Learning Monocular Depth in Dynamic Environment via Context-aware Temporal Attention

Zizhang Wu\textsuperscript{1*}, Zhuozheng Li\textsuperscript{1*}, Zhi-Gang Fan\textsuperscript{1*}, Yunzhe Wu\textsuperscript{1}, Yuanzhu Gan\textsuperscript{1} and Jian Pu\textsuperscript{2}

\textsuperscript{1}ZongmuTech
\textsuperscript{2}Fudan University
wuzizhang87@gmail.com, \{zhuozheng.li, zhigang.fan, nelson.wu, yuanzhu.gan\}@zongmu.com
jianpu@fudan.edu.cn

Abstract
The monocular depth estimation task has recently revealed encouraging prospects, especially for the autonomous driving task. To tackle the ill-posed problem of 3D geometric reasoning from 2D monocular images, multi-frame monocular methods are developed to leverage the perspective correlation information from sequential temporal frames. However, moving objects such as cars and trains usually violate the static scene assumption, leading to feature inconsistency deviation and misaligned cost values, which would mislead the optimization algorithm. In this work, we present CTA-Depth, a Context-aware Temporal Attention guided network for multi-frame monocular Depth estimation. Specifically, we first apply a multi-level attention enhancement module to integrate multi-level image features to obtain an initial depth and pose estimation. Then the proposed CTA-Refiner is adopted to alternatively optimize the depth and pose. During the CTA-Refiner process, context-aware temporal attention (CTA) is developed to capture the global temporal-context correlations to maintain the feature consistency and estimation integrity of moving objects. In particular, we propose a long-range geometry embedding (LGE) module to produce a long-range temporal geometry prior. Our approach achieves significant improvements (e.g., 13.5\% for the Abs Rel metric on the KITTI dataset) over state-of-the-art approaches on three benchmark datasets.

1 Introduction
Monocular depth estimation aims at predicting accurate pixel-wise depth from monocular RGB images. Due to its low cost and easy implementation, monocular depth estimation has achieved promising prospects in practical applications [Li et al., 2022; Wu et al., 2022; Mumuni and Mumuni, 2022]. Particularly, monocular depth estimation [Li et al., 2015; Liu et al., 2015; Ricci et al., 2018; Bhat et al., 2021; Yuan et al., 2022] under the single-frame setting has achieved convincing results by conducting robust convolutional neural networks with prior geometric constraints. Nevertheless, it is still challenging to precisely recover the 3D environment from a single monocular 2D image. On the other hand, noting that sequential image frames are achievable from the monocular camera, existing studies [Wang et al., 2019; Zhang et al., 2019; Patil et al., 2020] start paying greater attention to the depth estimation under the multi-frame setting. Inspired by the stereo matching task, multi-frame monocular depth estimation works typically employed the cost volume or cost map [Watson et al., 2021; Gu et al., 2023; Bae et al., 2022] to accomplish geometric reasoning and have gradually achieved state-of-the-art performance.

However, the widely applied static scene assumption [Klingner et al., 2020; Li et al., 2019] for the construction of the cost volume does not always hold in real-world scenarios. Specifically, moving objects such as cars, trains and pedestrians result in feature inconsistency deviation, misaligned cost values and degraded re-projection loss, which would mislead the optimization algorithm. To address this issue, recent works [Lee et al., 2021a; Feng et al., 2022; Wimbauer et al., 2021] attempted to solve dynamic problems by introducing an auxiliary object motion prediction module and segmentation masks to predict or disentangle dynamic objects explicitly. It inevitably increases the complexity and redundancy of the model and ignores the sustained temporal relation modeling of moving objects across long-range multiple frames, which thus limits the potential of sequential images for time-crucial industry implementation.

Figure 1: Given input images, our CTA-Depth with CTA-Refiner predicts more accurate depth maps compared to the initial prediction, especially for dynamic objects.
Thus, to efficiently boost the multi-frame context-temporal feature integration for dynamic targets without explicit auxiliary modules, we propose our CTA-Depth, a Context-aware Temporal Attention network for joint multi-frame monocular Depth and pose estimation. Specifically, we first utilize the multi-level attention enhancement (MAE) module for reliable initial estimation, which applies cross-scale attention layers to achieve ample interaction of different-scale features to equip the network with both local and global attentions. Furthermore, we develop the refiner CTA-Refiner to iteratively optimize our predictions with the inputs of context and temporal features. In specific, we develop our depth-pose context-aware temporal attention (CTA) with the cross-attention mechanism that assigns the temporal features as values, and the context features as queries and keys. As a result, it implicitly interacts with the context and temporal features to maintain the estimation integrity of moving objects across multiple sample frames. Additionally, to expand the temporal field of interest and aggregate useful clues for geometry reasoning, especially for dynamic targets within distant frames, we present our long-range geometry embedding (LGE) and provide it to the CTA process for seizing the long-range temporal geometry prior.

Qualitatively, as shown in Fig. 1(b), it is difficult for the single-frame method to recover the complete geometry of moving objects, while the predictions from our implicit CTA-Depth in Fig. 1(c) demonstrate its robustness at high-level feature recognition. Quantitatively, we conduct extensive experiments on three challenging benchmarks to validate the effectiveness of our pipeline against state-of-the-art models. We summarize our contributions as follows:

- We propose our CTA-Depth, an implicit, long-range Context-aware Temporal Attention guided network for supervised multi-frame monocular depth estimation, focusing on the dynamic object areas. It achieves state-of-the-art performance on challenging KITTI, VKITTI2, and nuScenes datasets.
- We design a novel depth-pose context-aware temporal attention (CTA), which implicitly learns the temporal geometric constraints for moving objects via attention-based integration.
- We introduce a novel long-range geometry embedding (LGE) module to promote geometry reasoning among the long-range temporal frames.
- We develop an effective multi-level attention enhancement (MAE) module to make global-aware initial depth and estimations. It promotes the distinction of far-away small objects from the static background.

2 Related Work

2.1 Monocular Depth Estimation

Convolutional neural networks with LiDAR supervision [Wang et al., 2015; Fu et al., 2018; Tang and Tan, 2018; Teed and Deng, 2019; Lee and Kim, 2020; Guizilini et al., 2021; Lee et al., 2021b] have shown promising results in monocular depth estimation. As a pioneer, Eigen et al. [Eigen et al., 2014] directly regressed depth by employing two stacked deep networks that made a coarse prediction from the whole image and then refined it locally. On the other hand, [Laina et al., 2016] adopted an end-to-end single CNN architecture with residual learning. To guide the encoded features to the desired depth prediction, [Lee et al., 2019] further deployed it with local planar guidance layers. Recently, [Ranftl et al., 2021] introduced a dense prediction transformer [Vaswani et al., 2017] for depth prediction. Meanwhile, [Bhat et al., 2021; Yang et al., 2021] developed global information processing with vision transformer [Dosovitskiy et al., 2021] for performance boost. Besides, [Yuan et al., 2022] adopted the swin-transformer [Liu et al., 2021] as the image encoder and the neural window fully-connected conditional random fields (NeWCRFs) module as the feature decoder. In particular, inspired by the RAFT [Teed and Deng, 2020] which employed a GRU-based recurrent operator to update optical flow, [Gu et al., 2023] proposed a multi-frame monocular approach with a deep recurrent optimizer to update the depth and camera poses alternately.

However, these cost-map-based multi-frame methods [Teed and Deng, 2019; Gu et al., 2023] lead to performance degradation within dynamic areas due to the static scene assumption. To solve this problem, we introduce a long-range geometry embedding module and effectively inject the proposed depth-pose context-aware temporal attention into the deep refinement network for the optimization process of depth and pose.

2.2 Depth Estimation in Dynamic Environment

Moving objects significantly hamper the multi-frame matching strategy due to the inevitable object truncation and occlusion. Specifically, both the re-projection loss calculation and the cost volume construction fall into catastrophic failure cases during the positional change of observed targets. Existing works [Li et al., 2019; Lee et al., 2021a; Watson et al., 2021] thus leveraged the segmentation mask to separate the static-scene depth loss from moving objects. In particular, they also proposed explicit object motion prediction and a disentanglement module to assist the cost volume construction. Specifically, SGDepth [Klingner et al., 2020] proposed a semantic masking scheme providing guidance to prevent dynamic objects from contaminating the photometric loss. DynamicDepth [Feng et al., 2022] introduced an object motion disentanglement module that takes dynamic category segmentation masks as input to explicitly disentangle dynamic objects. Considering time-critical tasks such as autonomous driving, instead of adopting explicit redundant static-dynamic separation algorithms, we developed an efficient implicit modeling pipeline with our novel context-aware temporal attention module. Besides, noting that previous works limited their methods to a few frames only due to an increase in computational cost, we developed a novel geometry embedding module to effectively encode semantic guidance from long-range time series.

As a result, our pipeline can dynamically interact with the long-range semantic flow with the current-frame spatial geometry in a fully differentiable manner and is thus available for real industry implementation.
Figure 2: Overview of our CTA-Depth. Given a reference image \( I_r \) and its \( N \) monocular sequential images \( \{ I_i \}_{i=1}^N \), we first group \( N \) pairs of network inputs, where each pair is composed of \( I_r \) and a sequential image \( I_i \). We then feed the two images into the MAE module and go through two heads to obtain the initial depth \( D_r^0 \) and pose \( P_r^0 \). Hence, the CTA-Refiner is proposed to alternately update the depth map and pose through iterations until the optimum solution, \( D_r^t \) and \( P_r^t \). In particular, with multiple temporal-neighboring frames \( I_i \), we also design a long-range geometry embedding module to provide long-range temporal geometric priors for the depth refiner efficiently.

3 Method

3.1 Overview

We demonstrate the framework of our approach in Fig. 2, which mainly consists of five components: the network inputs, the multi-level attention enhancement (MAE) module, the depth & pose context net, and the context-aware temporal attention refiner (CTA-Refiner) which includes the depth-pose context-aware temporal attention (CTA) module and the last long-range geometry embedding (LGE) module.

Given a reference image \( I_r \) within one video captured via a monocular camera, and its \( N \) video frames \( \{ I_i \}_{i=1}^N \), our goal is to predict the \( I_r \)’s accurate depth \( D_r^* \) and the relative camera poses \( \{ P_i^* \}_{i=1}^N \) for sequence image \( I_i \) with respect to \( I_r \). Specifically, we first regard the monocular video frames as network inputs. Then, we adopt the multi-level attention enhancement (MAE) module to extract representative visual features for the following depth head and (shared) pose head. Additionally, we introduce the multi-level feature setting and pyramid pooling module (PPM) [Zhao et al., 2017; Liu et al., 2018] to aggregate these features and deliver four cross-scale attention layers. The “\( R \)” denotes the rearranged up-scaling for feature maps.

3.2 Multi-level Attention Enhancement (MAE)

Affiliated with the optimizer-based pipeline, we deliver our multi-level attention enhancement (MAE) module to achieve the initial prediction of depth and pose. As shown in Fig. 3, we propose the multi-level feature setting and pyramid pooling module to reinforce the interest of the far-away small targets. In addition, to distinguish distant moving objects from the static background, we adopt cross-scale attention layers to enhance the interaction of different-scale features.

Specifically, as shown in Fig. 3, we utilize the feature encoder to extract four different-scale features from \( I_r \). The low-level features focus on the local details, while the high-level features seize the global context information, which both contribute to the rising interest of the distant targets [Lin et al., 2017; Liu et al., 2018]. Afterwards, we employ the pyramid pooling module (PPM) [Zhao et al., 2017] to aggregate these features and deliver four cross-scale attention layers to fuse the multi-level feature maps. We use the scaled features \( F_i \) as the query and key, and adopt the fusion features \( F_j \) as the value to stimulate the interaction of multi-scale features. Within each cross-attention layer, we also introduce the...
rearranged up-scaling operation [Yuan et al., 2022] to reduce
the network complexity and boundary contour refinement.
As shown in Fig. 5, our method achieves accurate depth pre-
diction of objects at different scales and long distances.

3.3 Context-aware Temporal Attention (CTA)

CTA-Refiner

We adopt the CTA-Refiner to iteratively refine initial esti-
mations to the final converged depth $D^*_t$ and pose $P^*_t$, together with two introduced extra inputs: the context features $F^c_r$ for depth and $F^c_i$ for the pose from the depth & pose context net. The refiner accepts these inputs to produce the prediction off-set $\Delta D^*_t$ and $\Delta P^*_t$ and then updates the results as follows:

$$D^{t+1}_t \leftarrow D^*_t + \Delta D^*_t, \quad (1)$$
$$P^{t+1}_t \leftarrow P^*_t + \Delta P^*_t. \quad (2)$$

In detail, we first calculate the cost map given $D^*_t$, $P^*_t$, $F^c_r$ and $F^c_i$, as shown in Fig. 2, where optimization for pose remains freezing when optimizing depth, and vice versa. Notably, the cost map measures the distance in feature space between $I_r$ and the sequence image $I_i$. Next, we adopt a simple feature extractor (two convolutions) to obtain the temporal features $F^t_i$ from the cost map, preparing for the following depth-pose CTA. Thus it implicitly rectifies the implied content-temporal inconsistency for moving objects and effectively promotes information integration between temporal features and depth-pose context features.

We formulate the cost map as the $L2$ distance between aligned feature maps $F^t_i$. Given the depth map $D$ of the reference image $I_r$ and the relative camera pose $T_i$ of another image $I_i$ with respect to $I_r$, the cost is constructed at each pixel $x$ in the reference image $I_r$:

$$C_i(x) = \| F^t_i(\pi(T_i \circ \pi^{-1}(x, D(x)))) - F^t_r(x) \|_2, \quad (3)$$

where $\pi()$ is the projection of 3D points in 3D space onto the image plane and $\pi^{-1}(x, D(x))$ is the inverse projection. The transformation converts 3D points from the camera space of $I_r$ to that of $I_i$. The multiple cost values are averaged at each pixel $x$ for multiple neighboring images.

Depth-pose Context-aware Temporal Attention (CTA)

The construction of the cost volume heavily relies on the static scene assumption, where it supposes that the object points remain static at time $t$ and $t^*$. Thus, we re-project the features at time $t$ to another plane with pose $t^*$ at time $t^*$, to achieve the matching cost values. However, moving objects break this assumption since targets such as cars, trains, or pedestrians with a certain speed could move within the time gap. This gives rise to the feature inconsistency deviation, degraded (mismatching) cost values and re-projection loss, and finally drawbacks our optimization for depth and pose. We discard explicit settings such as the object motion prediction module or disentangle module [Lee et al., 2021a; Feng et al., 2022; Wimbauer et al., 2021], which brings additional complexity and ignores the potential of complementary context-temporal information. Instead, we deliver our context-aware temporal attention (CTA) to implicitly rectify the mismatching problem, which efficiently cooperates the temporal features with context features via attention-based communication to achieve the feature consistency and estimation integrity.

Specifically, as shown in Fig. 4, taking depth optimization as an example, we first lift the temporal feature $F^t_i$ to value $V$ via the mapping function $\sigma(\cdot)$. Meanwhile, we create query $(Q)$ and key $(K)$ vectors by adding the mapping functions $\theta(\cdot)$ and $\phi(\cdot)$ from the context feature $F^c_r$, and prepare long-range geometry embedding (LGE). We first allocate the query, key and value as $Q = \theta(F^c_r) \otimes LGE, \quad K = \phi(F^c_r) \otimes LGE$, and $V = \sigma(F^t_i)$, respectively. Subsequently, the depth-context-aware temporal attention is denoted as:

$$F^d_i = f_s(Q \otimes K) \otimes V \oplus F^t_i, \quad (4)$$

where $f_s$ denotes the softmax operation, $\otimes$ denotes the pointwise addition and $\oplus$ denotes matrix multiplication.

Intuitively, compared with directly feeding $F^t_i$ and $F^c_i$ for refinement, our CTA explicitly aligns the features for moving objects through the cross-attention mechanism, to compensate for the mismatching discrepancy, which guarantees temporal-context feature fusion and seamless depth refinement. The temporal feature also helps the context feature to fulfill the moving target’s integrity, such as the second row of Fig. 1, where we rectify the wrong estimation in the ‘hole’ of the car head with temporal-content interaction. Similarly, for the pose optimization with fixed depth optimization, we employ context features $F^c_i$ to extract query $(Q)$ and key $(K)$ vectors with added LGE. Hence, we allocate the query, key and value as $Q = \theta(F^c_i) \otimes LGE, \quad K = \phi(F^c_i) \otimes LGE$, and $V = \sigma(F^t_i)$, respectively. Particularly, we lift the temporal feature $F^t_i$ for value $(V)$ vectors and finally adopt similar attention for pose context-aware temporal attention:

$$F^p_i = f_s(Q \otimes K) \otimes V \oplus F^t_i. \quad (5)$$

3.4 Long-range Geometry Embedding (LGE)

Existing works [Gu et al., 2023; Feng et al., 2022] usually adopt the two-frame correlation, such as the cost volume constructed from two frames, which limits the temporal interest field and wastes temporal helpful clues for geometry reasoning within the originally long-range multiple frames.
Therefore, to expand the temporal geometry relation range, especially for moving targets within long-range frames, we present our long-range geometry embedding module to create long-range geometry embedding (LGE), which seizes the beneficial long-range temporal hints for geometry and is inserted into the depth-pose CTA to promote the present depth and pose refinement.

Specifically, we demonstrate the detailed structure in Fig. 2, where we build the long-range cost map with the reference frame \(I_r\) and another arbitrary frame \(I_j(j \neq i)\). Hence, we can achieve N-1 cost maps with N-1 different-range frames. Notably, we achieve the temporal feature with long-range cost maps and convolution, which reveals the same as the procedure in CTA-Refiner. Afterwards, we propose the 1 \(\times\) 1 Conv, ReLU and Conv to deliver the geometry embedding \(\varphi(F_j)\). Specifically, the number of \(\varphi(F_j), j \neq i\) is N-1. Then, we aggregate this long-range temporal knowledge to create our embedding \(F_{ge}\) via the addition operation:

\[
F_{ge} = \sum_{j, j \neq i}^N \varphi(F_j), \tag{6}
\]

which will be added to the query and key vectors within the depth-pose CTA. Our approach efficiently reveals most of the geometry clues within the long-range temporal frames. It is also regarded as a temporal geometry prior to enhance the temporal-context association process.

### 3.5 Supervised Training Loss

We train our network by optimizing both depth and pose errors. Then, we formulate the depth loss as the L1 distance between the predicted depth map \(\hat{D}\) and the ground truth \(D\):

\[
L_{\text{depth}} = \sum_{s=1}^m \gamma^{m-s} \|D^s - \hat{D}\|_1, \tag{7}
\]

where the discounting factor \(\gamma\) is 0.85 and \(s\) denotes the stage number. There are \(m\) alternating update stages of depth and pose refinements. At each stage, we repeatedly refine the depth and pose \(n\) times. Next, our pose loss is defined based on the ground truth depth \(\hat{D}\) and pose \(\hat{T}_1\) with \(I_j\) relative to the reference image \(I_r\):

\[
L_{\text{pose}} = \sum_{s=1}^m \sum_{x} \gamma^{m-s} \|\pi(T^s \circ \pi^{-1}(x, \hat{D}(x))) - \pi(T_1 \circ \pi^{-1}(x, \hat{D}(x)))\|_1, \tag{8}
\]

where \(\circ\) means the Hadamard product. The pose loss summarizes the re-projection deviation of the pixel \(x\) according to the estimated camera pose \(T^s\) and the true pose \(T_1\) in each stage. \(\pi()\) is the projection of 3D points in 3D space onto the image plane. Its inverse projection \(\pi^{-1}(x, \hat{D}(x))\) maps the pixel \(x\) and its ground truth depth \(\hat{D}(x)\) back points in the 3D space. Finally, the total supervised loss is calculated by:

\[
L_{\text{supervised}} = L_{\text{depth}} + L_{\text{pose}}. \tag{9}
\]

### 4 Experiments and Results

#### 4.1 Datasets

**KITTI.** KITTI dataset [Geiger et al., 2012] is a popular benchmark for the task of autonomous driving, which provides over 93,000 depth maps with corresponding raw LiDAR scans and RGB images aligned with raw data. In experiments, we follow the widely-used KITTI Eigen split [Eigen et al., 2014] for network training, which is composed of 22,600 images from 32 scenes for training and 697 images from 29 scenes for testing. The corresponding depth of each RGB image is sampled sparsely by the LiDAR sensor.

**Virtual KITTI 2.** VKITTI2 dataset [Gaidon et al., 2016] is widely used for video understanding tasks, which consists of 5 sequence clones from the KITTI tracking benchmark and contains 50 high-resolution monocular videos generated from five different virtual worlds in urban settings under various imaging and weather conditions. These photo-realistic synthetic videos are fully annotated with depth labels.

**NuScenes.** NuScenes dataset [Caesar et al., 2020] is a large-scale multi-modal autonomous driving dataset that is the first to carry the completely autonomous vehicle sensor suite: 32-beam LiDAR, 6 cameras and 5 radars with 360° coverage. It comprises 1,000 scenes, where each scene lasts 20 seconds and is fully annotated with 3D bounding boxes for 23 classes and 8 attributes.

#### 4.2 Implementation Details

We implement our CTA-Depth in PyTorch and train it for 100 epochs with a mini-batch size of 4. The learning rate is \(2 \times 10^{-4}\) for both depth and pose refinement, which is decayed by a constant step (gamma=0.5 and step size=30). We set \(\beta_1 = 0.9\) and \(\beta_2 = 0.999\) in the Adam optimizer. We resize the input images to 320 \(\times\) 960 for training, and set the number of sequential images to 2 for CTA-Refiner by balancing both computation efficiency and prediction accuracy. For long-range geometry embedding, the number of temporally adjacent images is set to \(N = 3\). Since the output \(\varphi(F_j)\) of the LGE for the same image is fixed and not updated with the iterations, this provides more prior temporal information to CTA-Refiner while ensuring network efficiency. We fix \(m\) at 3 and \(n\) at 4 in experiments.

#### 4.3 Computation Time Analysis

Given the same Nvidia RTX A6000 GPU on the KITTI dataset, compared to the state-of-the-art one-frame method NeWCRFs [Yuan et al., 2022], the inference speed of our CTA-Depth, i.e., the number of images inferred per second (FPS), is greatly improved by 58.9\%, i.e., 5.53 (Ours) vs. 3.48 (NeWCRFs). This is because NeWCRFs use four swin-large transformers as multi-level encoders, while our method uses the lightweight ResNet18 backbone as the encoder to extract features. On the other hand, as shown in Table 2, although the FPS of CTA-Depth is slightly lower than that of the multi-frame method DRO [Gu et al., 2023] (5.53 vs. 6.25), our performance significantly outperforms DRO and MaGNet [Bae et al., 2022].
We first compare our CTA-Depth approach with the state-of-the-art methods on the KITTI Eigen split with the cap of 0-80m. Note that the seven widely-used metrics are calculated strictly following the baseline [Gu et al., 2023] and ground-truth median scaling is applied. “M” and “S” mean multiple and single frame. “M→S” means monocular multiple frame images are used in training while only a single frame image is used for inference.

We utilize bold to highlight the best results and color the multi-frame baseline results and our performance gain over them in blue.

Table 1: Quantitative results of supervised monocular depth estimation methods on the KITTI Eigen split. Note that the seven widely-used metrics are calculated strictly following the baseline [Gu et al., 2023] and ground-truth median scaling is applied. “M” and “S” mean multiple and single frame. “M→S” means monocular multiple frame images are used in training while only a single frame image is used for inference. We utilize bold to highlight the best results and color the multi-frame baseline results and our performance gain over them in blue.

Table 2: Quantitative results on KITTI Eigen split with the cap of 0-80m. Note that the seven widely-used metrics are calculated strictly following AdaBins [Bhat et al., 2021]. “Abs Rel” error occupies the main ranking metric. “*” means using additional data for training. We utilize bold to highlight the best results of single-frame methods and multi-frame methods.

4.4 Evaluation of Our Method

Evaluation on KITTI. We first compare our CTA-Depth approach against top performers of supervised monocular depth estimation on the KITTI dataset; see Tables 1 & 2 for the results. For a fair comparison, all methods are evaluated given the same sequential images. In Table 1, the seven widely-used evaluation metrics are calculated strictly following the work [Gu et al., 2023] and the ground-truth median scaling is applied to obtain the final output. In contrast, the seven evaluation metrics in Table 2 are calculated according to AdaBins [Bhat et al., 2021] and the final output is calculated by taking the average prediction of the image and its mirror image. Clearly, CTA-Depth achieves state-of-the-art performance across all evaluation metrics under different evaluation strategies. We further show the qualitative comparisons in Fig. 5 by comparing our method (c) with the recent approach [Gu et al., 2023] (b). As shown in the green boxes, our method yields finer depth estimation results for moving objects in dynamic scenes, small objects and object contours, such as the trams and traffic signs. In addition, as shown in the top row of Fig. 5, our predicted depth map for the tram window is more consistent with the rest parts of the tram.

Evaluation on Virtual KITTI 2. We further verify our method on the virtual KITTI 2 dataset as shown in Table 3. We use a subset of the virtual KITTI 2 dataset, which contains 1,700 images for training and 193 images for testing. Notably, our CTA-Depth achieved significantly better results than the multi-frame baseline methods over all evaluation metrics.

Evaluation on nuScenes. To further demonstrate the competitiveness of our approach, we also conduct an evaluation on the nuScenes dataset. In this experiment, we manually split a subset consisting of 2,600 images for training and 170 images for testing. The result is shown in Table 4. Again, the results show that our proposed method outperforms the baselines with a significant margin in all evaluation metrics.
4.5 Ablation Study

To inspect the importance of each module in our method, we conduct an ablation study on the KITTI dataset and provide the results in Table 5. From top to bottom, the proposed modules are added in turn until the full method is constructed.

**Baseline.** To verify the effectiveness of each component, we build a baseline model. This model has a similar network architecture as the full pipeline, which includes the encoder-decoder structure with a deep recurrent network. In other words, the proposed CTA module and LGM are removed. The MAE module keeps only the single-level ResNet18 feature net for the depth and pose estimations.

**Multi-level feature extraction.** We extract four levels of feature maps as shown in Fig. 3. The performance gain shows that it provides more valuable information for the model.

**Cross-scale attention.** Next, we use the same ResNet18 as the feature encoder and add cross-scale attention layers to decode the features at each level following Fig. 3.

**MAE module.** In addition to the two operations above, the rearrange upscale is added to reduce the network complexity. Together with the PPM, they consequently enhance the estimation performance.

**Depth-CTA and Pose-CTA.** We add Depth-CTA and Pose-CTA after the cost map to obtain dynamic features by implicitly modeling the temporal relation and learn the geometric constraints through the temporal frames. In this way, the learned adaptive dynamic features are fed to the GRU optimizers and yield a noticeable performance gain. The “Abs Rel” error is reduced from 0.056 to 0.047.

**LGE.** The long-range geometry embedding module provides temporal priors for dynamic objects in several temporal neighboring frames and enhances the learning of the CTA module by large margins.

**Multi-frame input.** Here, we set up three experiments with different numbers of input frames. The results show that the optimal performance is achieved when the adopted temporal frames are $I_{t-1}, I_t$ and $I_{t+1}$, i.e., two image pairs: $(I_{t-1}, I_t)$ and $(I_t, I_{t+1})$. However, the network becomes over-complicated and time-consuming to train or infer when utilizing more than three sample frames for depth refinement.

5 Conclusion

In this work, we present a novel CTA-Depth for multi-frame monocular depth estimation. To resolve the ambiguity caused by challenging dynamic scenes with moving objects, we propose the CTA-Refiner by designing context-aware temporal attention to implicitly leverage temporal information and model image self-similarities. In addition, we develop a novel long-range geometry embedding module to efficiently inject our refiner with geometry reasoning among the long-range temporal frames. Furthermore, we build a multi-level encoder-decoder network with the attention-enhanced predictor to obtain features with both global and local attentions. We achieve state-of-the-art performances on three challenging monocular depth estimation benchmarks. In the future, we would like to employ our multi-frame association mechanism in relevant tasks such as 3D object detection.

**Contribution Statement**

Zizhang Wu, Zhuozheng Li and Zhi-Gang Fan contributed equally to this work.
References


[Feng et al., 2022] Ziyue Feng, Liang Yang, Longlong Jing, Haiyan Wang, Yingli Tian, and Bing Li. Disentangling object motion and occlusion for unsupervised multi-frame monocular depth. In European Conf. on Computer Vision (ECCV), pages 228–244, 2022.


[Li et al., 2022] Yingyan Li, Yuntao Chen, Jiawei He, and Zhaoxiang Zhang. Densely constrained depth estimator for monocular 3d object detection. In European Conf. on Computer Vision (ECCV), pages 718–734. Springer, 2022.


