Neural Video Depth Stabilizer

Yiran Wang\textsuperscript{1}  Min Shi\textsuperscript{1}  Jiaqi Li\textsuperscript{1}  Zihao Huang\textsuperscript{1}  Zhiguo Cao\textsuperscript{1}  Jianming Zhang\textsuperscript{2}  Ke Xian\textsuperscript{3}  Guosheng Lin\textsuperscript{3}

\textsuperscript{1}School of Artificial Intelligence and Automation, Huazhong University of Science and Technology
\textsuperscript{2}Adobe Research
\textsuperscript{3}S-Lab, Nanyang Technological University

\{wangyiran, min_shi, lijiaqi_mail, zihaohuang, zgcao\}@hust.edu.cn
jianmzha@adobe.com, \{ke.xian, gslin\}@ntu.edu.sg

https://github.com/RaymondWang987/NVDS

Abstract

Video depth estimation aims to infer temporally consistent depth. Some methods achieve temporal consistency by finetuning a single-image depth model during test time using geometry and re-projection constraints, which is inefficient and not robust. An alternative approach is to learn how to enforce temporal consistency from data, but this requires well-designed models and sufficient video depth data. To address these challenges, we propose a plug-and-play framework called Neural Video Depth Stabilizer (NVDS) that stabilizes inconsistent depth estimations and can be applied to different single-image depth models without extra effort. We also introduce a large-scale dataset, Video Depth in the Wild (VDW), which consists of 14,203 videos with over two million frames, making it the largest natural-scene video depth dataset to our knowledge. We evaluate our method on the VDW dataset as well as two public benchmarks and demonstrate significant improvements in consistency, accuracy, and efficiency compared to previous approaches. Our work serves as a solid baseline and provides a data foundation for learning-based video depth models. We will release our dataset and code for future research.

1. Introduction

Monocular video depth estimation is a prerequisite for various video applications, e.g., bokeh rendering [24, 25, 47], 2D-to-3D video conversion [12], and novel view synthesis [16, 17]. An ideal video depth model should output depth results with both spatial accuracy and temporal consistency. Although the spatial accuracy has been greatly improved by recent advances in single-image depth models [15, 27, 28, 39, 45] and datasets [18, 41, 42], how to obtain temporal consistency, i.e., removing flickers in the predicted depth sequences, is still an open question. The prevailing video depth approaches [13, 23, 48] require test-time training (TTT). During inference, a single-image depth model is finetuned on the testing video with geometry constraints and pose estimation. These TTT-based methods have two main issues: limited robustness and heavy computation overhead. Due to the heavy reliance on camera poses, e.g., CVD [23] shows erroneous predictions and robust-CVD [13] produces obvious artifacts for many videos when camera poses [13, 29] are inaccurate. Moreover, test-time
training is extremely time-consuming. CVD [23] takes 40 minutes for 244 frames on four NVIDIA Tesla M40 GPUs.

This motivates us to build a learning-based model that learns to enforce temporal consistency from video depth data. However, like all the deep-learning models, learning-based paradigm requires proper model design and sufficient training data. Previous learning-based methods [5, 34, 40, 46] show worse performance than the TTT-based ones. Video depth data is also limited in scale and diversity.

To address the two aforementioned challenges, we first propose a flexible learning-based framework termed Neural Video Depth Stabilizer (NVDS), which can be directly applied to different single-image depth models. NVDS contains a depth predictor and a stabilization network. The depth predictor can be any off-the-shelf single-image depth model. Different from the previous learning-based methods [5, 34, 40, 46] that function as stand-alone models, NVDS is a plug-and-play refiner for different depth predictors. Specifically, the stabilization network processes initial flickering disparity estimated by the depth predictor and outputs temporally consistent results. Therefore, our framework can benefit from the cutting-edge depth models without extra effort. As for the design of stabilization network, inspired by attention [35] in other video tasks [1, 14, 22, 33], we adopt a cross-attention module in our framework. Each frame can attend relevant information from adjacent frames for temporal consistency. We also design a bidirectional inference strategy to further improve the consistency. As shown in Fig. 1(a), our NVDS outperforms the previous approaches in terms of consistency, accuracy, and efficiency significantly.

Moreover, we collect a large-scale natural-scene video depth dataset, Video Depth in the Wild (VDW), to support the training of robust learning-based models. Current video depth datasets are mostly closed-domain [7, 9, 31, 32, 37]. A few in-the-wild datasets [4, 36, 38] are still limited in quantity, diversity, and quality, e.g., Sintel [4] only contains 23 animated videos. In contrast, our VDW dataset contains 14,203 stereo videos of over 200 hours and 2.23M frames from four different data sources, including movies, animations, documentaries, and web videos. We adopt a rigorous data annotation pipeline to obtain high-quality disparity ground truth for these data. As shown in Fig. 1(b), to the best of our knowledge, VDW is the largest in-the-wild video depth dataset with diverse scenes.

We conduct evaluations on the VDW and two public benchmarks: Sintel [4] and NYUDV2 [31]. Our method achieves state-of-the-art in both the accuracy and the consistency. We also fit three different depth predictors [27, 28, 45] into our framework and evaluate them on NYUDV2 [31]. The results demonstrate the flexibility and effectiveness of our plug-and-play manner. Our main contributions can be summarized as follows:

• We propose a plug-and-play and bidirectional learning-based framework termed Neural Video Depth Stabilizer (NVDS), which can be directly adapted to different single-image depth predictors to remove flickers.

• We propose VDW dataset, which is currently the largest video depth dataset in the wild with the most diverse video scenes.

2. Related Work

Consistent Video Depth Estimation. In addition to predicting spatial-accurate depth, the core task of consistent video depth estimation is to achieve temporal consistency, i.e., removing the flickering effects between consecutive frames. Current video depth estimation approaches can be categorized into test-time training (TTT) ones and learning-based ones. TTT-based methods train an off-the-shelf single-image depth estimation model on testing videos during inference with geometry [13, 23, 48] and pose [13, 29, 30] constraints. The test-time training can be time-consuming. For example, as illustrated by CVD [23], their method takes 40 minutes on 4 NVIDIA Tesla M40 GPUs to process a video of 244 frames. Besides, TTT-based approaches are not robust on in-the-wild videos as they heavily rely on camera poses, which are not reliable for natural scenes. In contrast, the learning-based approaches train video depth models on video depth datasets by spatial and temporal supervision. ST-CLSTM [46] adopts long short-term memory (LSTM) to model temporal relations. FMNet [40] restores the depth of masked frames by the unmasked ones with convolutional self-attention [22]. Cao et al. adopt a spatial-temporal propagation network trained by knowledge distillation [10, 21]. However, those methods are independent and cannot refine the results from single-image depth models for consistency. Their performance on consistency and accuracy is also limited. For example, as shown by [40], ST-CLSTM [14] only exploits subsequences of several frames and produces obvious flickers in the outputs. In this paper, we propose a novel framework called Neural Video Depth Stabilizer (NVDS), which can be directly adapted to any off-the-shelf single-image depth models in a plug-and-play manner.

Video Depth Datasets. According to the scenes of samples, existing video depth datasets can be categorized into closed-domain datasets and natural-scene datasets. Closed-domain datasets only contain samples in certain scenes, e.g., indoor scenes [7, 31, 37], office scenes [32], and autonomous driving [9]. To enhance the diversity of samples, natural-scene datasets are proposed, which use computer-rendered videos [4, 38] or crawl stereoscopic videos from YouTube [36]. However, the scene diversity and scale of these datasets are still very limited for training robust video depth estimation models that can predict consistent depth in
3. Neural Video Depth Stabilizer

As shown in Fig. 2, the proposed Neural Video Depth Stabilizer (NVDS) consists of a depth predictor and a stabilization network. The depth predictor predicts the initial flickering disparity for each frame. The stabilization network predicts the initial flickering disparity for each frame. The stabilization network predicts the initial flickering disparity for each frame. The stabilization network refines the flickering disparity maps into temporally consistent ones. The stabilization network functions in a sliding window manner: the frame to be predicted fetches information from adjacent frames for stabilization. During inference, our NVDS framework can be directly adapted to any off-the-shelf depth predictors in a plug-and-play manner. We also devise bidirectional inference to further improve consistency.

3.1. Stabilization Network

The stabilization network takes RGB frames along with initial disparity maps as inputs. A backbone [43] encodes the input sequences into depth-aware features. The next step is to build inter-frame correlations. We use a cross-attention module to refine the depth-aware features with temporal information from relevant frames. Finally, the refined features are fed into a decoder which restores disparity maps with temporal consistency.

Depth-aware Feature Encoding. Stabilization network works in a sliding-window manner: each frame refers to a few previous frames, which are denoted as reference frames, to stabilize the depth. We denote the frame to be predicted as the target frame. Each sliding window consists of four frames.

Due to the varied scale and shift of disparity maps produced by different depth predictors, the initial disparity maps within a sliding window $F = \{F_1, F_2, F_3, F_4\}$ should be normalized into $F_{\text{norm}}$:

$$F_{\text{norm}}^i = \frac{F_i - \min(F)}{\max(F) - \min(F)}, i \in \{1, 2, 3, 4\}. \quad (1)$$

Then, the normalized disparity maps are concatenated with the RGB frames to form a RGB-D sequence. We use a transformer backbone [43] to encode the RGB-D sequence into depth-aware feature maps.

Cross-attention Module. With the depth-aware features, the subsequent phase entails the establishment of inter-frame correlations. We leverage a cross-attention module to build temporal and spatial dependencies across pertinent video frames. Specifically, in the cross-attention module, the target frame selectively attends the relevant features in the reference frames to facilitate depth stabilization. Pixels in the target frame feature maps serve as the query in the
cross-attention operation [35], while the keys and values are generated from the reference frames. Computational cost can become prohibitively high when employing cross-attention for each position in depth-aware features. Hence, we utilize a patch merging strategy [8] to down-sample the target feature map. Besides, we also restrict the cross-attention into a local window, whereby each token in the target features can only attend a local window in the reference frames. Let \( T \) denote the depth-aware feature of the target frame, while \( R_{1}, R_{2} \) and \( R_{3} \) represent the features for the three reference frames. \( T \) is partitioned into \( 7 \times 7 \) patches with no overlaps; each patch is merged into one token \( t \in \mathbb{R}^{c} \), where \( c \) is the dimension. For each \( t \), we conduct a local window pooling on \( R_{1}, R_{2}, \) and \( R_{3} \) and stack the pooling results into \( R_{p} \in \mathbb{R}^{c \times 3} \). Then, the cross-attention is computed as:

\[
t' = \text{softmax} \left( \frac{W_{q}t (W_{k}R_{p})^T}{\sqrt{c}} \right)W_{v}R_{p},
\]

where \( W_{q}, W_{k}, \) and \( W_{v} \) are learnable linear projections. The cross-attention layer is incorporated into a standard transformer block [35] with residual connection and multi-layer perceptron (MLP). We denote the resulting target feature by \( \text{Transformer block} \) with residual connection and multi-

layer perceptron (MLP).

3.2. Training the Stabilization Network

In the training phase, only the stabilization network is optimized. The depth predictor is the freezed pre-trained DPT-L [27]. For the stabilization network, we apply spatial and temporal loss that supervises the depth accuracy and temporal consistency respectively. The training loss can be formulated as:

\[
L = L_{s}(n-1) + L_{s}(n) + \lambda L_{l}(n, n-1),
\]

where \( L_{s}(n-1) \) and \( L_{s}(n) \) denote the spatial loss of frame \( n-1 \) and \( n \) respectively. \( L_{l}(n, n-1) \) denotes the temporal loss between frame \( n-1 \) and \( n \).

We adopt the widely-used affinity invariant loss and gradient matching loss [27, 28] as the spatial loss \( L_{s} \). As for the temporal loss, we adopt the optical flow based warping loss [5, 40] to supervise temporal consistency:

\[
L_{l}(n, n-1) = \frac{1}{M} \sum_{j=1}^{M} O_{n \rightarrow n-1}^{(j)} ||D_{n-1}^{(j)} - \hat{D}_{n-1}^{(j)}||_1 ,
\]

where \( \hat{D}_{n-1} \) is the predicted disparity \( D_{n-1} \) warped by the optical flow \( FL_{n \rightarrow n-1} \). In our implementation, we adopt the GMFlow [44] for optical flow. \( O_{n \rightarrow n-1} \) is the mask calculated as [5, 40] and \( M \) denotes pixel numbers. See supplementary for more details on loss functions.

3.3. Bidirectional Inference

Expanding the temporal receptive range can be beneficial for consistency, e.g., adding more former or latter reference frames. However, directly training the stabilization network with bidirectional reference frames will introduce large training burdens. To remedy this, we only train the stabilization network with the former three reference frames. To further enlarge the temporal receptive field and enhance consistency, we introduce a bidirectional inference strategy.

Unlike the training phase, during inference, both the former and latter frames will be used as the reference frames. An additional sliding window is added, where the reference frames are the subsequent three frames of the target. Let us define the stabilizing process as a function \( S(V_{t}, V_{r}) \), where \( V_{t} \) and \( V_{r} \) denotes the target RGB-D frame and the reference frames set. When denoting the RGB-D sequence as \( \{V_{j} | j \in \{1, 2, \ldots, N\} \} \), \( N \) represents frame numbers of a certain video, using this additional sliding window for stabilization can be formulated as:

\[
D^{\text{post}}_{n} = S(V_{n}, \{V_{n+1}, V_{n+2}, V_{n+3}\}) ,
\]

where \( V_{n} \) denotes the target frame. Likewise, using the original sliding window for stabilization can be denoted by:

\[
D^{\text{pre}}_{n} = S(V_{n}, \{V_{n-1}, V_{n-2}, V_{n-3}\}) .
\]

We ensemble the bidirectional results for a larger temporal receptive field as:

\[
D^{bi}_{n} = \frac{(D^{\text{pre}}_{n} + D^{\text{post}}_{n})}{2} .
\]

\( D^{bi}_{n} \) denotes the final disparity prediction of the \( n^{th} \) frame (target frame). This bidirectional manner can further improve the temporal consistency as demonstrated in Sec. 5.4.

Note that, the cross-attention module is shared by the two sliding windows for inference. Besides, the initial disparity maps and depth-aware features are pre-computed. Hence, the bidirectional inference only increases the inference time by 30% compared with single-direction inference and brings no extra computation for the training process.

In addition to bidirectional inference, the depth predictor can be reconfigurable during inference. For example, simply using a more advanced model NeWCRFs [45] as the depth predictor can obtain performance gain without extra training, as shown in Table 5. As the depth accuracy can be inherited from state-of-the-art depth predictor, our neural video depth stabilizer (NVDS) framework can focus on the learning of depth stabilization and combine the depth accuracy with temporal consistency.

3.4. Implementation Details

Model Architecture. We use the DPT-L [27], Midas-v2 [28], and NeWCRFs [45] as the single-image depth predictors during inference, while only the disparity maps from
The 3D video depth models trained on TartanAir cannot predict satisfactory results for scenes with moving people as such scenes are rare. Our VDW dataset shows advantages in diversity and quantity.

DPT-L are used during training. Midas-v2 [28] is the same depth model as TTT-based methods [13, 23, 48] for fair comparisons. Mit-b5 [43] is adopted as the backbone to depth model as TTT-based methods [13, 23, 48] for fair comparisons. Mit-b5 [43] is adopted as the backbone to depth model as TTT-based methods [13, 23, 48] for fair comparisons.

Training Recipe. All frames are resized so that the shorter side equals 384, and then randomly cropped to $384 \times 384$ for training. In each epoch, we randomly sample 72,000 input sequences. Note that the sampled frames in each epoch does not overlap. We use Adam optimizer to train the model for 30 epochs with a batchsize of 9. The initial learning rate is set to $6e - 5$ and decreases by $1e - 5$ for every five epochs. When finetuning our model on NYUDV2 [31], we use a learning rate of $1e - 5$ for only one epoch. In all experiments, the temporal loss weight $\lambda$ is set to 0.2.

4. VDW Dataset

As mentioned in Sec. 1, current video depth datasets are limited in both diversity and volume. To compensate for the data shortage and boost the performance of learning-based video depth models, we elaborate a large-scale natural-scene dataset, Video Depth in the Wild (VDW). To our best knowledge, our VDW dataset is currently the largest video depth dataset with the most diverse video scenes.

Dataset Construction. We collect stereo videos from four data sources: movies, animations, documentaries, and web videos. A total of 60 movies, animations, and documentaries in Blu-ray format are collected. We also crawl 739 web stereo videos from YouTube with the keywords such as “stereoscopic” and “stereo”. To balance the realism and diversity, only 24 movies, animations, and documentaries are retained. For instance, “Seven Wonders of the Solar System” is removed as it contains many virtual scenes. The disparity ground truth is generated with two main steps: sky segmentation and optical flow estimation. A model ensemble manner is adopted to remove errors and noises in the sky masks, which can improve the quality of the ground truth and the performance of the trained models, especially on sky regions as shown in Fig. 5. A state-of-the-art optical flow model GMFlow [44] is used to generate the disparity ground truth. Finally, a rigorous data cleaning procedure is conducted to filter the videos that are not qualified for our dataset. Fig. 3 shows some examples of our VDW dataset.

Dataset Statistics. VDW dataset contains 14,203 videos with a total of 2,237,320 frames. The total data collection and processing time takes over six months and about 4,000 man-hours. To verify the diversity of scenes and entities in our dataset, we conduct semantic segmentation by Mask2Former [6] trained on ADE20k [49]. Refer to our supplementary for more detailed construction process and data statistics.

Comparisons with Other Datasets. As shown in Table 1, VDW dataset shows advantages in diversity and quantity.
the proposed VDW dataset has significantly larger numbers of video scenes. Compared with the closed-domain datasets [7, 9, 31, 32, 37], the videos of VDW are not restricted to a certain scene, which is more helpful to train a robust video depth model. For the natural-scene datasets, our dataset has more than ten times the number of videos as the previous largest dataset WSVD [36]. Although WSVD [36] has 1.5M frames, the scenes (video numbers) are limited. Midas [28] also proposes their 3D Movies dataset with in-the-wild images and disparity. Compared with the 3D Movies dataset of Midas [28], VDW differs in two main aspects: (1) accessibility; and (2) dataset scale and format. Their 3D Movies dataset [28] is not released and only contains 75k images. In contrast, VDW contains 14,203 videos with 2.237M frames. It is also worth noticing that our VDW dataset has higher resolution and a rigorous data annotation and cleaning pipeline. We only collect videos with resolutions over 1080p and crop all our videos to 1880 × 800 to remove black bars and subtitles. See supplementary for more statistics and construction process.

5. Experiments

To prove the effectiveness of our framework Neural Video Depth Stabilizer (NVDS), we conduct experiments on 3 different datasets, which contain videos for real-world and synthetic, static and dynamic, indoor and outdoor.

5.1. Datasets and Evaluation Protocol

**VDW Dataset.** We use the proposed VDW as the training data for its diversity and quantity on natural scenes. We also evaluate the previous video depth approaches on the test split of VDW, serving as a new video depth benchmark. **Sintel Dataset.** Following [13, 48], we use the final version of Sintel [4] to demonstrate the generalization ability of our NVDS. We conduct zero-shot evaluations on Sintel [4]. All learning-based methods are not finetuned on Sintel dataset. **NYUDV2 Dataset.** Except for natural scenes, a closed-domain NYUDV2 [31] is adopted for evaluation. We pre-train the stabilization network on VDW and finetune the model on NYUDV2 [31] dataset. Besides, we also test our NVDS on DAVIS [26] for qualitative comparisons.

**Evaluation Metrics.** We evaluate both the depth accuracy and temporal consistency of different methods. For the temporal consistency metric, we adopt the optical flow based warping metric (\(OPW\)) following FMNet [40], which can be computed as:

\[
OPW = \frac{1}{N - 1} \sum_{n=2}^{N} \mathcal{L}^t(n, n - 1).
\]

We report the average \(OPW\) of all the videos in the testing sets. As for the depth metrics, we adopt the commonly-applied \(Rel\) and \(\delta_i\) (\(i = 1, 2, 3\)).

<table>
<thead>
<tr>
<th>Type</th>
<th>Method</th>
<th>Time(s)</th>
<th>VDW</th>
<th>Sintel</th>
<th>NYUDV2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>(\delta_1)</td>
<td>(Rel)</td>
<td>(OPW)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(\delta_2)</td>
<td>(Rel)</td>
<td>(OPW)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(\delta_3)</td>
<td>(Rel)</td>
<td>(OPW)</td>
</tr>
<tr>
<td>Image</td>
<td>Midas [28]</td>
<td>0.76</td>
<td>0.651</td>
<td>0.288</td>
<td>0.676</td>
</tr>
<tr>
<td></td>
<td>DPT [27]</td>
<td>0.97</td>
<td>0.730</td>
<td>0.215</td>
<td>0.470</td>
</tr>
<tr>
<td>Test-time Training</td>
<td>CVD [23]</td>
<td>352.58</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>Robust-CVD [13]</td>
<td>270.28</td>
<td>0.676</td>
<td>0.261</td>
<td>0.279</td>
</tr>
<tr>
<td></td>
<td>Zhang et al. [48]</td>
<td>464.83</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Learning Based</td>
<td>ST-CLSTM [46]</td>
<td>0.58</td>
<td>0.477</td>
<td>0.521</td>
<td>0.448</td>
</tr>
<tr>
<td></td>
<td>Cao et al. [5]</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>FMNet [40]</td>
<td>3.87</td>
<td>0.472</td>
<td>0.514</td>
<td>0.402</td>
</tr>
<tr>
<td></td>
<td>DeepV2D [34]</td>
<td>68.71</td>
<td>0.546</td>
<td>0.528</td>
<td>0.427</td>
</tr>
<tr>
<td></td>
<td>WSVD [36]</td>
<td>4.25</td>
<td>0.637</td>
<td>0.314</td>
<td>0.462</td>
</tr>
<tr>
<td></td>
<td>Ours(Midas)</td>
<td>1.55</td>
<td>0.700</td>
<td>0.240</td>
<td>0.180</td>
</tr>
<tr>
<td></td>
<td>Ours(DPT)</td>
<td>1.73</td>
<td>\textbf{0.742}</td>
<td>\textbf{0.208}</td>
<td>\textbf{0.147}</td>
</tr>
</tbody>
</table>

Table 2: **Comparisons with the state-of-the-art approaches.** We report the total time of processing eight 640 × 480 frames by different methods on one NVIDIA RTX A6000 GPU. Best performance is in boldface. Second best is underlined.

Table 3: **Comparisons of the learning-based approaches on NYUDV2 [31] dataset.** All the compared methods use NYUDV2 [31] as the training and evaluation data. Our neural video depth stabilizer trained from scratch also achieves better performance than all the other methods.
5.2. Comparisons with Other Video Depth Methods

Comparisons with the TTT-based methods. First focus on the test-time training (TTT) approaches [13, 23, 48]. As shown in Table 2, our learning-based framework outperforms TTT-based approaches by large margins in terms of inference speed, accuracy and consistency. Our NVDS shows at least 6.6% and 7.6% improvements for $\delta_1$ and $OPW$ than Robust-CVD [13] on VDW, Sintel [4], and NYUDV2 [31]. Our learning-based approach is over one hundred times faster than Robust-CVD [13]. Our strong performance demonstrates that learning-based frameworks are capable of attaining great performance with much higher efficiency than TTT-based methods [13, 23, 48].

It is also worth-noticing that the TTT-based approaches are not robust for natural scenes. CVD [23] and Zhang et al. [48] fail on some videos on VDW and Sintel [4] dataset due to erroneous pose estimation results. Hence, some of their results are not reported in Table 2. Refer to supplementary for more details. Although Robust-CVD [13] can produce results for all testing videos by jointly optimizing the camera poses and depth, it is still not robust for many videos and produces obvious artifacts as shown in Fig. 5.

Comparisons with the learning-based methods. The proposed neural video depth stabilizer also attains better accuracy and consistency than previous learning-based approaches [5, 36, 40, 46] on all the three datasets, including natural scenes and closed domain. As shown in Table 2, on our VDW and Sintel with natural scenes, the proposed NVDS shows obvious advantages: improving $\delta_1$ and $OPW$ by over 9% and 18.6% compared with previous learning-based methods. Note that, our NVDS can benefit from stronger single-image models and obtain better performance, which will be discussed in Table 5.

To better compare NVDS with previous learning-based method, we only use NYUDV2 [31] as training and evaluation data for comparisons. As shown in Table 3, the proposed NVDS improves the FMNet [40] by 9.9% and 3.8% in terms of $\delta_1$ and $OPW$. We also achieve better performance than DeepV2D [34], which is the previous state-of-the-art structure-from-motion-based methods but can only deal with completely static scenes. The results demonstrate that using our architecture alone can also obtain better video depth performance.

Qualitative Comparisons. We show some qualitative comparisons on natural-scene videos in Fig. 5. We draw the scanline slice over time. Fewer zigzagging pattern means better consistency. The initial estimation of DPT [27] in the sixth row contains flickers and blurs, which are eliminated with the proposed NVDS, as shown in the last row. Although the TTT-based Robust-CVD [13] shows competitive performances on the indoor NYUDV2 [31] dataset, it is not robust on the natural scenes. As can be observed in the fourth row, Robust-CVD produces obvious artifacts due to erroneous pose estimation.
5.3. Model Efficiency Comparisons

To evaluate the efficiency, we compare the inference time on a 640 × 480 video with eight frames. The inference is conducted on one NVIDIA RTX A6000 GPU. As shown in Table 2, the proposed NVDS reduces the inference time by hundreds of times compared to the TTT-based approaches CVD [23], Robust-CVD [13], and Zhang et al. [48]. The learning-based method DeepV2D [34] alternately estimates depth and camera poses, which is time-consuming. WSVD [36] is also slow because they need to compute optical flow [11] between consecutive frames while inference.

We also evaluate the efficiency of the proposed stabilization network and different depth predictors. Model parameters and FLOPs are reported in Table 7. The FLOPs are evaluated on a 384 × 384 video with four frames. Our stabilization network only introduces limited computation overhead compared with the depth predictors [27, 28, 45].

5.4. Ablation Studies

Here we verify the effectiveness of the proposed method. We first ablate the plug-and-play manner with different single-image depth models. Besides, We also discuss the bidirectional inference, the temporal loss, the reference frames, and baselines without the stabilization network.

**Plug-and-play Manner.** As shown in Table 5, we directly adapt our NVDS to three different state-of-the-art single-image depth models DPT [27], Midas [28], and NeWCRFs [45]. For NeWCRFs [45], we adopt their official checkpoint on NYUDV2 [31]. By post-processing their initial flickering disparity maps, our NVDS achieves better temporal consistency and spatial accuracy. With higher initial depth accuracy, the spatial performance of our NVDS is also improved. The experiment demonstrates the effectiveness of our plug-and-play manner. Visual comparisons with those three depth predictors are shown in Fig. 6. Depth maps and scanline slice prove our accuracy and consistency.
Bidirectional Inference. As shown in Table 6, whether using DPT [27] or Midas [28] as the single-image depth predictor, our NVDS can already enforce the temporal consistency with previous or post sliding window of target frame. The bidirectional inference can further improve the consistency with larger bidirectional temporal receptive fields.

**Temporal Loss.** As in Table 8 (a), without the temporal loss as explicit supervision, our stabilization network can enforce temporal consistency. Adding the temporal loss can further remove flickers and improve temporal consistency.

**Reference Frame Intervals.** We denote the inter-frame intervals as $l$. As shown in Table 8 (b), $l = 1$ attains the best performance in our experiments.

**Reference Frame Numbers.** As shown in Table 9 (b), using three reference frames ($n=3$) for a target frame achieves the best results. More reference frames ($n=4$) increase computational costs but bring no improvement, which can be caused by the temporal redundancy of videos.

**Baselines without Stabilization Network.** With DPT as the depth predictor, we train and evaluate two baselines without the stabilization network on the same subset as Table 8. We use the same temporal window as NVDS. The first baseline (Single-frame) can only process each frame independently. Temporal window and loss $L_t$ are used for consistency. The second baseline (Multi-frame) uses neighboring frames concatenated by channels to predict depth of the target frame. Training and inference strategies are kept the same as NVDS. As shown in the Table 9 (a), temporal flickers cannot be solved by simply adding temporal windows and training loss on those baselines. Proper designs are needed for inter-frame correlations. Our stabilization network improves consistency ($OPW$) significantly.

6. Conclusion

In this paper, we propose a Neural Video Depth Stabilizer framework and a large-scale natural-scene VDW dataset for video depth estimation. Different from previous learning-based video depth models that function as standalone models, our Neural Video Depth Stabilizer learns to stabilize the flickering results from the estimations of single-image depth models. In this way, Neural Video Depth Stabilizer can focus on the learning of temporal consistency, while inheriting the depth accuracy from the cutting-edge depth predictors without further tuning. We also elaborate on the VDW dataset to alleviate the data shortage. To our best knowledge, it is currently the largest video depth dataset in the wild. We hope our work can serve as a solid baseline and provide a data foundation for the learning-based video depth models.

**Limitations and future work.** Currently, we only offer one implementation for the NVDS framework. In future work, we will consider using more mechanisms in the stabilization network and adding more implementations for different applications, e.g., the lightweight models.

**Acknowledgments.** This work was funded by Adobe Research. Meanwhile, our work is also supported under the RIE2020 Industry Alignment Fund - Industry Collaboration Projects (IAF-ICP) Funding Initiative, as well as cash and in-kind contribution from the industry partner(s).
References


(CVPR), pages 16283–16292, 2022. 1


[34] Zachary Teed and Jia Deng. Deepv2d: Video to depth with differentiable structure from motion. In International Conference on Learning Representations, 2019. 2, 6, 7, 8


[37] Qiang Wang, Shizhen Zheng, Qingsong Yan, Fei Deng, Kaiyong Zhao, and Xiaowen Chu. Irs: A large naturalistic indoor robotics stereo dataset to train deep models for disparity and surface normal estimation. In IEEE International Conference on Multimedia and Expo (ICME), pages 1–6. IEEE Computer Society, 2021. 2, 5, 6, 8


[47] Xuaner Zhang, Kevin Matzen, Vivien Nguyen, Dillon Yao, You Zhang, and Ren Ng. Synthetic defocus and look-ahead autofocus for casual videography. ACM Transactions on Graphics (TOG), 38(4), 2019. 1
