NVDS+: Towards Efficient and Versatile Neural Stabilizer for Video Depth Estimation Supplementary Document

Yiran Wang, Min Shi, Jiaqi Li, Chaoyi Hong, Zihao Huang, Juewen Peng, Zhiguo Cao, *Member, IEEE*, Jianming Zhang, Ke Xian, and Guosheng Lin, *Member, IEEE*

This supplement contains the following contents:

- More details on the VDW dataset.
- More implementation details for NVDS+.
- More details on experimental settings.
- More quantitative and qualitative results.

We also elaborate a demo consisting of video visualizations and the illustration of our NVDS+ framework. Please refer to our project page for more video results and comparisons: https://raymondwang987.github.io/NVDS/.

APPENDIX A
MORE DETAILS ON THE VDW DATASET

A.1 Releasing of the VDW Dataset.

We have released the VDW dataset under strict conditions. We must ensure that the release won't violate any copyright requirements. To this end, we will not release any video frames or the derived data in public. Instead, we provide metadata and detailed toolkits, which can be used to reproduce VDW or generate your own data. All the metadata and toolkits are licensed under CC BY-NC-SA 4.0 [1], which can only be used for academic and research purposes. Refer to the VDW website https://raymondwang987.github.io/VDW/ for more information.

A.2 Dataset Construction

Data Acquisition and Pre-processing. Here we add more details on data acquisition and pre-processing (Sec. 4, page 7, main paper). Having obtained the raw videos, we use FFmpeg [2] and PySceneDetect [3] to split all the videos into 104,582 sequences. We manually check and remove the duplicated, chaotic, and blur scenes. Videos that are wrongly split by the scene detect tools are also removed. Finally, we reserve 32,405 videos with more than six million frames for disparity annotation.

Disparity Annotation. In Sec. 4 of the main paper, we mentioned that the disparity ground truth is obtained via sky segmentation and optical flow estimation. Here we specify the details. Compared with common practice [4], [5], we introduce a few engineering improvements to make the disparity maps more accurate. As the sky is considered to be infinitely far, pixels in the sky regions should be segmented and set to the minimum value in the disparity maps. We find that using a single segmentation model [6], [7] like prior arts [4], [5] causes errors and noises in the sky regions. Hence, we generate the sky masks in a model ensemble manner. Each frame along with its horizontally flipped copy are fed into two state-of-the-art semantic segmentation models SegFormer [8] and Mask2Former [9], which yields four sky masks in total. A pixel is considered as the sky when it is positive in more than two predicted sky masks. Besides, we also fill the connected regions with less than 50 pixels to further remove the noisy holes in the sky masks. Such ensemble strategy can improve the quality of the ground truth as shown in Fig. 1, and consequently improves the performance of the trained models, especially on skylines as shown in Fig. 8.

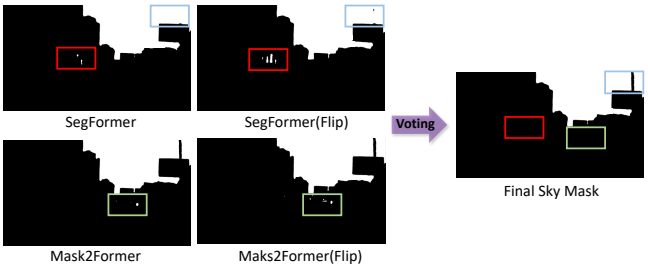

Figure 1: **Model ensemble strategy for sky segmentation on VDW dataset.** White area represents sky regions. Errors and noises in the rectangles are removed by model ensemble and voting, which improves the quality of the ground truth.

Following the practice of previous single-image depth datasets [4], [5], we adopt a state-of-the-art optical flow model GMFlow [10] to generate the ground truth disparity of the left- and right-eye views. The estimated optical flow is bidirectional. We perform a consistency check between the optical flow pairs to obtain the valid masks for training. We adopt the adaptive consistency threshold for each pixel as [11]. The ground truth of each video is normalized by its minimum and maximum disparity. Then, the disparity value is discretized into 65,535 intervals. Fig. 3 shows more examples of our VDW dataset.

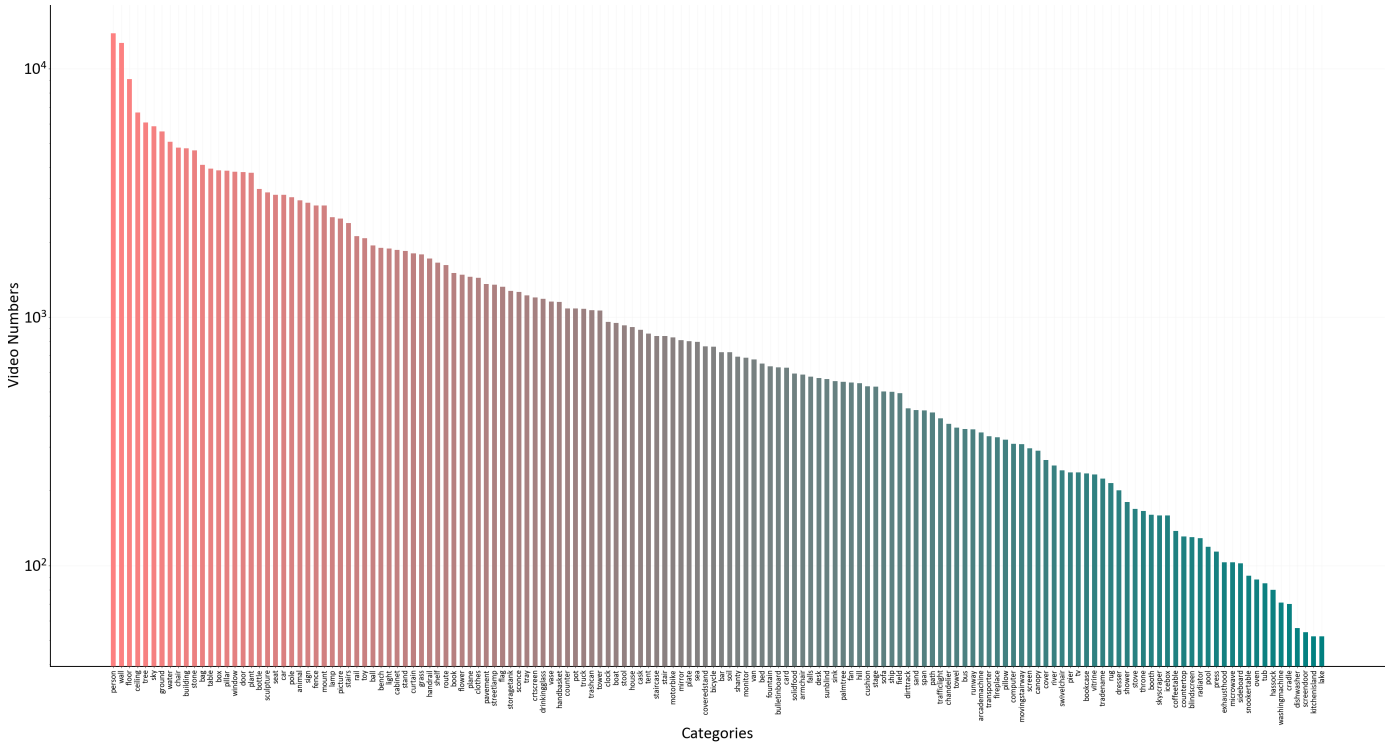

Figure 2: The statistics of the 150 semantic categories in VDW dataset.

Table 1: **Video and frame numbers statistics of VDW training set.** Our VDW dataset contains 14,203 videos from movies, animations, documentaries, and web videos.

Sources	Titles	Videos	Frames
Documentaries	Deepsea Challenge	210	38,078
	Kingdom of Plants	253	95,742
	Little Monsters	242	50,420
	Jerusalem	37	21,574
Animations	Coco	1,079	146,002
	Kung Fu Panda 3	959	68,405
Movies	Exodus: Gods and Kings	1,339	99,146
	Geostorm	857	52,028
	Hugo	301	25,091
	Mission: Impossible-Fallout	664	46,344
	Noah	1,160	85,161
	Pompeii	158	10,112
	Spider-Man: No Way Home	914	75,077
	The Legend of Tarzan	735	64,840
	The Three Musketeers	253	18,180
	Gravity	191	38,332
	Silent Hill 2	72	5,076
	Transformers: Age of Extinction	1,323	84,619
	Doctor Strange	299	23,779
	Battle of the Year	454	19,613
	Justice League	428	37,202
	The Hobbit 2	644	53,391
	The Great Gatsby	729	49,079
	Billy Lynn's Long Halftime Walk	242	29,137
Web Videos	YouTube	514	40,897
	bilibili	146	17,243
All	–	14,203	2,237,320

Invalid Sample Filtering. Having obtained the annotations, we further filter the videos that are not qualified for our dataset. According to optical flow and valid masks, samples with the following three conditions are removed: 1) more than 30% of pixels in the consistency masks are invalid; 2) more than 10% of pixels have vertical disparity larger than two pixels; 3) the average range of horizontal disparity is less than 15 pixels. Then, we manually check all the videos along with their corresponding ground truth, and remove the samples with obvious errors. Finally, we retain 14,203 videos with 2,237,320 frames in VDW dataset.

Table 2: **Video and frame numbers statistics of VDW test set.** VDW test set adopts different data sources from training data, *i.e.*, different movies, web videos, or animations.

Sources	Titles	Videos	Frames
Movies	Eternals	39	4,802
	Everest	17	2,922
	Fantastic Beasts and Where to Find Them	17	27,27
Animation	Frozen 2	10	1,098
Web Videos	bilibili	7	1,073
All	–	90	12,622

A.3 Data Statistics

Data Sources. Taking over 6 months to process, VDW training set contains 14,203 videos with 2,237,320 frames. The detailed data sources of training set and test set are listed in Table 1 and Table 2 respectively.

Frame Rates and Frame Numbers. For all our sequences, the lowest frame rate is 12 fps, the highest frame rate is 60 fps, and the average frame rate is 28.92 fps. Even some special videos, such as fast-forward or slow-motion sequences, are included in the VDW dataset. The minimum frame number is 18 while the maximum is 8,005.

Objects Presented in the VDW Dataset. To verify the diversity of objects in our videos. We conduct semantic segmentation with Mask2Former [9] trained on ADE20K [12]. All the 150 categories are covered in our dataset. The five categories that present most frequently are person (97.2%), wall (89.1%), floor (63.5%), ceiling (46.5%), and tree (42.3%).

Figure 3: **More examples of our VDW dataset.** Sky regions and invalid pixels are masked out.

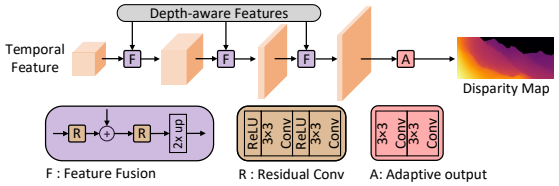

Figure 4: **The architecture of decoder for depth estimation.**

Each category can be found in at least 50 videos. Fig. 2 shows the detailed statistics of all the 150 categories.

A.4 Discussions of Data Characteristics

Animations. To enhance the diversity and generality of our VDW dataset, we include a small portion of animations (around 9% of frames), while the majority (91%) consists of real-world videos. The combination allows models to generalize well in natural scenes and produce robust predictions for animated videos. This could benefit various tasks that involve stylized videos, such as 3D video conversion, virtual reality, and video editing. Users can decide which parts to use, depending on their tasks and data requirements.

Disparity of Stereo Films. We mainly pursue data scale, diversity, and generality. Using other methods (*e.g.*, LiDAR, or Kinect) to annotate over 2 million frames in diverse scenes would incur much higher costs or may even be impractical. Thus, we adopt the disparity of stereo videos, which are more accessible and effective. However, the downside is that the disparity in stereo movies is not always trustworthy, as it could be adjusted for viewing comfort. We have implemented several measures to alleviate this problem, including removing overly unrealistic sequences and conducting rigorous data checking. Users can also combine the VDW with other datasets for training.

Appendix B
More Implementation Details for NVDS+

B.1 Decoder Architecture

Here we specify the decoder architecture for video depth estimation. The decoder architecture is illustrated in Fig. 4. To fuse the depth-aware features from the backbone [8] and temporal features from the cross-attention module, feature fusion modules (FFM) [7], [13] and skip connections are adopted. Resolutions are gradually increased while channel numbers are decreased. At last, we use an adaptive output module to adjust the channel and restore the disparity maps.

As for the decoder of video semantic segmentation, we apply the simple and common architecture as prior arts [8], [14], [15], [16].

B.2 Feature Encoder

Feature Encoders [8], [17], [18], [19], [20] possess strong scene understanding and feature encoding capabilities, because of their comprehensive structural designs and model pre-training. Compared to the details in RGB images, feature encoders [8], [17], [18], [19], [20] extract high-level scene and semantic information with large receptive fields. Therefore, encoders with different structures, *e.g.*, convolutional [18], [19], [21] or attention-based [8], [17], [20] backbones, have been widely used in dense prediction tasks such as depth estimation and semantic segmentation. Our NVDS+ also employs feature encoders [8], [18] to extract features.

To be specific, for video depth estimation, we adopt the Mit-b5 [8] in our $\text{NVDS}^{+}_{\textbf{Large}}$ model to encode depth-aware features, considering its strong performance and capacity. For the lightweight $\text{NVDS}^{+}_{\textbf{Small}}$ model, we utilize Mit-b0 [8] to achieve real-time processing. Besides, in our experiments of video semantic segmentation, we follow SSLTM [16] to leverage ResNet-50 [18] as the backbone, conducting fair comparisons with similar amounts of model parameters.

B.3 Loss Function

As mentioned in Sec. 3.2 in the main paper, the training loss for depth estimation consists of a spatial loss and a temporal loss. Here we specify the computation process.

For the spatial loss, we adopt the widely-used affinity invariant loss and gradient matching loss [5], [22] as $\mathcal{L}_s$. For the affinity invariant loss, let D and D^* denote the predicted disparity and ground truth respectively, we first calculate the scale and shift:

$$t(D) = median(D), s(D) = \frac{1}{M}\sum_{i=1}^{M}|D_i - t(D_i)|\,, \tag{1}$$

where M denotes the number of valid pixels. The prediction and the ground truth are aligned to zero translation and unit scale as follows:

$$\tilde{D} = \frac{D - t(D)}{s(D)}, \tilde{D}^* = \frac{D^* - t(D^*)}{s(D^*)}. \tag{2}$$

Then the affinity invariant loss can be formulated as:

$$\mathcal{L}_{af} = \frac{1}{M}\sum_{i=1}^{M}|\tilde{D} - \tilde{D}^*|. \tag{3}$$

Besides, we also adopt the multi-scale gradient matching loss [5], which can improve smoothness of homogeneous regions and sharpness of discontinuities in the disparity maps. The gradient matching loss is formulated as:

$$\mathcal{L}_{grad} = \frac{1}{M}\sum_{k=1}^{K}\sum_{i=1}^{M}(|\nabla_x R_i^k| + |\nabla_y R_i^k|), \tag{4}$$

where $R_i = \tilde{D}_i - \tilde{D}_i^*$, and R^k denotes the difference between the disparity maps at scale $k = 1, 2, 3, \cdots, K$ (the resolution is halved at each level). Following DPT [22], we set $K = 4$ and set the weight μ of $\mathcal{L}_{grad}$ to 0.5. The spatial loss can be expressed as:

$$\mathcal{L}_s = \mathcal{L}_{af} + \mu\mathcal{L}_{grad}, \tag{5}$$

As for the spatial loss of semantic segmentation, we adopt the widely-used cross-entropy loss for supervision.

Temporal Loss. In Sec. 3.2 of main paper, we mentioned that the temporal loss is masked with a visibility mask $O_{n\Rightarrow n-1}$ calculated from the warping discrepancy between frame F_n and the warped frame $\hat{F}_{n-1}$. This mask is obtained by:

$$O_{n\Rightarrow n-1} = \exp(-\gamma||F_n - \hat{F}_{n-1}||_2^2)\,. \tag{6}$$

We set $\gamma = 50$ and use bilinear sampling layer for warping.

Table 3: **Comparisons on VDW dataset.** The first 2 rows show the results of different single-image depth predictors. The next 5 rows contain video depth approaches. The last 2 rows consist of the results of our NVDS+. Best performance is in boldface. Second best is underlined.

Method	$\delta_1 \uparrow$	$\delta_2 \uparrow$	$\delta_3 \uparrow$	$Rel \downarrow$	$OPW \downarrow$
MiDaS-v2.1-Large [5]	0.651	0.857	0.935	0.288	0.676
DPT-Large [22]	<u>0.730</u>	<u>0.894</u>	<u>0.952</u>	<u>0.215</u>	0.470
ST-CLSTM [28]	0.477	0.709	0.838	0.521	0.448
FMNet [29]	0.472	0.716	0.837	0.514	0.402
DeepV2D [30]	0.546	0.722	0.835	0.528	0.427
WSVD [31]	0.637	0.831	0.914	0.314	0.462
Robust-CVD [32]	0.676	0.855	0.928	0.261	0.279
Ours-Large(MiDaS-v2.1-Large)	0.701	0.885	0.947	0.239	<u>0.148</u>
Ours-Large(DPT-Large)	**0.742**	**0.897**	**0.957**	**0.208**	**0.129**

Table 4: **Comparisons on the Sintel dataset.** We only report CVD [33] and Zhang *et al.* [34] on the 12 videos with valid outputs, while other methods are on the 23 videos.

Method	$\delta_1 \uparrow$	$\delta_2 \uparrow$	$\delta_3 \uparrow$	$Rel \downarrow$	$OPW \downarrow$
MiDaS-v2.1-Large [5]	0.485	0.693	0.787	0.410	0.843
DPT-Large [22]	**0.597**	<u>0.768</u>	<u>0.846</u>	<u>0.339</u>	0.612
ST-CLSTM [28]	0.351	0.571	0.706	0.517	0.585
FMNet [29]	0.357	0.579	0.712	0.513	0.521
DeepV2D [30]	0.486	0.674	0.760	0.526	0.534
WSVD [31]	0.501	0.709	0.804	0.439	0.577
CVD [33]	0.518	0.741	0.832	0.406	0.497
Robust-CVD [32]	0.521	0.727	0.833	0.422	0.475
Zhang *et al.* [34]	0.522	0.727	0.831	0.342	0.481
Ours-Large(MiDaS-v2.1-Large)	0.532	0.731	0.833	0.372	<u>0.447</u>
Ours-Large(DPT-Large)	<u>0.591</u>	**0.770**	**0.849**	**0.335**	**0.403**

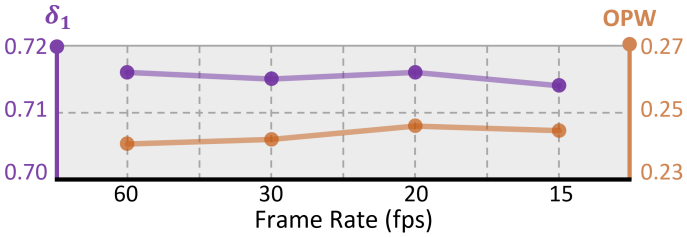

Figure 5: **Robustness across different frame rates.** The temporal window contains three reference frames. For a certain video, we conduct evaluations under frame rates from 15 to 60 fps. NVDS+ only exhibits minimal performance fluctuations, proving our robustness on various frame rates.

B.4 Depth and Disparity

Here, we illustrate the reasons for using disparity in our implementations. Firstly, our VDW dataset is annotated with the disparity from optical flow [10], making it straightforward for us to work with disparity. Secondly, we utilize different versions of MiDaS and DPT [5], [22], [23] as the initial predictors, which produce relative disparity maps. Keeping the input and output settings of NVDS+ similar to those of MiDaS and DPT [5], [22], [23], with disparity for training and inference, is convenient for the experiments. For other initial predictors that produce depth maps, their initial depth can be converted to disparity for input.

Besides, we also discuss the advantages and disadvantages of disparity and depth. Disparity is more sensitive to objects at close distances and can better distinguish between foreground objects and the background, which is beneficial for downstream tasks such as bokeh rendering [24], [25], 3D video conversion [26], and shallow depth of field effect [27]. On the other hand, depth maps can better differentiate distant objects, making them more suitable for autonomous driving tasks. Therefore, considering the applications in Sec. 5.7 of the main paper, using disparity could be more convenient for our experiments.

APPENDIX C
MORE EXPERIMENTAL RESULTS

C.1 Depth Metrics

Here we specify the evaluation metrics for depth accuracy. we adopt commonly-applied depth evaluation metrics: Mean relative error (Rel) and accuracy with threshold t.

Mean relative error (Rel): $\frac{1}{M}\sum_{i=1}^{M}\frac{||D_i - D_i^*||_1}{D_i^*}$;

Accuracy with threshold t: Percentage of D_i such that $max(\frac{D_i}{D_i^*}, \frac{D_i^*}{D_i}) = \delta < t \in \left[1.25, 1.25^2, 1.25^3\right]$, where M denotes pixel numbers, D_i and D_i^* are prediction and ground truth of pixel i.

C.2 Robustness across Various Frame Rates

The Impacts of Frame Rates. Similar to image resolution in the spatial dimension, we consider frame rates as the temporal resolution of videos. Videos with high frame rates represent small sampling intervals between consecutive frames. The inter-frame motions of moving objects and the camera could be smooth and coherent, providing sufficient temporal information. Thus, it could be easier for video depth models to predict consistent depth results. In contrast, lower frame rates represent larger sampling intervals and reduced inter-frame continuity. With lower resolution and less information in the temporal dimension, it becomes more challenging to stabilize the flickers in the predictions.

Robustness on Different Frame Rates. As illustrated in Sec. A.3, the proposed VDW dataset includes source videos with diverse frame rates. Therefore, our NVDS+ can acquire strong robustness across various frame rates. As shown in Fig. 5, we conduct evaluations under varied frame rates from 15 to 60 fps for a certain video. The temporal window is fixed with three reference frames. Our model only exhibits minimal performance fluctuations. The results prove that our setting of the temporal window is appropriate and sufficient, showing robustness against the variations of fps.

We utilize a sequence of 60 fps from the VDW dataset for the experiment in Fig. 5. Directly sampling the original video will result in varied frames for evaluation, making the metrics incomparable. Instead, we reduce the frame rates by increasing the inter-frame intervals of reference frames. For example, for the target frame n, using reference frames $i \in \{n \pm 3, n \pm 2, n \pm 1\}$ represents the original frame rate of 60 fps, while adopting $i \in \{n \pm 6, n \pm 4, n \pm 2\}$ represents 30 fps. In this way, we can still obtain the predictions for all original frames and compare the performance.

The Ideal Setting of the Temporal Window. No single setting can be optimal for all the videos. Different scenes, frame rates, objects, and motions in videos could all have an impact. For example, we cannot guarantee that three

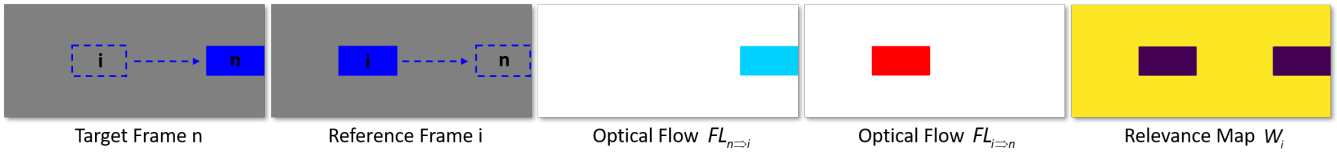

Figure 6: **The principle for flow-guided consistency fusion.** We conduct a toy experiment to illustrate the principle, proving that the flow-guided consistency fusion strategy does not introduce systematic errors in the presence of motion. For the relevance map W_i, brighter colors indicate higher values, while darker colors indicate lower fusion weights.

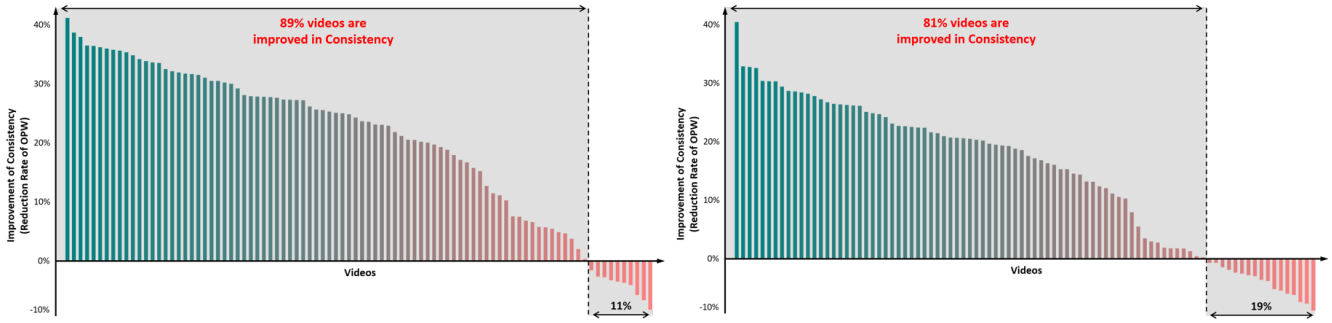

Figure 7: **The effectiveness of flow-guided consistency fusion**. We showcase the detailed statistics of consistency improvements brought by flow-guided consistency fusion over the bidirectional inference with averaging. The reduction rates of OPW are calculated for the 90 videos in the VDW test set, using MiDaS [5] or DPT [22] as varied depth predictors.

reference frames work best for all the videos, such as some special videos with extremely high frame rates, *e.g.*, over 120 fps. But this could be simply solved by adjusting the input inter-frame intervals without the need to retrain the model.

Based on Table 11 (b) and Table 12 (b) in the main paper, we utilize three reference frames with the inter-frame interval $l = 1$ as the standard setting for all our experiments on different datasets [35], [36], [37], [38], [39]. Fig. 5 proves our strong robustness across various frame rates. Thus, users can simply follow the default setting for most videos. They can also adjust these settings, *e.g.*, the inter-frame interval, according to their specific videos and applications.

C.3 Flow-Guided Consistency Fusion

We provide a toy experiment to illustrate the principle of flow-guided consistency fusion, showing that the strategy does not introduce systematic errors in the presence of motion, which is presented in Fig. 6. From reference frame i to target frame n, we assume that a deep blue rectangle (as the foreground object) is moving horizontally, while the gray areas represent the static background (e.g., a wall), as shown in the first and second columns of Fig. 6. The bidirectional optical flow $FL_{n\Rightarrow i}$ and $FL_{i\Rightarrow n}$ can be calculated and visualized in the third and fourth columns. The white areas represent the static background where the optical flow is zero. The sky blue and the red areas showcase pixels with motion (*i.e.*, large flow magnitude), representing the moving object in frame n and frame i. For the relevance map W_i, we add the magnitude of $FL_{n\Rightarrow i}$ and $FL_{i\Rightarrow n}$ to perform the negative exponential transformation as Eq. 8 of the main paper. In this way, as shown in the last column of Fig. 6, values of the relevance map W_i are very low at the positions of the moving object in both frame n and frame i. This prevents the fusion of the reference frame i and preserves the depth D_n^{bi} of the target frame n, as Eq. 9 of the main paper. The result is correct because, for the moving rectangle, foreground and background pixels are misaligned, and the reference frames should not be fused.

Consequently, flow-guided consistency fusion does not introduce systematic errors in the presence of motion. For moving objects and regions, the depth of reference frames tends not to be fused due to misalignment. The original bidirectional depth D_n^{bi} of the target frame will be preserved.

To further prove the effectiveness, we showcase the detailed statistics of consistency improvements brought by flow-guided consistency fusion over the bidirectional inference with averaging. As shown in Fig. 7, the reduction rates of OPW are calculated for the 90 videos in the VDW test set. With MiDaS [5] or DPT [22] as the depth predictor, 89% and 81% of all the videos achieve better consistency (above the X-axis) respectively. The depth accuracy is also maintained as proved by Table 5 of the main paper. Overall, bidirectional inference and flow-guided consistency fusion are simple but effective methods for improving consistency without introducing systematic errors, because of the adaptive fusion based on bidirectional optical flow, motion amplitude, and relevance maps. The experiments demonstrate that our approach only requires optical flow and works well for most testing videos.

On the other hand, the relations among camera motion, object motion, and depth variations could be complex in different scenarios. Our assumptions and methods could not fully cover all corner cases due to the diversity of real-world videos. We also try to explore some mechanisms that could be more comprehensive theoretically. However, these techniques introduce new problems in practice. For instance, camera motion compensation [40] can be adopted to decouple the camera and object motion. But their reliance on camera parameters (*e.g.*, the FOV) is impractical for in-the-wild videos, leading to failure cases and artifacts. There-

Table 5: **Zero-shot evaluations and model finetuning.** DPT-Large [22] and MiDaS-v2.1-Large [5] are adopted as different depth predictors. We report the results of $\text{NVDS}^{+}_{\textbf{Large}}$ with zero-shot evaluations (*i.e.*, only trained on the VDW dataset) and model finetuning on the NYUDV2 [35] dataset.

Method	$\delta_1 \uparrow$	$OPW \downarrow$
DPT [22]	0.928	0.811
Zero-Shot(DPT)	0.930	0.351
Finetune(DPT)	**0.950**	**0.339**

(a) DPT Initialization

Method	$\delta_1 \uparrow$	$OPW \downarrow$
MiDaS [5]	0.910	0.862
Zero-Shot(MiDaS)	0.919	**0.332**
Finetune(MiDaS)	**0.941**	0.347

(b) MiDaS Initialization

Table 6: **Runtime of the lightweight $\text{NVDS}^{+}_{\textbf{Small}}$ model.** We report the runtime of each component to predict one 896×384 frame on an NVIDIA RTX A6000 GPU. The $\text{NVDS}^{+}_{\textbf{Small}}$ model shows high efficiency for real-time applications.

Module	Component	Runtime (ms)	Overall (ms)
Depth Predictor	DPT-Swin2-Tiny [23]	24.18	24.18
	MiDaS-v2.1-Small [5]	22.35	22.35
Stabilization Network	Feature Encoder	1.34	6.29
	Cross-attention	0.89	
	Decoder	4.06	

fore, we use bidirectional optical flow to perform motion compensation in the flow-guided consistency fusion.

C.4 Zero-shot Evaluations and Model Finetuning

As presented in Table 5, we report the results of $\text{NVDS}^{+}_{\textbf{Large}}$ with zero-shot evaluations (*i.e.*, only trained on the VDW dataset) and model finetuning on the NYUDV2 [35] dataset. For zero-shot evaluations, our model improves both the temporal consistency and depth accuracy over the depth predictors [5], [22], showing the generalization ability of our method. Besides, the finetuning can further improve the depth accuracy for closed-domain applications, *e.g.*, the static indoor scenes of the NYUDV2 [35] dataset.

C.5 Runtime Analysis

For the $\text{NVDS}^{+}_{\textbf{Small}}$ model, we report the runtime of each component in milliseconds (ms), including the depth predictors [5], [23], the feature encoder, the cross-attention module, and the decoder. The stabilization network achieves faster inference speed than lightweight depth predictors [5], [23]. Combining all components, $\text{NVDS}^{+}_{\textbf{Small}}$ can still achieve real-time processing of over 30 fps.

C.6 More Quantitative Comparisons

In the main paper, only δ_1, Rel, and OPW are reported. The additional results on the VDW and the Sintel [38] dataset are shown in Table 3 and Table 4. Besides, as CVD [33] and Zhang *et al.* [34] cannot produce results on 11 of 23 videos in Sintel [38] dataset, we additionally report the results on the other 12 videos in Table 7.

C.7 More Qualitative Results.

We show more visual comparisons in Fig. 8 and 9. We draw the scanline slice over time. Fewer zigzagging pattern means better consistency. Please refer to our demo video and project page for more video results and comparisons.

Table 7: **Comparisons on the 12 videos of Sintel [38] dataset.** We test the 12 videos that CVD [33] and Zhang *et al.* [34] can produce results for comparisons.

Method	$\delta_1 \uparrow$	$\delta_2 \uparrow$	$\delta_3 \uparrow$	$Rel \downarrow$	$OPW \downarrow$
MiDaS-v2.1-Large [5]	0.670	0.853	0.902	0.246	0.712
DPT-Large [22]	**0.747**	0.874	0.917	**0.196**	0.671
ST-CLSTM [28]	0.477	0.711	0.827	0.366	0.547
FMNet [29]	0.492	0.728	0.825	0.363	0.516
DeepV2D [30]	0.509	0.735	0.827	0.384	0.575
CVD [33]	0.518	0.741	0.832	0.406	0.497
Zhang *et al.* [34]	0.522	0.727	0.831	0.342	0.481
WSVD [31]	0.621	0.822	0.891	0.305	0.581
Robust-CVD [32]	0.673	0.848	0.888	0.284	0.447
Ours-Large(MiDaS-v2.1-Large)	0.701	0.867	0.918	0.215	0.403
Ours-Large(DPT-Large)	0.741	**0.878**	**0.925**	0.201	**0.392**

APPENDIX D
IMAGE ATTRIBUTION

We properly attribute the sources of all images and figures throughout the paper, as presented in Table 8.

Table 8: **Attribution of the images in our main manuscript and supplement.** We report the image attribution of all figures throughout our paper. We also specify the images from movies, documentaries, and animations with their IMDB movie numbers. Web videos and public datasets do not have IMDB numbers, with – as representation.

Figures	Types	Attribution	IMDB Numbers
		Main Manuscript	
Fig. 1	Movie	Everest	2719848
Fig. 2	Movie	Everest	2719848
Fig. 4	Animation	Frozen 2	4520988
	Movie	Eternals	9032400
Fig. 5	Web Video	YouTube	–
	Documentary	Jerusalem	2385006
	Documentary	Kingdom of Plants	2117380
	Animation	Kung Fu Panda 3	2267968
	Movie	The Hobbit 2	1170358
	Movie	The Great Gatsby	1343092
Fig. 7	Movie	Eternals	9032400
	Animation	Frozen 2	4520988
Fig. 8	Public Dataset	DAVIS [39]	–
Fig. 9	Movie	Eternals	9032400
Fig. 10	Public Dataset	CityScapes [41]	–
Fig. 11	Public Dataset	NYUDV2 [35]	–
Fig. 12	Movie	Eternals	9032400
	Movie	Everest	2719848
	Web Video	NSFF [42] Demo	–
Fig. 13	Movie	Eternals	9032400
	Movie	Fantastic Beasts and Where to Find Them	3183660
Fig. 14	Public Dataset	Sintel [38]	–
		Supplementary Document	
Fig. 3	Web Video	YouTube	–
	Web Video	bilibili	–
	Documentary	Jerusalem	2385006
	Documentary	Kingdom of Plants	2117380
	Documentary	Little Monsters	11019830
	Documentary	Deepsea Challenge	2332883
	Animation	Kung Fu Panda 3	2267968
	Animation	Coco	2380307
	Movie	The Great Gatsby	1343092
	Movie	Mission: Impossible-Fallout	4912910
	Movie	Doctor Strange	1211837
	Movie	Transformers: Age of Extinction	2109248
	Movie	The Legend of Tarzan	0918940
	Movie	Exodus: Gods and Kings	1528100
Fig. 8	Web Video	YouTube	–
	Web Video	bilibili	–
	Movie	Eternals	9032400
	Movie	Fantastic Beasts and Where to Find Them	3183660
Fig. 9	Public Dataset	Sintel [38]	–

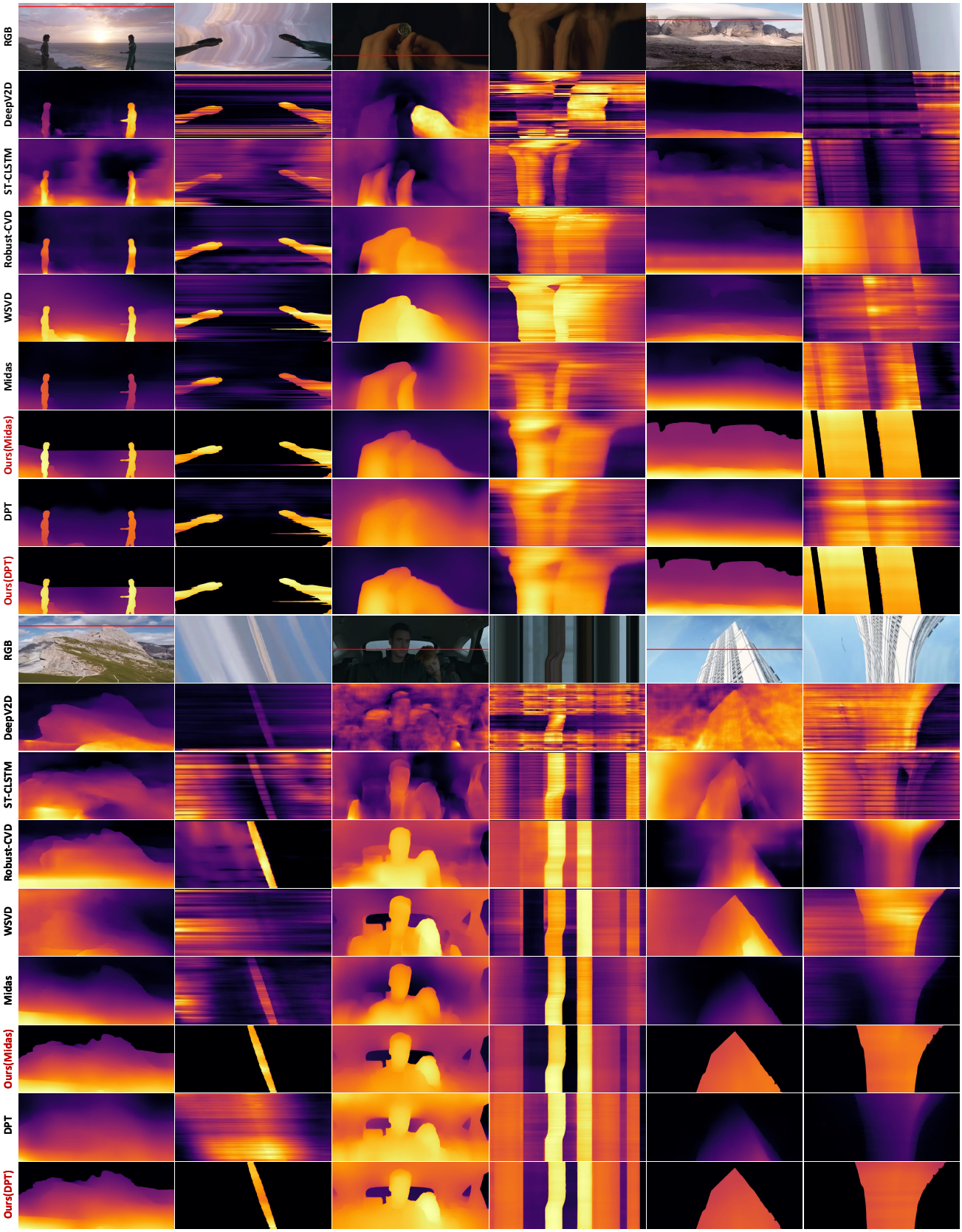

Figure 8: **More qualitative results on natural scenes.** The first image in each pair is the RGB frame, while the second is the scanline slice over time. Fewer zigzagging pattern means better consistency.

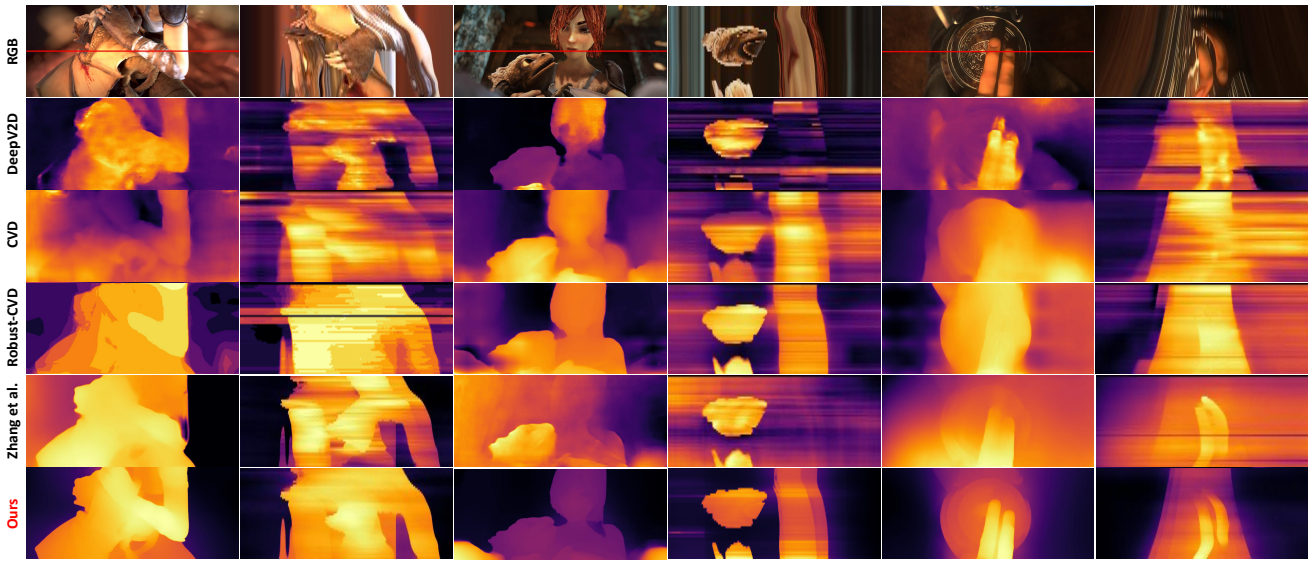

Figure 9: **Qualitative results on Sintel [38] dataset.** We compare the results of DeepV2D [30], CVD [33], Robust-CVD [32], and Zhang *et al.* [34]. Without relying on test-time training [32], [33], [34], we conduct zero-shot evaluations on Sintel [38] and achieve significantly better performance than those test-time-training-based methods [32], [33], [34].

REFERENCES

[1] Creative Commons organization, "Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International," https://creativecommons.org/licenses/by-nc-sa/4.0/legalcode.

[2] FFmpeg developers, "FFmpeg," https://ffmpeg.org, [Online; Accessed 2022].

[3] PySceneDetect developers, "PySceneDetect," http://scenedetect.com, [Online; Accessed 2022].

[4] K. Xian, C. Shen, Z. Cao, H. Lu, Y. Xiao, R. Li, and Z. Luo, "Monocular relative depth perception with web stereo data supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 311–320.

[5] 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 transactions on pattern analysis and machine intelligence*, vol. 44, no. 03, pp. 1623–1637, 2020.

[6] S. R. Bulo, L. Porzi, and P. Kontschieder, "In-place activated batchnorm for memory-optimized training of dnns," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 5639–5647.

[7] G. Lin, A. Milan, C. Shen, and I. Reid, "Refinenet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1925–1934.

[8] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: Simple and efficient design for semantic segmentation with transformers," *Advances in neural information processing systems*, vol. 34, pp. 12 077–12 090, 2021.

[9] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 1290–1299.

[10] H. Xu, J. Zhang, J. Cai, H. Rezatofighi, and D. Tao, "Gmflow: Learning optical flow via global matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 8121–8130.

[11] S. Meister, J. Hur, and S. Roth, "Unflow: Unsupervised learning of optical flow with a bidirectional census loss," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[12] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene parsing through ADE20K dataset," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 633–641.

[13] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2117–2125.

[14] G. Sun, Y. Liu, H. Ding, T. Probst, and L. Van Gool, "Coarse-to-fine feature mining for video semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 3126–3137.

[15] G. Sun, Y. Liu, H. Tang, A. Chhatkuli, L. Zhang, and L. Van Gool, "Mining relations among cross-frame affinities for video semantic segmentation," in *European Conference on Computer Vision (ECCV)*. Springer, 2022, pp. 522–539.

[16] J. Lao, W. Hong, X. Guo, Y. Zhang, J. Wang, J. Chen, and W. Chu, "Simultaneously short-and long-term temporal modeling for semi-supervised video semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 14 763–14 772.

[17] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2020.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[19] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1492–1500.

[20] H. Fan, B. Xiong, K. Mangalam, Y. Li, Z. Yan, J. Malik, and C. Feichtenhofer, "Multiscale vision transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 6824–6835.

[21] H. Zhang, C. Wu, Z. Zhang, Y. Zhu, H. Lin, Z. Zhang, Y. Sun, T. He, J. Mueller, R. Manmatha *et al.*, "Resnest: Split-attention networks," *arXiv preprint arXiv:2004.08955*, 2020.

[22] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 12 179–12 188.

[23] R. Birkl, D. Wofk, and M. Müller, "Midas v3. 1–a model zoo for robust monocular relative depth estimation," *arXiv preprint arXiv:2307.14460*, 2023.

[24] J. Peng, Z. Cao, X. Luo, H. Lu, K. Xian, and J. Zhang, "Bokehme: When neural rendering meets classical rendering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 16 283–16 292.

[25] X. Zhang, K. Matzen, V. Nguyen, D. Yao, Y. Zhang, and R. Ng, "Synthetic defocus and look-ahead autofocus for casual videography," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 4, 2019.

[26] K. Karsch, C. Liu, and S. B. Kang, "Depth transfer: Depth extraction from video using non-parametric sampling," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 11, pp. 2144–2158, 2014.

[27] L. Wang, X. Shen, J. Zhang, O. Wang, Z. Lin, C.-Y. Hsieh, S. Kong, and H. Lu, "Deeplens: shallow depth of field from a single image," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 6, 2018.

[28] H. Zhang, C. Shen, Y. Li, Y. Cao, Y. Liu, and Y. Yan, "Exploiting temporal consistency for real-time video depth estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 1725–1734.

[29] Y. Wang, Z. Pan, X. Li, Z. Cao, K. Xian, and J. Zhang, "Less is more: Consistent video depth estimation with masked frames modeling," in *Proceedings of the 30th ACM International Conference on Multimedia*, ser. MM '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 6347–6358. [Online]. Available: https://doi.org/10.1145/3503161.3547978

[30] Z. Teed and J. Deng, "Deepv2d: Video to depth with differentiable structure from motion," in *International Conference on Learning Representations*, 2019.

[31] C. Wang, S. Lucey, F. Perazzi, and O. Wang, "Web stereo video supervision for depth prediction from dynamic scenes," in *IEEE International Conference on 3D Vision (3DV)*. IEEE, 2019, pp. 348–357.

[32] J. Kopf, X. Rong, and J.-B. Huang, "Robust consistent video depth estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 1611–1621.

[33] X. Luo, J.-B. Huang, R. Szeliski, K. Matzen, and J. Kopf, "Consistent video depth estimation," *ACM Transactions on Graphics (ToG)*, vol. 39, no. 4, pp. 71–1, 2020.

[34] Z. Zhang, F. Cole, R. Tucker, W. T. Freeman, and T. Dekel, "Consistent depth of moving objects in video," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 4, pp. 1–12, 2021.

[35] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *European Conference on Computer Vision (ECCV)*. Springer, 2012, pp. 746–760.

[36] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[37] Y. Wang, M. Shi, J. Li, Z. Huang, Z. Cao, J. Zhang, K. Xian, and G. Lin, "Neural video depth stabilizer," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 9466–9476.

[38] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, "A naturalistic open source movie for optical flow evaluation," in *European Conference on Computer Vision (ECCV)*. Springer, 2012, pp. 611–625.

[39] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, and L. Van Gool, "The 2017 DAVIS challenge on video object segmentation," *arXiv preprint arXiv:1704.00675*, 2017.

[40] Camocomp developers, "CAmera MOtion COMPensation," https://github.com/daien/camocomp, [Online; Accessed 2024].

[41] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 3213–3223.

[42] Z. Li, S. Niklaus, N. Snavely, and O. Wang, "Neural scene flow fields for space-time view synthesis of dynamic scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 6498–6508.