boldsymbol{A}_i(:,\boldsymbol{p}))}, \tag{5}$$

where $\boldsymbol{C}_i$ tends to approach 1 only when there is a unique large affinity value within the given window, indicating high confidence in the feature correspondence. This formulation assists in assessing the reliability of feature matches by considering both the magnitude and the distribution of affinity values across spatial dimensions.

C. Positional Clue Aggregator

To ensure robust and effective camera pose estimation, it is essential to incorporate two primary positional clues. The first involves 2D feature flow and its corresponding pixel coordinates, which reflect the pixel-wise geometrical constraints with respect to cross-frame correlations. To fully exploit spatial information and positional clues, we propose a positional clue aggregator, which incorporates these elements into a compact, homogeneous positional embedding space. In the i-th stage, the absolute feature position $\boldsymbol{S}_i^a \in \mathbb{R}^{h\times w\times 2}$ can be straightforwardly obtained as follows:

$$\boldsymbol{S}_i^a = \mathrm{Concat}(\mathrm{Meshgrid}(h,w)), \tag{6}$$

where the $\mathrm{Meshgrid}$ function generates 2D grid coordinates using matrix indexing. Meanwhile, the dense point cloud $\boldsymbol{P}_i^c \in \mathbb{R}^{h\times w\times 3}$ in the i-th stage can be easily obtained from the perspective camera model. For an arbitrary 3D point $\boldsymbol{p}^c = [x^c, y^c, z^c]^\top$ in camera coordinates, its corresponding homogeneous pixel coordinates $\tilde{\boldsymbol{p}} = [u, v, 1]^\top$ satisfy the following relationship:

$$\boldsymbol{p}^c = z^c \boldsymbol{K}^{-1} \tilde{\boldsymbol{p}}, \tag{7}$$

where $\boldsymbol{K}$ represents the camera intrinsic matrix, and z^c can be approximated using predictions from DepthNet. Thus, $\boldsymbol{P}_i^c$ can be generated by iterating through pixel coordinates.

Having obtained the feature flow $\boldsymbol{S}_i^r$, absolute feature position $\boldsymbol{S}_i^a$, their corresponding confidence $\boldsymbol{C}_i$, and the downsampled dense point cloud $\boldsymbol{P}_i^c$, we proceed to encode them into a homogeneous position embedding space $\boldsymbol{F}_i^p$. First, we normalize $\boldsymbol{S}_i^r$, $\boldsymbol{S}_i^a$ and $\boldsymbol{P}_i^c$ into the range $[0, 1]$ using linear mapping, facilitating a uniform feature representation across different scales. Subsequently, these three positional priors are integrated into positional embeddings $\boldsymbol{F}_i^p$ as follows:

$$\boldsymbol{F}_i^p = \boldsymbol{C}_i(f(\tilde{\boldsymbol{S}}_i^r, \boldsymbol{\Theta}_i^s) + f(\tilde{\boldsymbol{S}}_i^a, \boldsymbol{\Theta}_i^s)) + f(\tilde{\boldsymbol{P}}_i, \boldsymbol{\Theta}_i^p), \tag{8}$$

where $f(\cdot, \boldsymbol{\Theta}_i)$ represents two consecutive convolutional layers with learnable parameters $\boldsymbol{\Theta}_i$ that map a 2D or 3D position vector into a higher embedding dimension. Notably, $\boldsymbol{\Theta}_i^s$ is shared between $\boldsymbol{S}_i^r$ and $\boldsymbol{S}_i^a$ to maintain positional encoding consistency. This approach effectively integrates spatial context from feature flow and 3D point cloud representations, enhancing the accuracy and robustness of camera pose estimation by leveraging comprehensive positional clues.

D. Hierarchical Positional Embedding Injector

It has been demonstrated that multi-scale feature aggregation across different modalities improves the capacity of deep neural networks in various computer vision tasks [67]. However, as a task heavily reliant on geometric properties, camera pose estimation requires not only substantial semantic information but also accurate geometric cues. It is crucial to achieve a sensible balance between these features across different scales. This is due to the fact that the shallower layers of the network tend to contain less semantic details but richer geometric representations, whereas the deeper layers excel in capturing refined semantic abstractions but may deteriorate the meaningful spatial clues due to downsampling operations.

In this work, our proposed hierarchical positional embedding injector aims to effectively integrate low-level positional embeddings $\mathcal{F}^p$ into high-level semantic features $\mathcal{F}^s$ across different scales. Unlike previous works, such as [13], [58], which utilize the deepest features for camera pose decoding, we hierarchically aggregate fused semantic and positional features at multiple resolutions to preserve both high-level semantic and low-level positional information. For the features $\boldsymbol{F}_i^s \in \mathcal{F}^s$ and $\boldsymbol{F}_i^p \in \mathcal{F}^p$ from the i-th stage, we first employ a channel reduction block to transform $\boldsymbol{F}_i^p$ into compact embeddings. Subsequently, the compressed positional embeddings are integrated into the semantic features $\boldsymbol{F}_i^s$ with a learnable gate γ_i, which automatically modulates the importance of semantic and spatial information.

The motivation for introducing the gating mechanism lies in leveraging the strengths of different network layers: the shallower layers of the network encode more precise positional embeddings, while the deeper layers preserve richer semantic information. In contrast to prior arts [67] which indiscriminately fuse the cross-modal information, our approach ensures the network adaptively focuses on semantic and positional information with different scales. Afterwards, the selectively fused features are combined with those from the preceding layer, yielding spatial-semantic co-attentive feature representations. These operations can be written as follows:

$$\boldsymbol{F}_i = \begin{cases} f_i(\gamma_i f_c(\boldsymbol{F}_i^p) + (1-\gamma_i)\boldsymbol{F}_i^s + \boldsymbol{F}_{i-1}, \boldsymbol{\Theta}_i), & \text{if } 1 \le i < k \\ \gamma_i f_c(\boldsymbol{F}_i^p) + (1-\gamma_i)\boldsymbol{F}_i^s + \boldsymbol{F}_{i-1}, & \text{if } i = k \end{cases} \tag{9}$$

where k denotes the number of stages in the backbone, γ_i regulates the importance between semantic features $\boldsymbol{F}_i^s$ and positional features $\boldsymbol{F}_i^p$, $f_i(\cdot, \boldsymbol{\Theta_i})$ represents a combination of convolutional, rectified linear unit (ReLU), and downsampling layers with learnable parameters $\boldsymbol{\Theta}_i$, and $f_c(\cdot)$ denotes the channel reduction function performed by a convolutional layer with a kernel size of 1×1. $\boldsymbol{F}_0$ is initialized as a zero matrix

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3460868

TABLE I
QUANTITATIVE COMPARISON ON THE KITTI EIGEN BENCHMARK [57]. IN THE TRAIN COLUMN, **S** DENOTES TRAINING WITH SYNCHRONIZED STEREO IMAGE PAIRS, **M** DENOTES TRAINING WITH MONOCULAR SEQUENCES, AND **MS** DENOTES TRAINING WITH BOTH MONOCULAR SEQUENCES AND STEREO PAIRS. THE BEST RESULTS ARE SHOWN IN BOLD TYPE.

Method	training	Resolution	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$
SC-DepthV3 [58]	M	256 × 832	0.118	0.756	4.709	0.188	0.864	0.960	0.984
Monodepth2 [6]	MS	192 × 640	0.106	0.818	4.750	0.196	0.874	0.957	0.979
HR-Depth [59]	MS	192 × 640	0.107	0.785	4.612	0.185	0.887	0.962	0.982
PackNet-SfM [21]	M	192 × 640	0.111	0.785	4.601	0.189	0.878	0.960	0.982
DIFFNet [60]	M	192 × 640	0.102	0.764	4.483	0.180	0.890	0.964	0.983
MonoViT-tiny [61]	M	192 × 640	0.102	0.733	4.459	0.177	0.895	0.965	0.984
Swin-Depth [62]	M	192 × 640	0.106	0.739	4.510	0.182	0.890	0.964	0.984
Lite-Mono [63]	M	192 × 640	0.107	0.765	4.561	0.183	0.886	0.963	0.983
Lite-Mono-8M [63]	M	192 × 640	0.101	0.729	4.454	0.178	0.897	0.965	0.983
Dynamo-Depth [64]	M	192 × 640	0.112	0.758	4.505	0.183	0.873	0.959	0.984
ManyDepth [44]	M	192 × 640	0.098	0.770	4.459	0.176	0.900	0.965	0.983
DynamicDepth [65]	M	192 × 640	0.096	0.720	4.458	0.175	0.897	0.964	0.984
TriDepth [40]	M	192 × 640	0.093	0.665	4.272	0.172	0.907	0.967	0.984
SQLdepth [13]	M	192 × 640	0.091	0.713	4.204	0.169	0.914	0.968	0.984
SCIPaD (Ours)	M	192 × 640	**0.090**	**0.650**	**4.056**	**0.166**	**0.918**	**0.970**	**0.985**
Monodepth2 [6]	MS	320 × 1024	0.106	0.806	4.630	0.193	0.876	0.958	0.980
HR-Depth [59]	MS	320 × 1024	0.101	0.716	4.395	0.179	0.899	0.966	0.983
DIFFNet [60]	M	320 × 1024	0.097	0.722	4.435	0.174	0.907	0.967	0.984
MonoViT-tiny [61]	M	320 × 1024	0.096	0.714	4.292	0.172	0.908	0.968	0.984
Lite-Mono-8M [63]	M	320 × 1024	0.097	0.710	4.309	0.174	0.905	0.967	0.984
ManyDepth [44]	M	320 × 1024	0.087	0.685	4.142	0.167	0.920	0.968	0.983
SQLdepth [13]	M	320 × 1024	0.087	0.659	4.096	0.165	0.920	0.970	0.984
SCIPaD (Ours)	M	320 × 1024	**0.086**	**0.636**	**4.006**	**0.165**	**0.922**	**0.968**	**0.984**

TABLE II
QUANTITATVE COMPARISON USING THE IMPROVED KITTI GROUND TRUTH PROVIDED IN [66].

Method	training	Resolution	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$
Monodepth2 [6]	MS	192 × 640	0.080	0.466	3.681	0.127	0.926	0.985	0.995
PackNet-SfM [21]	M	192 × 640	0.078	0.420	3.485	0.121	0.931	0.986	0.996
ManyDepth [44]	M	192 × 640	0.070	0.399	3.455	0.113	0.941	0.989	0.997
DynamicDepth [65]	M	192 × 640	0.068	0.362	3.454	0.111	0.943	0.991	0.998
TriDepth [40]	M	192 × 640	0.068	0.359	3.341	0.110	0.944	0.989	0.997
SQLdepth [13]	M	192 × 640	0.061	0.317	3.055	0.100	0.957	0.992	0.997
SCIPaD (Ours)	M	192 × 640	**0.059**	**0.287**	**2.871**	**0.095**	**0.964**	**0.993**	**0.998**

with the same shape of $\boldsymbol{F}_1^s$, and the final aggregated output $\boldsymbol{F}_4$ is subsequently passed through a multilayer perceptron (MLP) for camera pose decoding.

IV. EXPERIMENTS

A. Datasets, Evaluation Metrics, and Implementation Details

Our proposed method is evaluated on three public datasets, including the KITTI Raw dataset [68], the KITTI Odometry dataset [57], and the Make3D dataset [69]. Specifically, we evaluate the depth estimation performance on the KITTI Raw dataset, and evaluate the camera pose estimation results on the KITTI Odometry dataset. Moreover, we also assess the generalizability of our model on the Make3D dataset using the weights pretrained on the KITTI Raw dataset.

- **KITTI Raw** [68]: This dataset consists of driving videos recorded in urban environments, and it is widely used in self-supervised monocular depth estimation research. Following previous works [13], [37], [58], [70], we adopt the Eigen split, which uses 39,810 images for training, 4,424 images for evaluation, and 697 images for testing. Depth ranges are capped at 80 m, and all images are resized to the resolution of 192×640 pixels or 320×1024 pixels for network training.
- **KITTI Odometry** [57]: This dataset contains well-rectified stereo images with ground-truth trajectories in 22 driving scenarios. Following previous work [6], we use Seqs. 00-08 for model training and test our method on Seqs. 09-10. The absolute trajectory error is calculated

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3460868

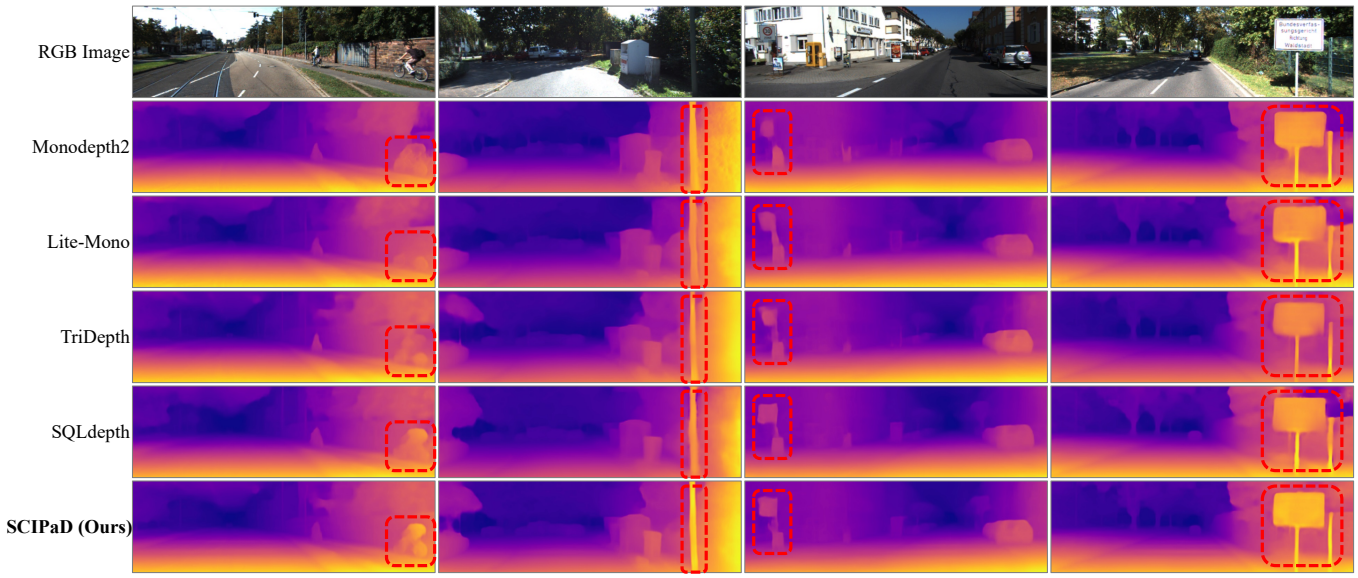

Fig. 4. Qualitative experimental results on the KITTI Eigen benchmark. The regions highlighted in the red boxes illustrate that our method produces locally consistent depth maps with enhanced details.

TABLE III
VISUAL ODOMETRY RESULTS ON THE KITTI ODOMETRY DATASET [57].

Method	Seq. 09			Seq. 10		
	e_t (%)	e_r (%)	ATE (m)	e_t (%)	e_r (%)	ATE (m)
SfMLearner [37]	19.15	6.82	77.79	40.40	17.69	67.34
GeoNet [38]	28.72	9.80	158.45	23.90	9.00	43.04
DeepMatchVO [71]	9.91	3.80	27.08	12.18	5.90	24.44
Monodepth2 [6]	36.70	16.36	99.14	49.71	25.08	86.94
SC-Depth [50]	12.16	4.01	58.79	12.23	6.20	16.42
SCIPaD (Ours)	**7.43**	**2.46**	**26.15**	**9.82**	**3.87**	**15.51**

by averaging over all overlapping five-frame snippets in the test sequences, following the approach proposed in [6].

- **Make3D** [69]: We evaluate the generalizability of the proposed method on the Make3D dataset, which contains 134 test images of outdoor scenes. The proposed model, initially trained on the KITTI Raw dataset, is directly applied to these test images for evaluation.

Adhering to the experiments presented in the previous works [6], [37], we quantify the model's performance using the mean absolute relative error (Abs Rel), the mean squared relative error (Sq Rel), the root mean squared error (RMSE), the root mean squared log error (RMSE log), and the accuracy under threshold ($\delta_i < 1.25^i$, $i = 1, 2, 3$). Detailed definitions of these metrics can be found in [9]. For visual odometry performance evaluation, we follow the standard evaluation metrics introduced in [57], including the average translational error e_t (%), the average rotational error e_r (%), and the absolute trajectory error ATE (m) [72].

The proposed method is implemented in PyTorch and trained on an NVIDIA RTX 4090 GPU. We adopt the TriDepth [40] framework as our baseline. Following [6], we utilize a snippet of three sequential video frames as a training sample. During training, images are augmented with random color jitter and horizontal flips. We employ the Adam optimizer [73] and conduct the training over 30 epochs, starting with a learning rate of 10^{-4}, which we reduce by a factor of 10 for the final 5 epochs. In line with practices from [6], [13], [44], we initialize the encoder of our network using pretrained weights from ImageNet [17].

B. Comparison with State-of-The-Arts

Depth Estimation Results. As presented in Table I, our proposed SCIPaD achieves SoTA performance on both resolutions on the KITTI Raw dataset. It can be observed that SCIPaD demonstrates superior performance compared to all existing self-supervised methods, achieving an 8.8% reduction in Sq Rel, and a 0.4% improvement in δ_1 compared to SQLdepth [13], a previous SoTA approach. Moreover, SCIPaD significantly outperforms its counterparts trained with additional stereo image pairs such as Monodepth2 [6] and HR-Depth [59], achieving an average error reduction of 15.5% in Abs Rel and an average performance gain of 1.1% in δ_1.

Fig. 4 shows the qualitative experimental results on the KITTI Eigen benchmark. Compared with previous SoTA methods, which often produce blurry edges in the foreground, our method generates more distinctive boundaries (*e.g.*, the first and third columns) and maintains better depth consistency in continuous regions (*e.g.*, the second and fourth columns). This superior performance is primarily due to our method's ability to determine accurate and robust camera poses, which in turn generate more refined self-supervisory cues for photometric reconstruction.

Due to the limited quality of the original ground-truth data in KITTI, we additionally present evaluation results using the improved KITTI ground truth [66] in Table II. Even when compared with the previous SoTA method SQLdepth [13],

This article has been accepted for publication in IEEE Transactions on Intelligent Vehicles. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIV.2024.3460868

TABLE IV
COMPARISONS OF EXISTING MODELS WITH AND WITHOUT OUR PROPOSED POSENET EMBEDDED ON THE KITTI EIGEN BENCHMARK. MANUAL REPLICATIONS WITH RELEASED CODE ARE INDICATED BY THE SYMBOL †.

Method	with Ours	Resolution (Pixels)	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	$\delta < 1.25 \uparrow$	$\delta < 1.25^2 \uparrow$	$\delta < 1.25^3 \uparrow$
SC-DepthV3† [58]		256 × 832	0.119	0.756	4.699	0.188	0.863	0.960	0.984
	✓	256 × 832	0.113	0.742	4.603	0.185	0.870	0.962	0.984
Monodepth2† [6]		192 × 640	0.116	0.923	4.856	0.193	0.878	0.959	0.981
	✓	192 × 640	0.115	0.902	4.827	0.192	0.877	0.959	0.981
ManyDepth† [44]		192 × 640	0.101	0.800	4.583	0.182	0.894	0.962	0.982
	✓	192 × 640	0.098	0.758	4.430	0.177	0.899	0.964	0.983
TriDepth† [40]		192 × 640	0.095	0.718	4.408	0.176	0.896	0.964	0.983
	✓	192 × 640	0.094	0.699	4.325	0.167	0.903	0.969	0.984

TABLE V
ABLATION STUDIES OF SCIPAD DESIGN ON THE KITTI RAW AND KITTI ODOMETRY DATASETS.

Configuration		KITTI Raw Dataset				KITTI Odometry Dataset					
						Seq. 09			Seq. 10		
		Abs Rel ↓	$\delta_1 \uparrow$	$\delta_2 \uparrow$	$\delta_3 \uparrow$	e_t (%)	e_r (%)	ATE (m)	e_t (%)	e_r (%)	ATE (m)
Full Implementation		**0.090**	0.918	**0.970**	**0.985**	7.43	2.46	26.15	**9.82**	**3.87**	**15.51**
CAFFE	Unfrozen Backbone	0.090	**0.919**	0.969	0.985	**7.37**	**2.10**	**24.84**	12.53	4.84	19.26
	w/o Confidence Reweighting	0.092	0.912	0.967	0.984	10.92	3.03	45.61	12.49	6.31	25.83
	w/o Feature Normalization	0.091	0.917	0.968	0.984	9.84	2.89	41.84	12.97	6.94	27.42
PCA	w/o Feature Flow	0.095	0.907	0.965	0.984	15.38	5.26	60.05	23.39	9.55	43.29
	w/o Absolute Feature Position	0.091	0.914	0.968	0.984	10.01	2.98	41.84	12.97	6.94	27.42
	w/o Depth Predictions	0.092	0.913	0.967	0.984	10.52	3.12	46.51	13.55	6.72	28.81
	w/ Shared Embedding Layer	0.094	0.903	0.965	0.984	16.84	3.51	55.36	18.45	8.36	39.82
HPEI	w/o Gating Mechanism	0.092	0.913	0.967	0.984	10.42	3.09	46.85	13.75	6.54	26.88
	Decode $\mathcal{F}^s$ Only	0.093	0.907	0.967	0.984	15.24	3.49	52.23	15.42	7.23	30.14
	Decode $\mathcal{F}^p$ Only	0.096	0.897	0.964	0.983	12.14	4.52	52.83	12.41	6.33	35.64

SCIPaD continues to demonstrate superior performance across all metrics. Notably, it achieves a 3.3% reduction in Abs Rel and a 9.5% reduction in Sq Rel at a resolution of 640×192 pixels.

Camera Pose Estimation Results. We use the KITTI Odometry dataset [57] to evaluate the camera pose estimation performance of our proposed SCIPaD. Models trained on monocular videos often struggle to recover absolute depth metrics due to scale ambiguity. To address this issue, we align the scale of their predicted results with the ground truth using 7-DoF optimization. As presented in Table III, SCIPaD significantly outperforms other monocular visual odometry methods. Compared to the previous SoTA frameworks, our proposed method achieves a reduction of 22.2% in e_t, 34.8% in e_r, and 4.5% in ATE, respectively.

We also provide a qualitative comparison of the trajectories produced by different methods. As shown in Fig. 6, we evaluate monocular visual odometry methods on Seq. 09 (left) and Seq. 10 (right) of the KITTI Odometry dataset, and SCIPaD exhibits superior results, with minimal drift among all SoTA methods. This demonstrates the high performance and robustness of our method in ego-motion recovery and long-term trajectory estimation.

C. Ablation Study

As shown in Table IV, we incorporate the PoseNet in SCIPaD into existing open-source methods and demonstrate that our method significantly enhances their depth estimation performances. Remarkably, this integration results in a substantial performance boost across all metrics for the original models.

Furthermore, we investigate the rationality and efficacy of our proposed SCIPaD. As illustrated in Table V, we conduct ablation experiments regarding the inner design of CAFFE, PCA, and HPEI. First, we observe that unfreezing the feature flow backbone results in minor performance gains but significantly increases the computational burden. Therefore, we opt to maintain a frozen feature flow to achieve a balance between accuracy and computational efficiency. Moreover, we notice that the removal of feature normalization in (1) and confidence reweighting in (8) leads to reduced performance across the two datasets, confirming the necessity of these components. Second, feature flow, absolute feature position, and DepthNet predictions are three spatial clues to be aggregated. Removing these elements sequentially highlighted that feature flow is most critical for depth estimation and ego-motion recovery, while the absolute feature position contributes the least. Including DepthNet predictions enhances

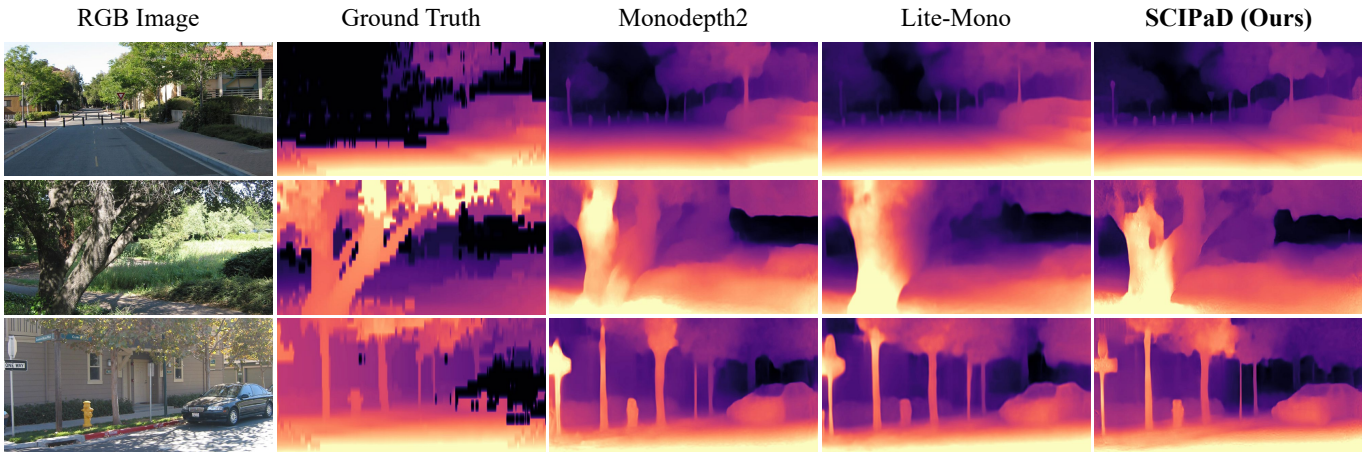

Fig. 5. Qualitative zero-shot results on the Make3D dataset [69].

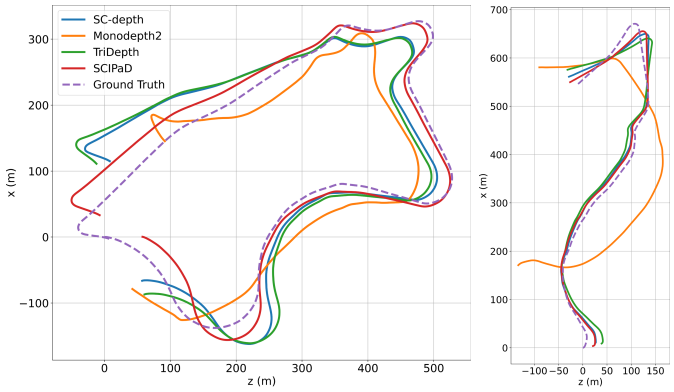

Fig. 6. Comparison of the estimated trajectories using Seqs. 09 and 10 on the KITTI Odometry dataset [57]. All predictions are rescaled to align with the ground truth for a fair comparison.

pose estimation significantly. Using the same embedding layer for aggregating all three spatial clues, as noted in (8), led to a noticeable performance decline, suggesting the need for distinct processing of each clue. Third, we evaluate the efficacy of the gating mechanism in (9) as well as the overall contributions of semantic features $\mathcal{F}^s$ and positional embeddings $\mathcal{F}^p$. The results indicate that both components are crucial to the system's performance, significantly impacting the depth-pose joint estimation results.

D. Zero-Shot Performance Evaluation

To further evaluate the generalizability of SCIPaD, we conduct a zero-shot test on the Make3D dataset [69] using the pretrained weights obtained from the KITTI dataset. Following the evaluation settings used in [13], the test images are center-cropped to a 2:1 ratio for a fair comparison. As presented in Table VI and Fig. 5, SCIPaD outperforms other methods in zero-shot performance, and produces finer-grained depth maps with more accurate scene details. These results demonstrate the exceptional zero-shot generalizability of our model.

TABLE VI
QUANTITATIVE ZERO-SHOT PERFORMANCE COMPARISON ON THE MAKE3D DATASET [69].

Method	Abs Rel	Sq Rel	RMSE	RMSE log
Monodepth2 [6]	0.321	3.378	7.252	0.163
HR-Depth [59]	0.305	2.944	6.857	0.157
CADepth [74]	0.319	3.564	7.152	0.158
DIFFNet [60]	0.298	2.901	6.753	0.153
MonoViT [61]	0.286	2.758	6.623	0.147
SCIPaD (Ours)	0.284	2.712	6.593	0.147

V. CONCLUSION

In this article, we introduced SCIPaD, a novel architecture designed for unsupervised learning of ego-motion and monocular depth estimation. SCIPaD estimates confidence-aware feature flow from a CAFFE, and aggregates spatial clues into homogeneous positional representations using a PCA. Finally, an HPEI selectively injects positional embeddings into semantic information for robust ego-motion decoding. Our proposed method achieves remarkable state-of-the-art results and improved generalizability on the KITTI Raw, KITTI Odometry, and Make3D datasets. Future work will focus on developing a lightweight version of SCIPaD and further enhancing the generalizability performance of the depth-pose joint learning framework.

REFERENCES

[1] J. Li *et al.*, "RoadFormer: Duplex Transformer for RGB-Normal Semantic Road Scene Parsing," *IEEE Transactions on Intelligent Vehicles*, 2024, DOI: 10.1109/TIV.2024.3388726.
[2] R. Fan *et al.*, "SNE-RoadSeg: Incorporating Surface Normal Information into Semantic Segmentation for Accurate Freespace Detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2020, pp. 340–356.
[3] R. Fan and M. Liu, "Road damage detection based on unsupervised disparity map segmentation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 11, pp. 4906–4911, 2020.
[4] R. Fan *et al.*, "Graph attention layer evolves semantic segmentation for road pothole detection: A benchmark and algorithms," *IEEE Transactions on Image Processing*, vol. 30, pp. 8144–8154, 2021.

[5] Y. Feng *et al.*, "Freespace Optical Flow Modeling for Automated Driving," *IEEE/ASME Transactions on Mechatronics*, vol. 29, no. 2, pp. 1511–1520, 2024.

[6] C. Godard *et al.*, "Digging into Self-Supervised Monocular Depth Estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 3828–3838.

[7] R. Fan *et al.*, "Road Surface 3D Reconstruction Based on Dense Subpixel Disparity Map Estimation," *IEEE Transactions on Image Processing*, vol. 27, no. 6, pp. 3025–3035, 2018.

[8] Y. Feng *et al.*, "D2NT: A High-Performing Depth-to-Normal Translator," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 12 360–12 366.

[9] D. Eigen *et al.*, "Depth Map Prediction from a Single Image Using a Multi-Scale Deep Network," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 27, 2014.

[10] F. Liu *et al.*, "Learning Depth from Single Monocular Images Using Deep Convolutional Neural Fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 10, pp. 2024–2039, 2016.

[11] Z. Huang *et al.*, "Online, Target-Free Lidar-Camera Extrinsic Calibration via Cross-Modal Mask Matching," *IEEE Transactions on Intelligent Vehicles*, 2024, in press.

[12] H. Zhao *et al.*, "Dive Deeper into Rectifying Homography for Stereo Camera Online Self-Calibration," *2024 International Conference on Robotics and Automation (ICRA)*, 2024, in press.

[13] Y. Wang *et al.*, "SQLdepth: Generalizable Self-Supervised Fine-Structured Monocular Depth Estimation," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 38, no. 6, 2024, pp. 5713–5721.

[14] A. Kendall *et al.*, "PoseNet: a Convolutional Network for Real-Time 6-Dof Camera Relocalization," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2015, pp. 2938–2946.

[15] C. Campos *et al.*, "ORB-SLAM3: an Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap Slam," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[16] T. Qin *et al.*, "VINS-Mono: a Robust and Versatile Monocular Visual-Inertial State Estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[17] J. Deng *et al.*, "ImageNet: a Large-Scale Hierarchical Image Database," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 248–255.

[18] N. Jia *et al.*, "TFGNet: Traffic salient object detection using a feature deep interaction and guidance fusion," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 3, pp. 3020–3030, 2024.

[19] Z. Wu *et al.*, "S^3M-Net: Joint Learning of Semantic Segmentation and Stereo Matching for Autonomous Driving," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 2, pp. 3940–3951, 2024.

[20] F. Xue *et al.*, "Toward Hierarchical Self-Supervised Monocular Absolute Depth Estimation for Autonomous Driving Applications," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 2330–2337.

[21] V. Guizilini *et al.*, "3D Packing for Self-Supervised Monocular Depth Estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 2485–2494.

[22] F. El Jamiy and R. Marsh, "Survey on Depth Perception in Head Mounted Displays: Distance Estimation in Virtual Reality, Augmented Reality, and Mixed Reality," *IEEE Transactions on Image Processing*, vol. 13, no. 5, pp. 707–712, 2019.

[23] L. Li *et al.*, "Unsupervised-Learning-Based Continuous Depth and Motion Estimation with Monocular Endoscopy for Virtual Reality Minimally Invasive Surgery," *IEEE Transactions on Robotics*, vol. 17, no. 6, pp. 3920–3928, 2020.

[24] M. Zhang *et al.*, "DCPI-Depth: Explicitly infusing dense correspondence prior to unsupervised monocular depth estimation," *CoRR*, 2024.

[25] Y. Cao *et al.*, "Estimating Depth from Monocular Images as Classification Using Deep Fully Convolutional Residual Networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 11, pp. 3174–3182, 2018.

[26] M. Song *et al.*, "Monocular Depth Estimation Using Laplacian Pyramid-Based Depth Residuals," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 11, pp. 4381–4393, 2021.

[27] W. Yin *et al.*, "Enforcing Geometric Constraints of Virtual Normal for Depth Prediction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 5684–5693.

[28] H. Fu *et al.*, "Deep Ordinal Regression Network for Monocular Depth Estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 2002–2011.

[29] J. H. Lee *et al.*, "From Big to Small: Multi-Scale Local Planar Guidance for Monocular Depth Estimation," *CoRR*, 2021.

[30] S. F. Bhat *et al.*, "AdaBins: Depth Estimation Using Adaptive Bins," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 4009–4018.

[31] G. Yang *et al.*, "Transformer-Based Attention Networks for Continuous Pixel-Wise Prediction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 16 269–16 279.

[32] L. Yang *et al.*, "Depth Anything: Unleashing the Power of Large-Scale Unlabeled Data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[33] R. Birkl *et al.*, "MiDaS v3.1 – a Model Zoo for Robust Monocular Relative Depth Estimation," *CoRR*, 2023.

[34] M. Oquab *et al.*, "DINOv2: Learning Robust Visual Features without Supervision," *Transactions on Machine Learning Research*, 2024.

[35] R. Garg *et al.*, "Unsupervised CNN for Single View Depth Estimation: Geometry to the Rescue," in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Springer International Publishing, 2016, pp. 740–756.

[36] C. Godard *et al.*, "Unsupervised Monocular Depth Estimation with Left-Right Consistency," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 270–279.

[37] T. Zhou *et al.*, "Unsupervised Learning of Depth and Ego-Motion from Video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1851–1858.

[38] Z. Yin and J. Shi, "GeoNet: Unsupervised Learning of Dense Depth, Optical Flow and Camera Pose," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 1983–1992.

[39] A. Ranjan *et al.*, "Competitive Collaboration: Joint Unsupervised Learning of Depth, Camera Motion, Optical Flow and Motion Segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 12 240–12 249.

[40] X. Chen *et al.*, "Self-Supervised Monocular Depth Estimation: Solving the Edge-Fattening Problem," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023, pp. 5776–5786.

[41] H. Jung *et al.*, "Fine-Grained Semantics-Aware Representation Enhancement for Self-Supervised Monocular Depth Estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 12 642–12 652.

[42] M. Poggi *et al.*, "On the Uncertainty of Self-Supervised Monocular Depth Estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 3227–3237.

[43] N. Yang *et al.*, "D3VO: Deep Depth, Deep Pose and Deep Uncertainty for Monocular Visual Odometry," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 1281–1292.

[44] J. Watson *et al.*, "The Temporal Opportunist: Self-Supervised Multi-Frame Monocular Depth," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 1164–1174.

[45] R. Li *et al.*, "UnDeepVO: Monocular Visual Odometry through Unsupervised Deep Learning," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 7286–7291.

[46] Y.-Y. Jau *et al.*, "Deep Keypoint-Based Camera Pose Estimation with Geometric Constraints," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 4950–4957.

[47] W. Zhao *et al.*, "Towards Better Generalization: Joint Depth-Pose Learning without PoseNet," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 9151–9161.

[48] P. H. Christiansen *et al.*, "UnsuperPoint: End-to-end Unsupervised Interest Point Detector and Descriptor," *CoRR*, 2019.

[49] J. Tang *et al.*, "Self-Supervised 3D Keypoint Learning for Ego-Motion Estimation," in *Proceedings of the Conference on Robot Learning (CoRL)*. PMLR, 2021, pp. 2085–2103.

[50] J.-W. Bian *et al.*, "Unsupervised Scale-Consistent Depth Learning from Video," *International Journal of Computer Vision*, vol. 129, no. 9, pp. 2548–2564, 2021.

[51] J. Liu *et al.*, "Towards Better Data Exploitation in Self-Supervised Monocular Depth Estimation," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 763–770, 2023.

[52] R. Wang *et al.*, "PlaneDepth: Self-Supervised Depth Estimation via Orthogonal Planes," in *Proceedings of the IEEE/CVF Conference on*

Computer Vision and Pattern Recognition (CVPR), 2023, pp. 21 425–21 434.
[53] K. He *et al.*, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.
[54] B. K. Horn and B. G. Schunck, "Determining Optical Flow," *Artificial Intelligence*, vol. 17, no. 1-3, pp. 185–203, 1981.
[55] X. Zhu *et al.*, "Deep Feature Flow for Video Recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2349–2358.
[56] O. Chapelle and M. Wu, "Gradient Descent Optimization of Smoothed Information Retrieval Metrics," *Information Retrieval*, vol. 13, no. 3, pp. 216–235, 2010.
[57] A. Geiger *et al.*, "Vision Meets Robotics: the KITTI Dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.
[58] L. Sun *et al.*, "SC-DepthV3: Robust Self-Supervised Monocular Depth Estimation for Dynamic Scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.
[59] X. Lyu *et al.*, "HR-Depth: High Resolution Self-Supervised Monocular Depth Estimation," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 35, no. 3, 2021, pp. 2294–2301.
[60] H. Zhou *et al.*, "Self-Supervised Monocular Depth Estimation with Internal Feature Fusion," in *Proceedings of the British Machine Vision Conference (BMVC)*, 2021.
[61] C. Zhao *et al.*, "MonoViT: Self-Supervised Monocular Depth Estimation with a Vision Transformer," in *International Conference on 3D Vision (3DV)*, 2022, pp. 668–678.
[62] D. Shim and H. J. Kim, "SwinDepth: Unsupervised Depth Estimation Using Monocular Sequences via Swin Transformer and Densely Cascaded Network," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 4983–4990.
[63] N. Zhang *et al.*, "Lite-Mono: a Lightweight CNN and Transformer Architecture for Self-Supervised Monocular Depth Estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 18 537–18 546.
[64] Y. Sun and B. Hariharan, "Dynamo-Depth: Fixing Unsupervised Depth Estimation for Dynamical Scenes," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2023.
[65] Z. Feng *et al.*, "Disentangling Object Motion and Occlusion for Unsupervised Multi-Frame Monocular Depth," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2022, pp. 228–244.
[66] J. Uhrig *et al.*, "Sparsity Invariant CNNs," in *International Conference on 3D Vision (3DV)*. IEEE, 2017, pp. 11–20.
[67] F. Yu *et al.*, "Deep Layer Aggregation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 2403–2412.
[68] A. Geiger *et al.*, "Are We Ready for Autonomous Driving? the KITTI Vision Benchmark Suite," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 3354–3361.
[69] A. Saxena *et al.*, "Make3D: Learning 3D Scene Structure from a Single Still Image," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 824–840, 2009.
[70] Y. Chen *et al.*, "Self-Supervised Learning with Geometric Constraints in Monocular Video: Connecting Flow, Depth, and Camera," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 7063–7072.
[71] T. Shen *et al.*, "Beyond Photometric Loss for Self-Supervised Ego-Motion Estimation," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 6359–6365.
[72] J. Sturm *et al.*, "A Benchmark for the Evaluation of RGB-D SLAM Systems," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2012, pp. 573–580.
[73] D. P. Kingma and J. Ba, "Adam: a Method for Stochastic Optimization," in *International Conference on Machine Learning (ICML)*, 2015.
[74] J. Yan *et al.*, "Channel-Wise Attention-Based Network for Self-Supervised Monocular Depth Estimation," in *International Conference on 3D Vision (3DV)*. IEEE, 2021, pp. 464–473.

Yi Feng (Postgraduate Student Member, IEEE) received the B.E. degree in automation from Tongji University, Shanghai, China, in 2022. He is currently pursuing his Ph.D degree, supervised by Prof. Rui Fan, with the MIAS Group in the College of Electronics and Information Engineering at Tongji University. His research interests include computer vision and deep learning.

Zizhan Guo is currently pursuing his M.Sc. degree with the MIAS Group at Tongji University, supervised by Prof. Rui Fan. His research interests include computer vision and deep learning, with a particular emphasis on depth estimation.

Qijun Chen (Senior Member, IEEE) received the B.S. degree in automation from Huazhong University of Science and Technology, Wuhan, China, in 1987, the M.S. degree in information and control engineering from Xi'an Jiaotong University, Xi'an, China, in 1990, and the Ph.D. degree in control theory and control engineering from Tongji University, Shanghai, China, in 1999. He is currently a Full Professor in the College of Electronics and Information Engineering, Tongji University, Shanghai, China. His research interests include robotics control, environmental perception, and understanding of mobile robots and bioinspired control.

Rui Fan (Senior Member, IEEE) received the B.Eng. degree in Automation from the Harbin Institute of Technology in 2015 and the Ph.D. degree (supervisors: Prof. John G. Rarity and Prof. Naim Dahnoun) in Electrical and Electronic Engineering from the University of Bristol in 2018. He worked as a Research Associate (supervisor: Prof. Ming Liu) at the Hong Kong University of Science and Technology from 2018 to 2020 and a Postdoctoral Scholar-Employee (supervisors: Prof. Linda M. Zangwill and Prof. David J. Kriegman) at the University of California San Diego between 2020 and 2021. He began his faculty career as a Full Research Professor with the College of Electronics & Information Engineering at Tongji University in 2021, and was then promoted to a Full Professor in the same college, as well as at the Shanghai Research Institute for Intelligent Autonomous Systems in 2022.

Prof. Fan served as an associate editor for ICRA'23 and IROS'23/24, an area chair for ICIP'24, and a senior program committee member for AAAI'23/24/25. He is the general chair of the AVVision community and organized several impactful workshops and special sessions in conjunction with WACV'21, ICIP'21/22/23, ICCV'21, and ECCV'22. He was honored by being included in the Stanford University List of Top 2% Scientists Worldwide in both 2022 and 2023, recognized on the Forbes China List of 100 Outstanding Overseas Returnees in 2023, and acknowledged as one of Xiaomi Young Talents in 2023. His research interests include computer vision, deep learning, and robotics, with a specific focus on humanoid visual perception under the two-streams hypothesis.