Flow-Guided Consistent Video Depth Estimation for Cross-Dataset Generalization

Jaeseok Jang and Chang-Su Kim
Korea University, Korea
jsjang@mcl.korea.ac.kr, changsukim@korea.ac.kr

***Abstract*—Consistent video depth estimation aims to predict temporally consistent depth maps for videos. Various techniques have been developed to utilize temporal correlations between frames; however, most of these techniques generalize poorly to unconstrained scenes. Some recent methods based on test-time training require an impractical amount of inference time. Other methods apply attention mechanisms to video features to accomplish temporal consistency, but they do not consider scene variations across frames. In this paper, we propose a novel algorithm called flow-guided consistent video depth estimator (FCVD), in which a spatial feature alignment process leverages optical flow to modulate scene variations. Zero-shot cross-dataset evaluation results demonstrate that the proposed FCVD algorithm outperforms existing techniques in terms of depth accuracy and temporal consistency.**

I. Introduction

Temporally consistent video depth estimation is essential for many applications, including 3D reconstruction, 2D-to-3D conversion, and bokeh rendering. Various attempts have been made to predict spatially precise and temporally consistent depth maps from monocular videos. Although recent techniques have significantly improved spatial accuracy, capturing temporal dependencies across consecutive frames still remains a challenge. This often leads to flickering depth predictions, as illustrated in Fig. 1. Moreover, most previous video depth estimation methods lack generalization capability across diverse scene types, *e.g.* indoor vs. outdoor scenes and stationary vs. dynamic scenes.

Several techniques [1–4] have been proposed to transfer information between frames to achieve temporal consistency. For example, [3] and [4] adopt attention mechanisms to capture inter-frame correlations, while [1] applies temporal loss functions on predictions of consecutive frames. However, these methods are typically optimized for closed domains *e.g.* indoor scenes, so their robustness for unconstrained and diverse scenes may be limited.

Geometry-based methods [5–11] such as structure-from-motion approaches have also been intensively investigated. While some of these methods [6–8] struggle to generalize to unseen data, Teed and Deng's algorithm [5] shows robust performance. Nevertheless, its performance heavily relies on camera pose estimation, which often fails in dynamic scenes. Although test-time training methods [9–11] achieve robust performance, they suffer from a long inference time.

Meanwhile, several techniques [12–14] have been proposed to predict temporally consistent depth maps with strong generalization capability. They leverage existing depth predictors and enhance the predictions via attention mechanisms and temporal loss functions. Xian *et al.* [12] incorporate a transformer module into the structure of existing depth estimator, while Wang *et al.* [13, 14] operate as add-on frameworks, which take the prediction results of existing models as input. However, they do not consider spatial variations across frames and struggle to predict temporally consistent results in the presence of large motions.

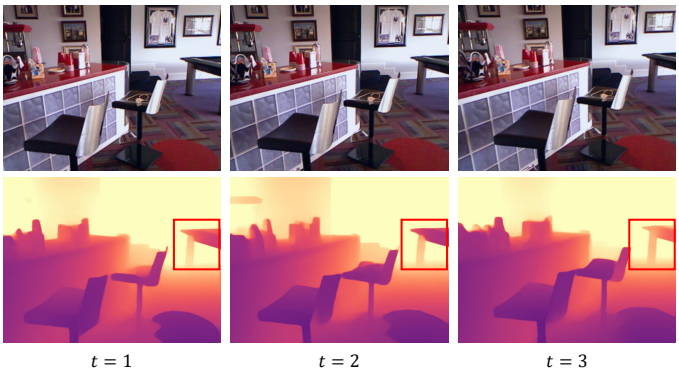

Fig. 1. An example of flickering depth predictions produced by an existing depth estimation method.

In this paper, we propose a flow-guided consistent video depth estimator (FCVD), which effectively attends to recent frames using optical flow. As done in [12–14, 16–18], we perform predictions in a disparity space to handle scale and shift ambiguities [16]. This allows our algorithm to be trained on diverse datasets and achieve strong generalization capability in zero-shot cross-dataset evaluations. The proposed FCVD functions as an add-on framework for existing depth predictors, similar to [13, 14].

We conduct zero-shot cross-dataset evaluations on widely-used depth datasets: NYUDv2 [19], KITTI [20], Sintel [21] and DDAD [22]. The proposed FCVD outperforms previous methods in terms of both depth accuracy and temporal consistency. The main contributions of this paper are as follows:

- We propose an add-on video depth estimation framework, FCVD, which leverages optical flow to estimate temporally consistent depth maps for unconstrained videos.
- The results of zero-shot cross-dataset evaluations demonstrate that the proposed FCVD outperforms previous methods on depth accuracy and temporal consistency.

979-8-3315-7206-8/25/$31.00

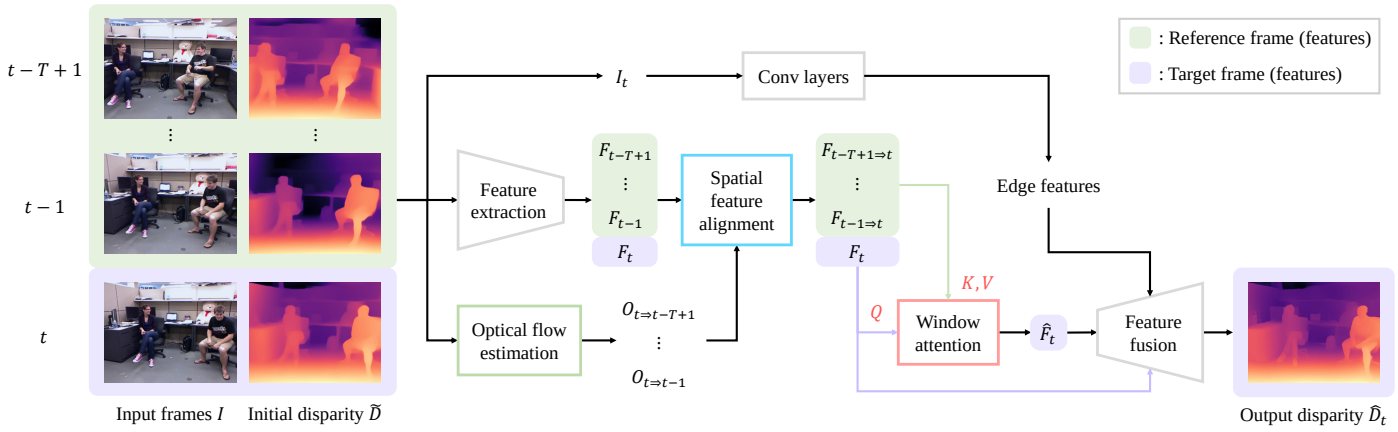

Fig. 2. An overview of the proposed FCVD. Given an input video clip of length T, the last frame I_t is considered as a target frame, while the remaining frames serve as reference frames. The initial disparity maps $\tilde{D}$ are obtained by an existing image depth predictor. We estimate optical flow between the target frame and each reference frame using [15]. We leverage the optical flow in the spatial feature alignment process to enable the window attention module and capture spatial correlations across the extracted feature maps of the input frames more effectively. The window attention module attends the target frame feature map F_t with each of the aligned feature maps. The attended feature map $\hat{F}_t$ is then fused with F_t and additional edge features from I_t to produce a temporally consistent disparity map $\hat{D}_t$ for the target frame I_t. The FCVD algorithm operates using a sliding window over frames in the inference stage.

II. Proposed Algorithm

Fig. 2 is an overview of the proposed FCVD framework. Using an existing image depth predictor [18], we first obtain initial disparity maps. Note that each initial disparity map is normalized within the range $[0, 1]$ to address scale and shift ambiguities and achieve strong generalization capability [16]. Given an input video clip and its initial disparity maps, FCVD aims to estimate a temporally consistent disparity map for a target frame, which is the last frame I_t of the input clip. FCVD consists of four stages: feature extraction, spatial feature alignment, window attention, and feature fusion. A backbone network [23] extracts features from the input frames. We then spatially align each reference frame feature map $\{F_k\}_{k=t-T+1}^{t-1}$ with the target frame feature map F_t to enable the subsequent window attention module to more effectively capture spatial correlations across frames. Next, we perform local window attention across the aligned feature maps. Finally, we fuse the attended feature map $\hat{F}_t$ with the original target frame feature map F_t and additional edge features from I_t, producing the final output disparity map $\hat{D}_t$. The entire framework operates using a sliding window over frames in the inference stage.

A. Feature Extraction

Given an input RGB-D sequence $\{I_k, \tilde{D}_k\}_{k=t-T+1}^{t}$ of length T, the backbone network [23] extracts multi-scale feature maps $X_0 = \{\boldsymbol{x}_0^l\}_{l=1}^4$, where $\boldsymbol{x}_0^l \in \mathbb{R}^{T\times\frac{H}{2^{(l+1)}}\times\frac{W}{2^{(l+1)}}\times C_l}$ denotes the feature map at the l-th level. We transform each feature map in X_0 into a feature map of shape $T\times\frac{H}{4}\times\frac{W}{4}\times C$ using a multi-layer perceptron (MLP) and bilinear scaling. The transformed feature maps are then fused into a single feature map $X \in \mathbb{R}^{T\times\frac{H}{4}\times\frac{W}{4}\times C}$ where the embedding dimension $C = 256$. Note that the features from the T frames are processed independently. In X, we denote the feature map corresponding to frame I_k as F_k, where $F_k \in \mathbb{R}^{\frac{H}{4}\times\frac{W}{4}\times C}$.

B. Spatial Feature Alignment

We align each of the reference frame feature maps $\{F_k\}_{k=t-T+1}^{t-1}$ to the target frame feature map F_t. To this end, we compute the backward optical flow $\{O_{t\Rightarrow k}\}_{k=t-T+1}^{t-1}$ from the target frame to each reference frame using [15]. Fig. 3 illustrates the spatial feature alignment process. Directly using the warped feature maps as keys and values for cross-attention can degrade performance due to inaccurate optical flow. To alleviate this issue, we filter the warped feature maps and update them using a gating mechanism [24].

First, we compute the visibility mask $M_{t\Rightarrow k}$ [2] from I_k to I_t, which is defined as:

$$M_{t\Rightarrow k} = \exp\left(-\beta \left\| I_t - I_{k\Rightarrow t} \right\|_2^2\right) \tag{1}$$

where $I_{k\Rightarrow t}$ is the frame I_k warped toward t. The visibility mask assigns higher values to pixels that are similar in both frames. We set $\beta = 50$. We also warp F_k toward t, denoted as $F_{k\Rightarrow t}^0$, and concatenate it with F_t and $M_{t\Rightarrow k}$. These concatenated features are passed through convolutional layers to compute the forget gate f and the input gate i using sigmoid activations, and the update feature g using a tanh activation. Finally, we filter $F_{k\Rightarrow t}^0$ using f and then update it using i and g, yielding the final aligned feature map $F_{k\Rightarrow t}$.

The alignment process is formulated as

$$\begin{aligned} f &= \sigma\left(W_{f1}F_{k\Rightarrow t}^0 + W_{f2}F_t + W_{f3}M_{t\Rightarrow k} + b_f\right), \\ i &= \sigma\left(W_{i1}F_{k\Rightarrow t}^0 + W_{i2}F_t + W_{i3}M_{t\Rightarrow k} + b_i\right), \\ g &= \tanh\left(W_{g1}F_{k\Rightarrow t}^0 + W_{g2}F_t + W_{g3}M_{t\Rightarrow k} + b_g\right), \\ F_{k\Rightarrow t} &= f \circ F_{k\Rightarrow t}^0 + i \circ g, \end{aligned} \tag{2}$$

where $\circ$ denotes the Hadamard product. Employing this alignment process, we obtain spatially aligned and reliable reference frame feature maps $\{F_{k\Rightarrow t}\}_{k=t-T+1}^{t-1}$.

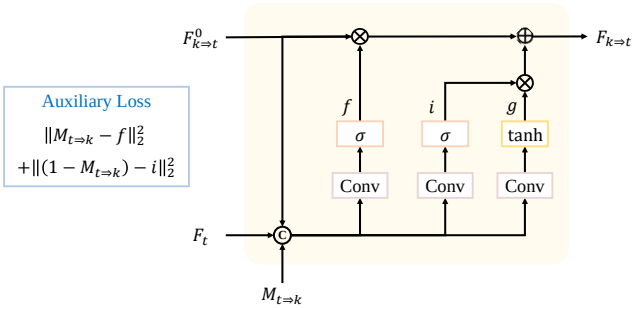

Fig. 3. Illustration of the spatial feature alignment process. Starting from the warped reference frame feature map $F^0_{k\Rightarrow t}$, we filter out unreliable information and update the feature map using a gating mechanism.

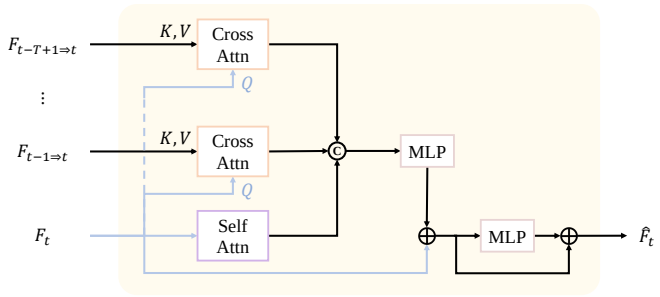

Fig. 4. Structure of an attention block in the attention module. The target frame feature map is updated progressively through each attention block. We use three attention blocks but show only one block for clarity.

To train the spatial feature alignment module, we use an auxiliary loss defined as:

$$\mathcal{L}_{aux} = \|M_{t\Rightarrow k} - f\|_2^2 + \|(1 - M_{t\Rightarrow k}) - i\|_2^2 . \quad (3)$$

This loss encourages the spatial feature alignment module to suppress unreliable regions in the warped features using the forget gate, and to update those regions based on target frame information via the input gate.

C. Window Attention Module

Fig. 4 shows the structure of an attention block in our window attention module. We apply self-attention to F_t and cross-attention between F_t and $\{F_{k\Rightarrow t}\}_{k=t-T+1}^{t-1}$. For cross-attention, we extract queries from F_t, while keys and values from each $F_{k\Rightarrow t}$. To improve computational efficiency in the attention module, we adopt the neighborhood attention (NA) mechanism [25]. Specifically, for each query token from F_t, we only consider key and value tokens in a local window around the position of the query token.

The self- and cross-attention operations in an attention block produce T attended feature maps. We apply MLP to fuse these T feature maps; the fused feature map replaces F_t in the next attention block. We use three attention blocks with the attention window size $k = 7$. After passing through all attention blocks, the output target frame feature map is denoted by $\hat{F}_t$.

D. Feature Fusion and Decoding

We utilize the target frame feature map F_t, its attended counterpart $\hat{F}_t$, and additional edge features extracted from I_t to generate the final disparity output $\hat{D}_t$. Specifically, the attended feature map $\hat{F}_t$ is first combined with F_t via a residual convolutional block. A subsequent residual block then integrates the edge features to further improve the spatial accuracy.

E. Loss Functions

We freeze the weights of the initial depth model [18] and train the proposed FCVD using three loss functions: spatial loss, temporal loss, and auxiliary loss. We adopt the spatial loss function in [16], which consists of affinity invariant loss and multi-scale gradient matching loss,

$$\mathcal{L}_{spa} = \mathcal{L}_{a\_inv} + \alpha \mathcal{L}_{grad}. \quad (4)$$

We set the weight α to 0.5 as in [16].

To enforce temporal consistency, we adopt the temporal loss in [13], given by

$$\mathcal{L}_{tem}(t, t-1) = \frac{1}{N} \sum_{i=1}^{N} M^{(i)}_{t\Rightarrow t-1} \left\| \hat{D}^{(i)}_t - \hat{D}^{(i)}_{t-1\Rightarrow t} \right\|_1 , \quad (5)$$

where N denotes the number of pixels, $M_{t\Rightarrow t-1}$ is the visibility mask in (1), and $\hat{D}_{t-1\Rightarrow t}$ is the prediction $\hat{D}_{t-1}$ warped toward t.

The final training loss is formulated as

$$\mathcal{L} = \frac{\mathcal{L}_{spa}(t-1) + \mathcal{L}_{spa}(t)}{2} + \lambda \mathcal{L}_{tem}(t, t-1) + \mathcal{L}_{aux}, \quad (6)$$

where $\lambda = 0.2$ is the weight for the temporal loss, and $\mathcal{L}_{aux}$ is the auxiliary loss in (3).

III. Datasets

We train the proposed FCVD on a mixed dataset comprising TartanAir [28] and IRS [29]. We conduct zero-shot cross-dataset evaluation on four widely-used benchmarks: NYUDv2 [19], KITTI [20], Sintel [21] and DDAD [22].

TartanAir: It is a synthetic dataset collected in photo-realistic environments with various light conditions, weather, and moving objects. It consists of 369 video sequences across 30 environments. We use training and validation splits in [30].

IRS: It is a large synthetic indoor robotics stereo dataset, containing over 100K stereo images and corresponding ground truth disparities. We only use the left images for training.

NYUDv2: It is an indoor dataset, which consists of 464 scenes with depths up to 10m. We use the official test split [19]. Since long continuous test sequences are not available, we use 654 video clips of length 5, where the last frames correspond to test frames, following the setting in [26].

Sintel: It is a synthetic dataset derived from 3D animated films, with depths up to 72m. It is a challenging dataset due to large

TABLE I
QUANTITATIVE RESULTS OF ZERO-SHOT CROSS-DATASET EVALUATION. ALL VALUES ARE MULTIPLIED BY 100 FOR CLARITY.

	NYUDv2			Sintel			KITTI			DDAD		
	$\delta_1 \uparrow$	*Rel* $\downarrow$	*OPW* $\downarrow$	$\delta_1 \uparrow$	*Rel* $\downarrow$	*OPW* $\downarrow$	$\delta_1 \uparrow$	*Rel* $\downarrow$	*OPW* $\downarrow$	$\delta_1 \uparrow$	*Rel* $\downarrow$	*OPW* $\downarrow$
Depth Anything [18]	96.53	5.24	8.05	78.33	19.24	25.22	95.34	7.48	14.05	88.80	10.67	12.10
ST-CLSTM [26]	88.28	11.21	15.43	40.43	88.04	61.70	34.39	44.57	45.17	23.12	83.18	49.30
FMNet [3]	85.46	12.49	18.57	40.42	94.10	65.44	40.01	38.32	41.71	28.29	62.12	33.89
WSVD [27]	71.06	20.01	31.63	47.94	63.68	67.57	39.60	36.99	36.63	23.45	62.82	28.18
DeepV2D [5]	79.41	15.43	28.08	45.13	77.06	116.82	56.02	28.29	69.56	32.94	58.41	85.90
Robust-CVD [10]	88.68	10.90	**4.42**	60.69	33.03	24.38	74.78	16.63	**10.61**	72.30	20.41	**8.05**
ViTA [12]	91.14	9.43	10.39	66.47	27.32	24.43	89.98	10.11	13.28	81.64	**15.50**	11.78
NVDS [13]	**92.48**	**8.28**	5.56	**74.47**	**23.35**	16.04	90.62	10.18	16.94	81.16	17.16	12.16
FCVD (Proposed)	92.35	8.29	5.05	73.48	23.75	**14.50**	**91.33**	**9.66**	12.34	**82.43**	15.80	11.44

motions and blurry effects. We use the final versions of 23 video sequences for evaluation.

KITTI: It is an outdoor dataset, composed of real-world driving scenes with depths to 80m. We use the Eigen test split [31], which includes 28 test video sequences. Following [12], we limit each sequence to a maximum of 200 frames and use Garg crop [32] for evaluation.

DDAD: It is another outdoor dataset designed for autonomous driving. Compared to KITTI, it is more densely annotated and covers a longer range of depths, up to 200m. We use the official 50 validation video sequences for evaluation.

IV. EXPERIMENTAL RESULTS

A. Implementation Details

We set the input clip length to $T = 4$. Since the training dataset of VDW [13] is not publicly available, we initialize our backbone with their pretrained weights. We use the AdamW optimizer with cosine annealing. We perform random crop augmentation with an input resolution of 384×384 and train FCVD for 25 epochs with a batch size of 4 and a learning rate of 1×10^{-5}. We use four NVIDIA RTX 4090 GPUs.

B. Evaluation Metrics

We evaluate the performance of the proposed FCVD in terms of both depth accuracy and temporal consistency. For depth accuracy, we use the commonly adopted mean absolute value of relative errors and percentage of pixels with $\delta > 1.25$, denoted by *Rel* and δ_1, respectively. We adopt the optical flow based warping (*OPW*) metric [3, 13] to evaluate temporal consistency, which is defined as:

$$OPW = \frac{1}{N-1} \sum_{t=2}^{N} \mathcal{L}_{tem}\left(t, t-1\right), \tag{7}$$

where N is the length of an input video sequence.

C. Comparative Assessment

Table I presents quantitative results on zero-shot cross-dataset evaluation. Note that we compare the performance of the proposed FCVD against the video-based depth estimation methods only, while [18] is an image-based depth estimation method. The methods in Table I are categorized into four groups: non-geometric methods [3, 26, 27], a geometric method [5], a test-time training method [10], and robust methods [12, 13]. The non-geometric methods are trained without geometric constraints. For [26] and [3], models pretrained on the NYUDv2 dataset are used. These two are the only methods trained using one of the evaluation datasets. Teed and Deng [5] represent a geometric method that employs a multi-view stereo approach to perform matching over video frames. While [10] also adopts geometric constraints, it is a test-time training method to optimize network parameters during inference. The robust methods [12, 13] and the proposed FCVD aim to predict depth maps without geometric constraints. In the case of [12], which does not function as an add-on framework, we use the DPT-Large version [17]. We apply their overlap inference strategy when evaluating [12]. Since [13] functions as an add-on framework similar to FCVD, we train it based on [18] in the same way as FCVD. Fig. 5 shows qualitative comparisons of competing methods.

It is observed from Table I and Fig. 5 that the non-geometric methods [3, 26, 27] exhibit limited generalization capability. While they can estimate reliable depths for indoor scenes, their performance degrades significantly in dynamic or outdoor scenes. The performance of the geometric method [5] is also inconsistent across datasets. In particular, its temporal consistency notably drops on the dynamic scenes of Sintel, where the geometric constraints are ineffective. As shown in Fig. 5, [5] often fails to match frames in the presence of large motions. The test-time training method [10] achieves strong temporal consistency and generalizes well across datasets. However, its depth accuracy is suboptimal compared to the robust methods [12, 13], and it often produces visual artifacts in Fig. 5. The quantitative results demonstrate that [12] and [13] generalize well to unconstrained scenes across diverse domains. Nonetheless, their predictions often lose some details in Fig. 5, whereas the proposed FCVD preserves such details more faithfully. Overall, the proposed FCVD outperforms the existing video depth estimation methods.

Fig. 6 compares the temporal consistency. For each sampled video sequence from Sintel, KITTI, and DDAD, we select a horizontal scanline and extract slices of the predicted disparity maps along this scanline over time. These slices are concatenated to visualize the temporal behavior of the predictions.

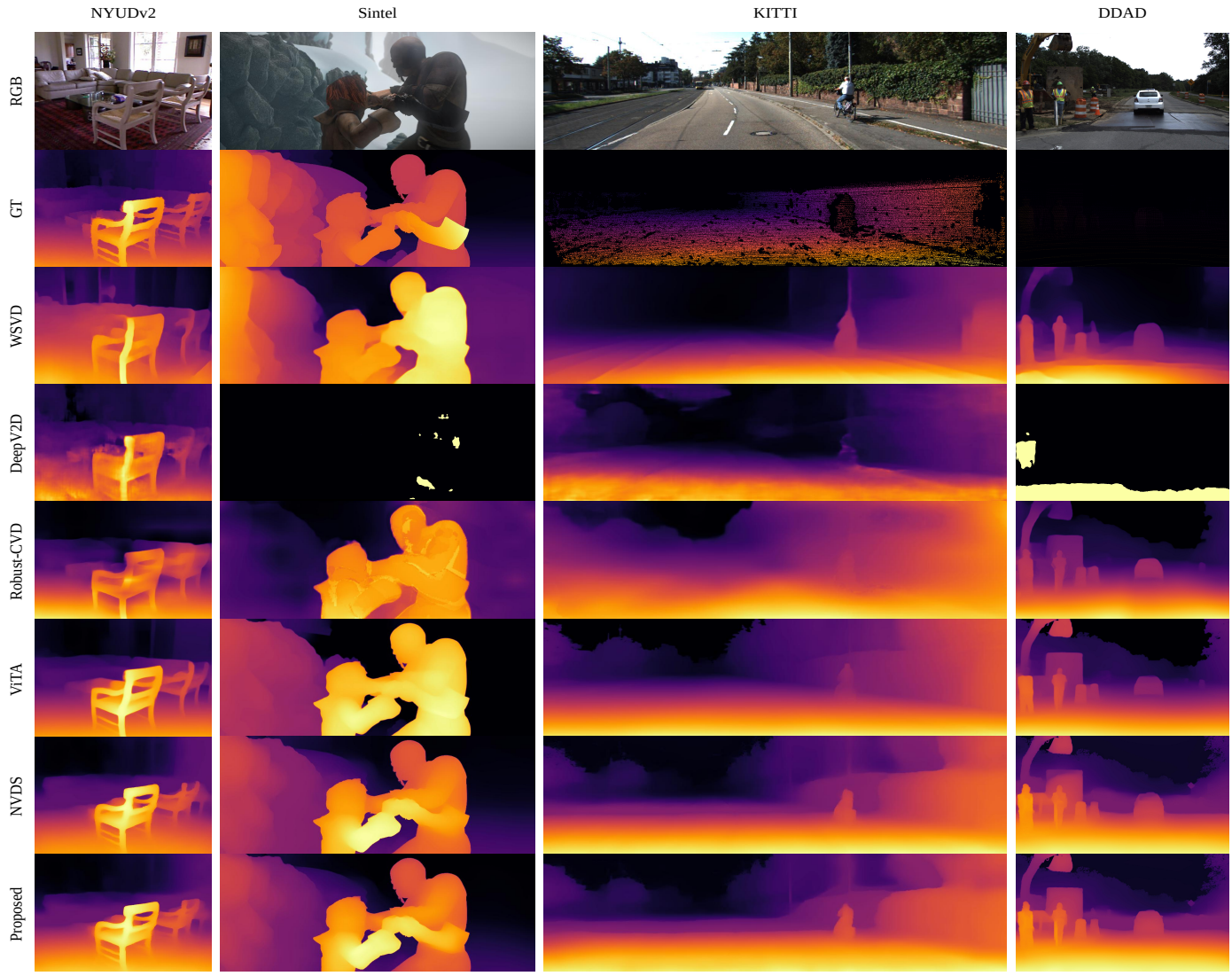

Fig. 5. Qualitative comparisons of video depth estimation methods on four benchmarks. Note that the ground-truth depths for KITTI and DDAD are sparse.

Fewer vertical color fluctuations indicate higher temporal consistency, while sharper vertical edges reflect clearer object boundaries. We observe that [10] struggles to preserve object boundaries. In the Sintel and KITTI results, both FCVD and [13] produce more accurate disparities and clearer object boundaries than [12]. Although FCVD and [13] exhibit similar temporal trends overall, the Sintel results indicate that FCVD achieves superior temporal consistency than [13].

V. Conclusions

In this paper, we propose a flow-guided consistent video depth estimator (FCVD) for cross-dataset generalization. Compared to existing approaches, the proposed FCVD effectively leverages optical flow to integrate information from reference frames into a target frame. As FCVD is designed as an add-on framework, it can be applied to any existing depth estimation model to improve the temporal consistency of its predictions. Experimental results on zero-shot cross-dataset evaluation demonstrate that FCVD produces accurate and temporally consistent depth maps across unconstrained domains.

Acknowledgements

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government (MSIT) (No. RS-2024-00397293 and No. RS-2022-NR068986).

References

[1] S. Li *et al.*, "Enforcing temporal consistency in video depth estimation," in *Proc. IEEE ICCV Workshop*, 2021.

[2] Y. Cao, Y. Li, H. Zhang, C. Ren, and Y. Liu, "Learning structure affinity for video depth estimation," in *Proc. ACM Multimedia*, 2021.

[3] Y. Wang *et al.*, "Less is more: Consistent video depth estimation with masked frames modeling," in *Proc. ACM Multimedia*, 2022.

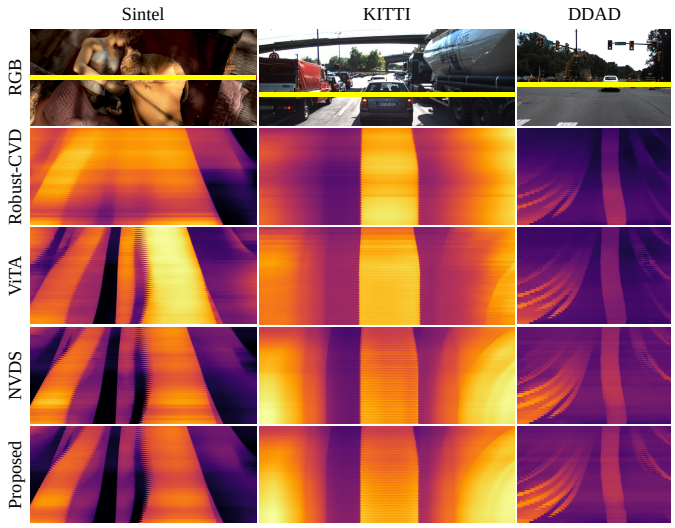

Fig. 6. Visualization of temporal consistency. Temporal evolutions of depths of the selected scanlines in yellow are shown.

[4] R. Yasarla *et al.*, "MAMo: Leveraging memory and attention for monocular video depth estimation," in *Proc. IEEE ICCV*, 2023.

[5] Z. Teed and J. Deng, "DeepV2D: Video to depth with differentiable structure from motion," in *Proc. ICLR*, 2020.

[6] J.-W. Bian *et al.*, "Unsupervised scale-consistent depth learning from video," *Int. J. Comput. Vis.*, vol. 129, pp. 2548–2564, 2021.

[7] J.-W. Bian *et al.*, "Auto-rectify network for unsupervised indoor depth estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, pp. 9802–9813, 2022.

[8] N. Zhang, F. Nex, G. Vosselman, and N. Kerle, "Lite-Mono: A lightweight CNN and transformer architecture for self-supervised monocular depth estimation," in *Proc. IEEE CVPR*, 2023.

[9] X. Luo, J.-B. Huang, R. Szeliski, K. Matzen, and J. Kopf, "Consistent video depth estimation," in *Proc. ACM Multimedia*, 2020.

[10] J. Kopf, X. Rong, and J.-B. Huang, "Robust consistent video depth estimation," in *Proc. IEEE CVPR*, 2021.

[11] Z. Zhang, F. Cole, R. Tucker, W. T. Freeman, and T. Dekel, "Consistent depth of moving objects in video," in *Proc. ACM Multimedia*, 2021.

[12] K. Xian, J. Peng, Z. Cao, J. Zhang, and G. Lin, "ViTA: Video transformer adaptor for robust video depth estimation," *IEEE Trans. Multimedia*, vol. 26, pp. 3302–3316, 2023.

[13] Y. Wang *et al.*, "Neural video depth stabilizer," in *Proc. IEEE ICCV*, 2023.

[14] Y. Wang *et al.*, "NVDS+: Towards efficient and versatile neural stabilizer for video depth estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 1, pp. 583–600, 2024.

[15] H. Xu, J. Zhang, J. Cai, H. Rezatofighi, and D. Tao, "GMFlow: Learning optical flow via global matching," in *Proc. IEEE CVPR*, 2022.

[16] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 3, pp. 1623–1637, 2020.

[17] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," in *Proc. IEEE ICCV*, 2021.

[18] L. Yang *et al.*, "Depth Anything: Unleashing the power of large-scale unlabeled data," in *Proc. IEEE CVPR*, 2024.

[19] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *Proc. ECCV*, 2012.

[20] J. Uhrig *et al.*, "Sparsity invariant CNNs," in *Int. Conf. 3D Vis.*, 2017.

[21] D. J. Butler, J. Wulf, G. B. Stanley, and M. J. Black, "A naturalistic open source movie for optical flow evaluation," in *Proc. ECCV*, 2012.

[22] V. Guizilini, R. Ambrus, S. Pillai, A. Raventos, and A. Gaidon, "3D packing for self-supervised monocular depth estimation," in *Proc. IEEE CVPR*, 2020.

[23] E. Xie *et al.*, "SegFormer: Simple and efficient design for semantic segmentation with transformers," in *Proc. NIPS*, 2021.

[24] X. Shi *et al.*, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. NIPS*, 2015.

[25] A. Hassani, S. Walton, J. Li, S. Li, and H. Shi, "Neighborhood attention transformer," in *Proc. IEEE CVPR*, 2023.

[26] H. Zhang *et al.*, "Exploiting temporal consistency for real-time video depth estimation," in *Proc. IEEE ICCV*, 2019.

[27] C. Wang, S. Lucey, F. Perazzi, and O. Wang, "Web stereo video supervision for depth prediction from dynamic scenes," in *Int. Conf. 3D Vis.*, 2019.

[28] W. Wang *et al.*, "TartanAir: A dataset to push the limits of visual slam," in *IEEE/RSJ Int. Conf. Intell. Robots. Syst.*, 2020.

[29] Q. Wang *et al.*, "IRS: A large naturalistic indoor robotics stereo dataset to train deep models for disparity and surface normal estimation," in *IEEE Int. Conf. Multimedia and Expo*, 2021.

[30] Z. Teed, L. Lipson, and J. Deng, "Deep patch visual odometry," in *Proc. NIPS*, 2023.

[31] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Proc. NIPS*, 2014.

[32] R. Garg, V. Kumar, G. Carneiro, and I. Reid, "Unsupervised CNN for single view depth estimation: Geometry to the rescue," in *Proc. ECCV*, 2016.