Less is More: Consistent Video Depth Estimation with Masked Frames Modeling

Yiran Wang
wangyiran@hust.edu.cn
School of AIA, Huazhong University of Science and Technology

Zhiyu Pan
zhiyupan@hust.edu.cn
School of AIA, Huazhong University of Science and Technology

Xingyi Li
xingyi_li@hust.edu.cn
School of AIA, Huazhong University of Science and Technology

Zhiguo Cao
zgcao@hust.edu.cn
School of AIA, Huazhong University of Science and Technology

Ke Xian*
kexian@hust.edu.cn
School of AIA, Huazhong University of Science and Technology

Jianming Zhang
jianmzha@adobe.com
Adobe Research

ABSTRACT

Temporal consistency is the key challenge of video depth estimation. Previous works are based on additional optical flow or camera poses, which is time-consuming. By contrast, we derive consistency with less information. Since videos inherently exist with heavy temporal redundancy, a missing frame could be recovered from neighboring ones. Inspired by this, we propose the frame masking network (FMNet), a spatial-temporal transformer network predicting the depth of masked frames based on their neighboring frames. By reconstructing masked temporal features, the FMNet can learn intrinsic inter-frame correlations, which leads to consistency. Compared with prior arts, experimental results demonstrate that our approach achieves comparable spatial accuracy and higher temporal consistency without any additional information. Our work provides a new perspective on consistent video depth estimation.

CCS CONCEPTS

• **Computing methodologies → Computer vision tasks**.

KEYWORDS

depth estimation, temporal consistency, masked frames modeling

ACM Reference Format:
Yiran Wang, Zhiyu Pan, Xingyi Li, Zhiguo Cao, Ke Xian, and Jianming Zhang. 2022. Less is More: Consistent Video Depth Estimation with Masked Frames Modeling. In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisboa, Portugal.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3503161.3547978

1 INTRODUCTION

Monocular video depth estimation plays a vital role in many applications, such as 2D-to-3D video conversion [23], scene reconstruction [48], and bokeh rendering [38, 56]. To obtain pleasant and immersive experience, both spatial accuracy and temporal consistency are required. In recent years, the spatial accuracy of single image depth estimation [28, 41, 42, 52] has been significantly improved. However, the temporal consistency remains an open question. When deploying single image depth estimation approaches on videos, it would cause an obvious flickering problem due to the independent computation of each frame. To estimate consistent video depth, as shown in Fig. 1(a), most previous approaches [5, 24, 35, 55, 57] model inter-frame correlations based on extra temporal clues. Such clues can be provided by additional computations, *e.g.*, optical flow [5, 22, 49], pose estimation [24, 35, 43], and generative adversarial networks (GANs) [17, 55]. As a consequence, those methods totally fail when the temporal clues are inaccurate. Meanwhile, these approaches are time-consuming. For example, for a video of 244 frames, CVD [35] takes about 40 minutes for test-time training on 4 NVIDIA Tesla M40 GPUs. This begs the question – *is it possible to achieve temporal consistency without explicitly modeling inter-frame correlations?*

Languages, images, and videos are of high redundancy in nature. For example, we can still guess what a sentence means if some

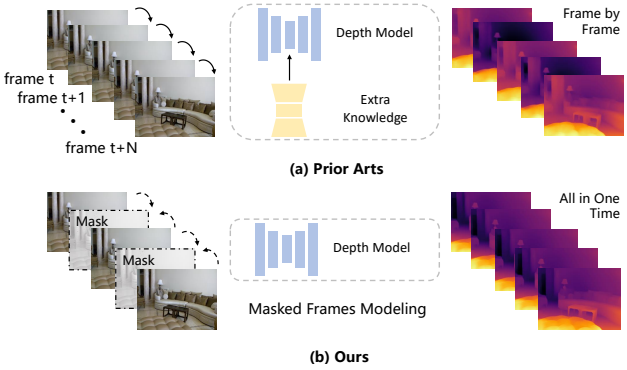

Figure 1: Prior arts leverage extra knowledge from optical flow [5, 24, 35] or GANs [55]. Most of them [5, 24, 55] transfer temporal knowledge serially and produce depth results frame by frame. By contrast, we derive the consistency with less information by masked frames modeling. Our method can produce depth results of all input frames in one time.

*Corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MM '22, October 10–14, 2022, Lisboa, Portugal.
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9203-7/22/10…$15.00
https://doi.org/10.1145/3503161.3547978

words are blocked; we can still recognize what a picture depicts if some patches are missing; and we can still imagine how an object moves if some video frames are removed. Inspired by this, we explore temporal consistency with implicit constraints by reconstructing the masked frames, which reduces the dependency on additional optical flow or camera poses. In particular, we design a subtask to recover depth maps of the masked frames based on their neighboring ones. As illustrated in Fig. 1(b), different from previous works, we introduce the masking and predicting paradigm to video depth estimation. Randomly masking some input frames, our model is trained to reconstruct the depth structures of unmasked frames. This can force our model to predict the depth results based on possibly relevant frames. In this manner, the inter-frame correlations are enhanced and the model acquires a larger temporal receptive field. Therefore, the video depth results can be more consistent.

We propose the frame masking network (FMNet) for consistent video depth estimation. The input video sequences are processed by a spatial feature extractor that generates the representation of each frame. To model the global inter-frame correlations, we adopt a transformer [50] architecture as the temporal feature extractor. However, we can only represent frames as feature maps to keep their spatial structures, which makes the cost of attention computation unacceptable. Inspired by ConvTransformer [34], we choose to approximate the attention computation of feature maps by convolution [26]. The temporal encoder randomly masks a certain portion of input frames and encodes the temporal correlation features based on the remaining frames. We use a learnable mask token to fill in the masked positions. The completed sequence is then processed by the temporal decoder and depth predictor to estimate depth structures of both masked and unmasked frames. In this manner, the temporal encoder can learn temporal inter-frame correlations implicitly. The temporal correlations knowledge enables the FMNet to generate consistent depth results.

Experimental results demonstrate that our FMNet can generate depth results with both spatial accuracy and temporal consistency. Compared with other state-of-the-art approaches, quantitative and qualitative results show the superiority of our FMNet in temporal consistency. We also achieve comparable spatial accuracy with prior arts. Further analysis on the NYU Depth V2 dataset shows that a very high masking ratio benefits both accuracy and consistency. Different from previous works relying on optical flow, pose estimation, or GANs, our work provides a novel information-reductive perspective for temporal consistency in video depth estimation. The main contributions of this paper can be summarized as follows:

- We propose a masked video transformer for consistent video depth estimation without relying on optical flow, pose estimation, and GANs.
- To the best of our knowledge, we are the first to introduce the ConvTransformer for video depth estimation, which can encode inter-frame correlations in parallel.

2 RELATED WORK

2.1 Consistent Video Depth Estimation

Consistent video depth estimation focuses on removing flickering in video depth results. It takes both spatial accuracy and temporal consistency into account. Unfortunately, approaches [3, 27, 41, 42, 51, 52] for single image depth estimation fail to handle the temporal consistency. They can not model the inter-frame correlations, causing obvious flickering in video depth results. Various approaches are devoted to tackling this problem based on two paradigms: test-time optimization and training-time modeling.

Test-time optimization approaches refine video depth results from existing models. CVD [35] refines depth models [16, 42] by warping frame pairs using camera poses [43, 44] and optical flow [22]. Their main limitation is that accurate camera poses are required. Unfortunately, pose estimation itself is a challenging problem and would fail in the presence of dynamic object motion. To address this, Robust-CVD [24] jointly optimizes camera poses and depth alignments. Although their method improves the robustness of video depth estimation, such joint optimization is inefficient and time-consuming, as we explained earlier.

Training-time modeling approaches generate consistent video depth without the time-consuming test-time training process. ST-CLSTM [55] applies LSTM and GAN for the temporal consistency. Cao *et al.* [5] propose a convolutional spatial-temporal propagation network [9] trained with knowledge distillation [20, 31, 32] and optical flow [22]. These methods suffer from heavy training burdens with additional optical flow or GANs for supervision. By contrast, our work utilizes the masked frames predicting strategy and does not require any additional information.

2.2 Transformer

RNNs and LSTMs [21, 45] have dominated the sequence processing area for a long time until the emergence of transformer [50]. The transformer can model correlations among sequence elements in parallel, which can encode more global representations. Some works [7, 11, 13, 33, 59] have shown the superiority of transformer in image-based tasks. For video data, ViViT [1] designs their model by factorizing the spatial and temporal dimension as a straightforward extension of ViT [11]. Liu *et al.* [34] replace the traditional attention mechanism with convolutional self-attention and introduce the ConvTransformer architecture. Their model can directly process video sequences, which is more appropriate for video-related tasks. In this work, we follow the idea of ConvTransformer [34] to model inter-frame correlations.

2.3 Masked Data Modeling

Recent works explore to learn structure information by reconstructing the masked signals. For language data, GPT [4, 39, 40] and BERT [10] propose to predict the masked words in sentences, which shows superiority in natural language processing. For image data, iGPT [8] and BEiT [2] convert the image patches to visual tokens and predict the masked ones. MAE [18] directly masks the image patches and learns image representations by reconstructing the original image. To the best of our knowledge, we are the first to adopt the idea of masked data modeling to deal with the task of video depth estimation. We find that this strategy can also benefit the learning of consistent representations for videos.

2.4 Structure from Motion

Structure-from-motion (SFM) methods [47, 48] predict depth maps by feature matching over multiple frames. BA-Net [47] regresses

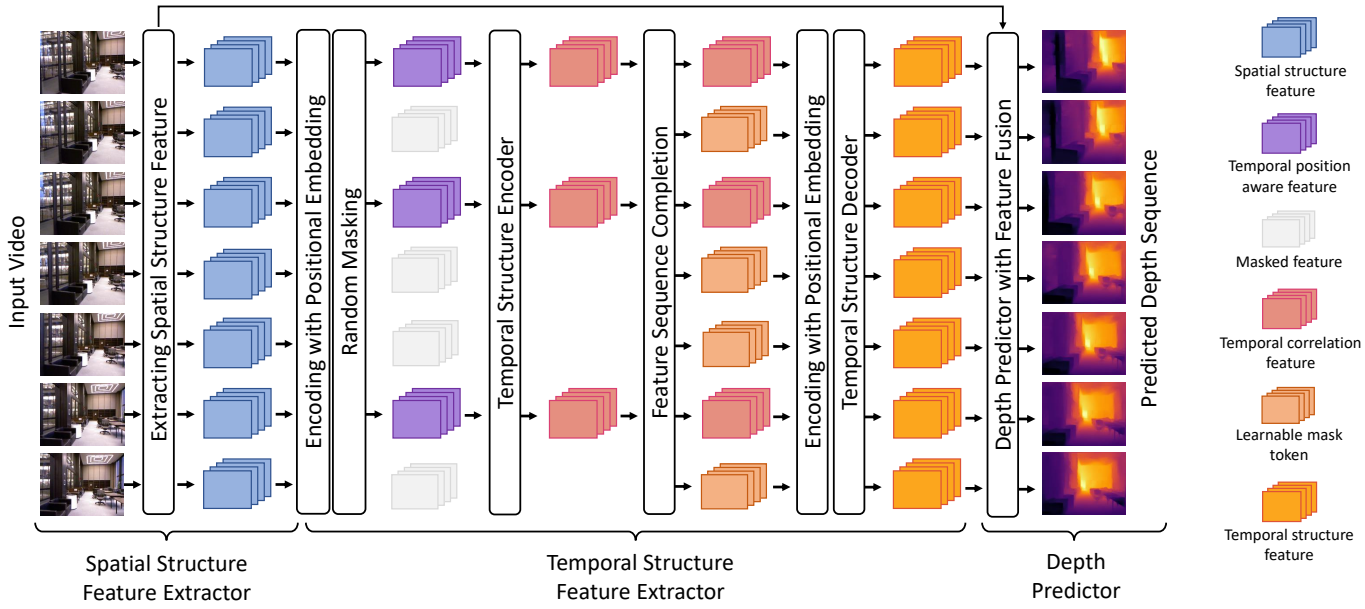

Figure 2: Illustration of our FMNet for consistent video depth estimation. Taking N consecutive frames as input, the spatial feature extractor generates the feature of each frame. The temporal structure encoder randomly masks a portion of input frames and encodes temporal correlation features. We use a learnable mask token to fill in the masked positions. The completed sequence is processed by the temporal structure decoder to restore the masked temporal structure features. Finally, the depth predictor fuses the spatial and temporal structure features to predict consistent depth maps of all input frames in one time.

depth maps via feature-metric bundle adjustment. DeepV2D [48] alternately updates depth and camera motion. They conduct cost volumes and feature matching in their depth module. Those SFM-based methods highly rely on accurate camera poses. They only benefit static scenes but do not account for dynamical objects due to the failure of pose estimation. By contrast, our FMNet is not limited by pose estimation. Our method is also significantly faster than DeepV2D [48] because the feature matching and pose estimation are inefficient and time-consuming.

3 PROPOSED METHOD

3.1 Overview

Here we present an overview of our method. Fig. 2 shows the technical pipeline of our FMNet framework. Given N consecutive frames as input, the spatial structure feature extractor, *i.e.*, a CNN encoder, extracts spatial features of input frames independently. It cannot model input-frame correlations. Although RNNs [45] and LSTMs [21] can somehow deal with the temporal correlations, we observe that ST-CLSTM [55] generates obvious flickering (see Sec. 4.4) between adjacent frames due to its seriality and locality. Hence, we naturally resort to transformer [50] for its parallelism and globality. Nevertheless, the original transformer is not appropriate for processing sequences of high-dimensional features due to the computational overhead of attention mechanism. Following [34], we adopt the idea of ConvTransformer, which can directly process features sequences without partitioning patches and flatten operation. To represent the chronological order of input frames, we add positional embeddings to each sequence of spatial features.

Recent MAE [18] observes and utilizes the spatial redundancy of single images. By masking and reconstructing image patches, the MAE encoder [11] is forced to learn high-level semantic information. Inspired by this, our FMNet leverages the high redundancy of videos in temporal dimension. We design our masked frames predicting strategy to force the temporal structure feature extractor to learn temporal correlations among frames. To be specific, as illustrated in Fig. 2, we randomly mask a portion of the spatial feature maps with positional embeddings. The remaining unmasked features are fed into our temporal structure encoder to build inter-frames temporal correlations. The outputs of the temporal structure encoder are feature maps corresponding to the unmasked frames with the same shape as its input. The next step is to restore temporal structure features of the masked frames based on the remaining ones. We use a shared and learnable mask token to represent the masked positions and complete the full sequence of feature maps. We add positional embeddings to the full sequence again and feed it into our temporal structure decoder. Our temporal structure decoder has the same ConvTransformer architecture as the encoder, which has only one transformer layer. The heavy computational burden of the full features sequence is only sustained by the lightweight one-layer decoder, while the six-layer encoder only takes the unmasked features as input. In this way, the computational cost of our method could be reduced compared with other methods based on optical

flow or camera poses. The temporal structure decoder predicts the sequence of temporal structure features for all input frames.

Finally, we adopt our depth predictor, *i.e.*, a CNN decoder, to restore depth maps of the input sequence. We employ five up-projection modules to improve the spatial resolution and decrease the channel numbers. We use the feature fusion module (FFM) [29, 30] and skip connection from the spatial structure feature extractor to the depth predictor to fuse the spatial and temporal structure features. As for training, we adopt the widely-used scale-invariant loss [12]. Our method does not need extra temporal loss functions based on camera poses, optical flow, or GANs. With the same training paradigm as single image depth estimation, our FMNet can predict video depth results with both spatial accuracy and temporal consistency because of our masked frames predicting strategy.

We will illustrate our ConvTransformer architecture in Sec. 3.2 and our masked frames predicting strategy in Sec. 3.3. The loss function for training is in Sec. 3.4.

3.2 ConvTransformer

Given a sequence of N input video frames $F = \{F_0, F_1, \cdots, F_{N-1}\}$ where $F_i \in \mathbb{R}^{H\times W\times 3}$, our ConvTransformer is built to model temporal correlations among input frames. However, taking the original video frames as direct input is inappropriate due to the high resolution. We consider our spatial structure feature extractor as feature embeddings to extract the spatial feature maps of input frames $f = \{f_0, f_1, \cdots, f_{N-1}\}$, where $f_i \in \mathbb{R}^{h\times w\times c}$, h, w, and c denote feature maps height, width, and the number of channels, respectively. If we denote the spatial structure feature extractor by $\mathcal{F}_{\theta_\mathcal{F}}$, where $\theta_\mathcal{F}$ refers to the parameters of $\mathcal{F}_{\theta_\mathcal{F}}$, we extract the spatial feature maps of input frames using the shared extractor with same parameters:

$$f_i = \mathcal{F}_{\theta_\mathcal{F}}(F_i), \quad i \in [0, N-1]\,. \tag{1}$$

We use our spatial structure feature extractor to extract the spatial features and our ConvTransformer to represent the temporal correlation features among input frames. In this section, we do not consider our masked frames predicting strategy for the time being. In order to illustrate the ConvTransformer architecture, we assume the sequence of feature maps $f = \{f_0, f_1, \cdots, f_{N-1}\}$ as its input.

Similar to the original transformer [50], the ConvTransformer needs positional embeddings to represent the chronological order of input feature maps. We adopt the same positional embeddings as [34], which are 3D tensors with the same shape as input feature maps. It can be expressed as an extension of the original transformer [50] sine and cosine positional embeddings:

$$\begin{aligned} PE(pos, (x, y), 2k) &= \sin(pos/10000^{2k/c})\,, \\ PE(pos, (x, y), 2k+1) &= \cos(pos/10000^{2k/c})\,, \end{aligned} \tag{2}$$

where pos denotes the position of a certain feature map in the full sequence, (x, y) denotes spatial locations in feature maps, $2k$ refers to the channel dimension, and c denotes the feature maps channels amount. Given a feature map $f_i, i \in [0, N-1]$, its positional embedding PE_i has the same shape $h \times w \times c$ as f_i. The positional embeddings can be directly added to the original feature maps:

$$p_i = f_i + PE_i, \quad i \in [0, N-1]\,. \tag{3}$$

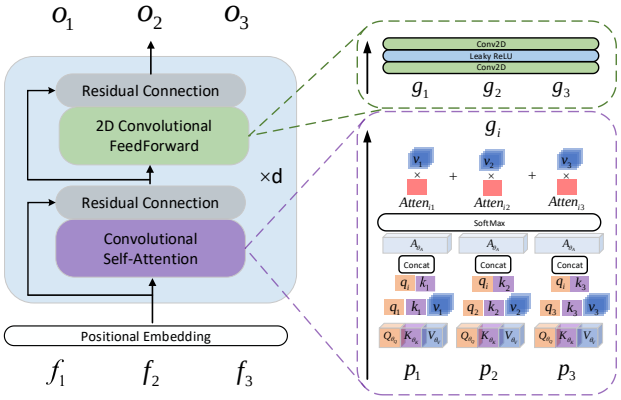

Figure 3: The Architecture of ConvTransformer.Queries, keys, and values are generated by 3 convolution sub-networks: $\mathcal{Q}_{\theta_\mathcal{Q}}$, $\mathcal{K}_{\theta_\mathcal{K}}$, and $\mathcal{V}_{\theta_\mathcal{V}}$. The attention of q_i and k_j is approximated by convolutional sub-network $\mathcal{A}_{\theta_\mathcal{A}}$.

The sequence of feature maps with positional embeddings $p = \{p_0, p_1, \cdots, p_{N-1}\}$ where $p_i \in \mathbb{R}^{h\times w\times c}$ can be fed into the ConvTransformer. The ConvTransformer is a stack of several identical ConvTransformer layers. The architectures of ConvTransformer and its layers are shown in Fig. 3. Each layer consists of 2 sub-layers: the convolutional self-attention layer and the feed-forward layer. When getting the sequence of input feature maps, three different convolution sub-networks $\mathcal{Q}_{\theta_\mathcal{Q}}$, $\mathcal{K}_{\theta_\mathcal{K}}$, and $\mathcal{V}_{\theta_\mathcal{V}}$ are used to generate the query sequence $q = \{q_0, q_1, \cdots, q_{N-1}\}, q_i \in \mathbb{R}^{h\times w\times 1}$, the key sequence $k = \{k_0, k_1, \cdots, k_{N-1}\}, k_i \in \mathbb{R}^{h\times w\times 1}$, and the value sequence $v = \{v_0, v_1, \cdots, v_{N-1}\}, v_i \in \mathbb{R}^{h\times w\times c}$:

$$\begin{aligned} q_i = \mathcal{Q}_{\theta_\mathcal{Q}}(p_i)\,, k_i = \mathcal{K}_{\theta_\mathcal{K}}(p_i)\,, v_i = \mathcal{V}_{\theta_\mathcal{V}}(p_i)\,, \\ i \in [0, N-1]\,. \end{aligned} \tag{4}$$

In this way, the attention map of two arbitrary frames i and j can be obtained by another convolution sub-network $\mathcal{A}_{\theta_\mathcal{A}}$:

$$Atten(i, j) = \mathcal{A}_{\theta_\mathcal{A}}(concat[q_i, k_j])\,, \quad i, j \in [0, N-1]\,, \tag{5}$$

where $concat$ refers to a concatenation operation of q_i and k_j in the channel dimension, $Atten(i, j) \in \mathbb{R}^{h\times w\times 1}$.

For frame i, when getting attention maps of all input frames $j \in [0, N-1]$, a $SoftMax$ operation is applied to the attention maps. The output g_i corresponding to feature map f_i can be calculated as a weighted sum of all values:

$$Atten(i, j) = SoftMax(Atten(i, j)), \quad j \in [0, N-1]\,, \tag{6}$$

$$g_i = \sum_{j=0}^{N-1} Atten(i, j) v_j, \quad g_i \in \mathbb{R}^{h\times w\times c}\,. \tag{7}$$

With the ConvTransformer architecture, the next step is to force the model to learn temporal correlations among input frames. We design our masked frames predicting strategy to build the inter-frame temporal correlations. The masked frames predicting strategy will be elaborated in the next section.

3.3 Masked frames predicting

With the ConvTransformer described in Sec. 3.2, the next problem is how to enforce the model to build inter-frame temporal consistency. In this section, we will elaborate on our masked frames predicting strategy for learning video consistency.

Given a sequence of N input video frames $F = \{F_0, F_1, \cdots, F_{N-1}\}$, we can get the corresponding sequence of feature maps with positional embeddings $p = \{p_0, p_1, \cdots, p_{N-1}\}$ as shown in Eq. (1) and Eq. (3). If we directly fed all the feature maps into ConvTransformer, the input would be highly redundant and the computational burden could be heavy. The high redundancy is not beneficial for learning temporal consistency. Previous methods [2, 18] reduce the redundancy of single images by masking a certain portion of patches. They force their model to predict the masked tokens or patches, and learn high-level semantic information rather than low-level details.

Inspired by the similarity between images spatial redundancy and videos temporal redundancy, we design our masked frames predicting strategy. For the spatial feature maps of N input frames, we randomly mask a certain portion of them. In our case, considering the higher redundancy of videos, we adopt a very high masking ratio, which benefits the final results for both accuracy and consistency in our experiments. Retaining only a minority of the input frames ensures that we can represent inter-frame temporal correlations with lower redundancy. We illustrate our masking sampling strategies and masking ratios in Sec. 4.3. We also conduct ablation study on masking ratios in Sec. 4.6.

The unmasked feature maps p_{um} will be fed into the temporal structure encoder to build temporal correlations. The encoder is a six-layer ConvTransformer described in Sec. 3.2. With the masking strategy, the video redundancy and computational cost are reduced. The encoder learns temporal correlations between the unmasked frames. If we denote the temporal structure encoder by E_{θ_E}, the output temporal correlation features of unmasked frames t_{um}, where $t_{um} \subset \mathbb{R}^{h \times w \times c}$ can be expressed as:

$$t_{um} = E_{\theta_E}(p_{um}) \,. \tag{8}$$

The next step is to complete the masked features according to the unmasked temporal correlation features t_{um}. We set a learnable mask token $m \in \mathbb{R}^{h \times w \times c}$ to represent the masked feature maps. The mask token is shared across all masked frames. We use the mask token to complete the corresponding positions of masked frames. The unmasked temporal correlation features t_{um} remain on original positions. Positional embeddings as Eq. (2) and Eq. (3) are added to the full sequence again because the original mask token does not have temporal information. We denote the full sequence with m, t_{um}, and positional embeddings by t_f. We use our temporal structure decoder D_{θ_D}, a one-layer ConvTransformer, to reconstruct temporal features of masked positions based on the unmasked temporal correlation features:

$$t_r = D_{\theta_D}(t_f) \,, \tag{9}$$

where $t_r = \{t_{r_1}, t_{r_2}, \cdots, t_{r_{N-1}}\}$ denotes the temporal structure features sequence of all input frames. In this way, the features of masked frames are completed according to the unmasked temporal correlation features. The video redundancy is reduced by masking and the temporal consistency is built by mask reconstruction. Experimental results show that our masked frames predicting strategy can efficiently improve the consistency of video depth results.

Finally, the last step is to recover depth maps from the temporal structure features sequence t_r. Our depth predictor is a CNN decoder with five up-projection modules to improve the spatial resolution and decrease the channel numbers. Skip connections and feature fusion model [29, 30] are used to fuse the temporal structure features and the spatial structure features. Combining spatial and temporal information, the depth predictor can finally predict video depth results with both spatial accuracy and temporal consistency. With our masked frames predicting strategy illustrated, we will elaborate on the loss function in the next section.

3.4 Training Loss

We adopt the widely-used scale-invariant loss [12] to measure the discrepancy between output depth and ground truth. Given the predicted depth map for one frame $d \in \mathbb{R}^{H \times W}$ and the ground truth $d^* \in \mathbb{R}^{H \times W}$, the loss can be expressed as:

$$L(u) = \alpha \sqrt{\frac{1}{n} \sum_i u_i^2 - \frac{\lambda}{n^2} (\sum_i u_i)^2} \,,$$
$$u_i = \log d_i - \log d_i^* \,,$$

where n denotes the number of pixels with valid ground truth value and i denotes pixels index. Similar to many other works, we use $\lambda = 0.85$ and $\alpha = 10$ in all our experiments.

Previous works utilize optical flow, camera poses, or GANs, which could fail when those extra information is inaccurate. Our method does not rely on additional information. Based on our masked frames predicting strategy, our FMNet can produce consistent video depth results only by the same training paradigm as single image depth estimation.

4 EXPERIMENTS

In this section, we evaluate our FMNet on the indoor NYU Depth V2 [46] dataset and the outdoor KITTI [15] dataset. Firstly, we briefly introduce the datasets. In Sec.4.2, we describe the evaluation metrics for spatial accuracy and temporal consistency. Some implementation details are shown in Sec. 4.3. We further illustrate our motivation by experiments in Sec. 4.4. The comparison results with state-of-the-art methods are shown in Sec. 4.5. We conduct ablation studies to prove the effectiveness of our methods in Sec. 4.6. We also compare the inference speed of different methods in Sec. 4.7.

4.1 Datasets

NYU Depth V2 contains 464 videos taken from indoor scenes. We apply the same train/test split as Eigen *et al.* [12] with 249 videos for training and 654 samples from the rest 215 videos for testing. We use the resolution of 640×480 for training.

KITTI contains 61 outdoor video scenes captured by cameras and depth sensors mounted on a driving car. We apply the same train/test split as Eigen *et al.* [12] with 32 videos for training and 697 samples from the rest 29 videos for testing. We use the resolution of 1216×352 for training.

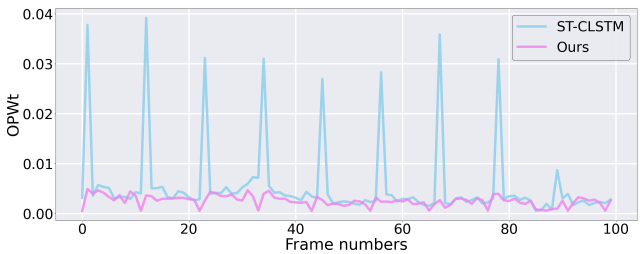

Figure 4: Motivation review. The X-axis represents the frame numbers and Y-axis means our temporal consistency metric $\mathbf{OPW_t}$ for each consecutive frame pair.

4.2 Evaluation Metrics

We evaluate the performance of our FMNet using depth estimation metrics and the optical flow consistency metric. We adopt the commonly applied depth metrics *Rel*, *RMSE*, log 10, and $\delta_i (i = 1, 2, 3)$.

As for the temporal consistency metric, we adopt the optical flow based warping loss (*OPW*) proposed by [5]. They adopt it as a loss function for temporal consistency supervision. In our case, we do not use it for training. Instead, we use *OPW* to quantitatively evaluate the temporal consistency. The *OPW* is defined as follows:

$$\begin{aligned} OPW_t &= \frac{1}{n}\sum_{i=1}^{n} M^{(i)}_{t+1\Rightarrow t} ||d^{(i)}_{t+1} - \hat{d}^{(i)}_t||_1 \,, \\ OPW &= \sum_{t=0}^{T-1} OPW_t \,, \end{aligned} \tag{10}$$

where $\hat{d}_t$ is the predicted depth d_t warped by the backward optical flow $FL_{t+1\Rightarrow t}$ between input frames F_t and F_{t+1}. In our implementation, we adopt the trained RAFT [49] model to compute optical flow between frames. For a video with T frames, we calculate the $OPWt$ between each consecutive frame pair and add them together as the consistency index for the certain video. $M^{(i)}_{t+1\Rightarrow t}$ is the visibility mask calculated from the warping discrepancy between frame F_{t+1} and the warped frame $\hat{F}_t$:

$$M^{(i)}_{t+1\Rightarrow t} = \exp(-\beta ||F_{t+1} - \hat{F}_t||^2_2) \,. \tag{11}$$

Identical to [5], we set $\beta = 50$ and use bilinear sampling layer for frames warping.

As for the evaluation on a test dataset with many testing videos, we calculate the *OPW* as Eq. (10) for each video and add them together as the final *OPW* index on the test dataset.

4.3 Implementation Details

We train our FMNet for 20 epochs on the NYU Depth V2 dataset. We use 0.0001 as the initial learning rate, which decreases by a factor of 0.1 after every 5 epochs. We cut videos to sequences with $N = 12$ consecutive frames as input. As for the KITTI dataset, we train our proposed FMNet for 30 epochs. The initial learning rate is also 0.0001 and decreases by a factor of 0.1 after 10 epochs. We adopt the sequence length $N = 8$. Without loss of generality, our spatial structure feature extractor is based on the ResNeXt-101 [53].

As for the masking strategy in training time, we input N frames each time and randomly mask $N - 2$ frames. The masking ratio is 83.33% with $N = 12$ on the NYU Depth V2 dataset and 75% with $N = 8$ on the KITTI dataset. The random masking strategy can be assumed as a form of data argumentation so that our model can learn inter-frame correlations with various time intervals. Therefore, we do not apply any other data augmentation methods. Considering the higher redundancy of videos than that of single images, we only retain 2 frames for each input sequence. The redundancy will be minimized and the inter-frame temporal correlations still remain.

However, random masking strategy for inference could cause randomness in depth results. Instead, we use uniform masking for inference. For example, we will retain the fourth and eighth frames with $N = 12$. We further ablate our masking ratios in Sec. 4.6.

To prove the effectiveness of our approach, we also implement a baseline model without our temporal structure feature extractor. The baseline only consists of the spatial structure feature extractor and the depth predictor. It can be assumed as a CNN model for single image depth estimation without transformer and masking.

4.4 Motivation Review

In this section, we conduct an experiment to further expound on the motivation of our approach. In practice, a certain video clip will be partitioned into several video sequences with N frames due to the limitation of computational resources. In our case, we set $N = 12$ in Fig. 4. The traditional temporal models such as the ST-CLSTM [55] process a certain input video sequence frame by frame. The inter-frame temporal correlations are built relying on the memory cell in a serial manner. We adopt both the ST-CLSTM [55] and our FMNet to process the same input video clip which is divided into several N frames sequences. We compare the temporal consistency metric OPW_t for each consecutive frame pair.

At the junction of two sequences, we can observe that the ST-CLSTM [55] causes obvious flickering and inconsistency due to the disabled memory cell with initialization value, which is a common phenomenon for all input video clips. Passing the memory cell value of the previous sequence to the next one is also unreasonable and could cause error accumulation. The temporal consistency is more attached to adjacent frames. Passing the memory cell value too far has no benefit for consistency because the video scene might change completely. In this way, ST-CLSTM [55] inevitably causes the obvious inconsistency between adjacent sequences due to its seriality and locality.

In order to solve this problem, we design our FMNet based on ConvTransformer and masked frames predicting. The ConvTransformer can process input frames in a parallel manner. The masked frames predicting strategy is inspired by the high temporal redundancy of videos and the image patch masking strategy in recent MAE [18]. Compared with previous methods, our model has better characteristics of parallelism and globality. The model is forced to predict the depth results based on all possibly relevant input frames. In this way, our FMNet is equipped with a larger temporal receptive field and achieves better temporal consistency whether inside or between input sequences without relying on additional optical flow or camera poses.

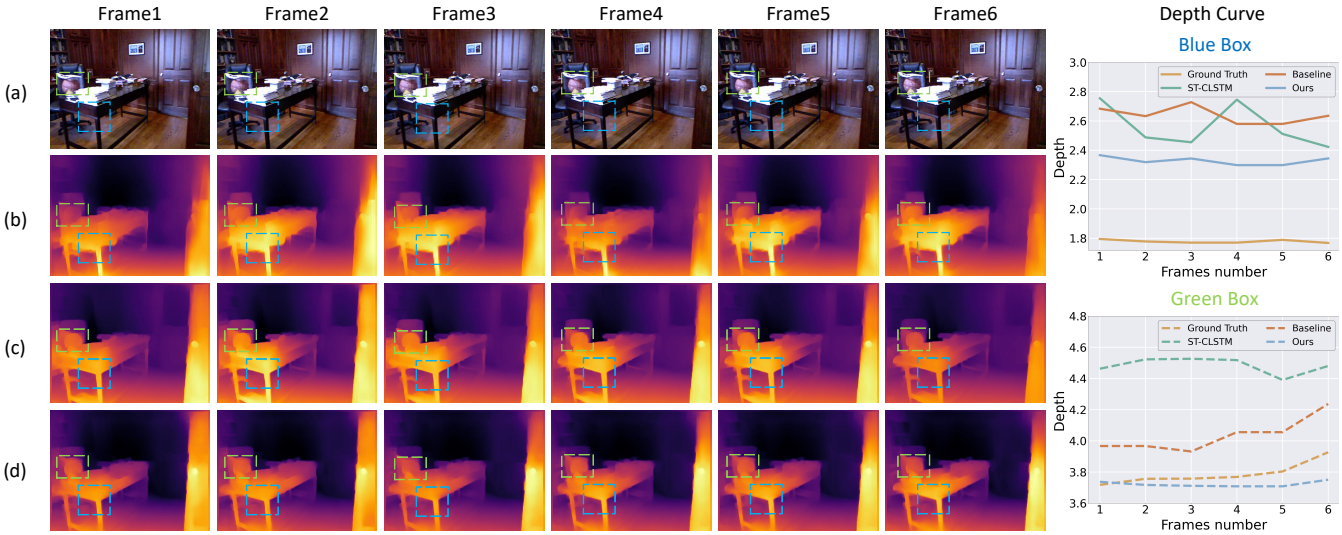

Figure 5: Qualitative depth results on the NYU Depth V2 dataset. The four rows are: (a) RGB inputs; (b) Results of ST-CLSTM [55]; (c) Baseline results; (d) Results of our FMNet. We highlight obviously different regions in dashed rectangular. For better comparison, we draw depth curves on the last column. Each curve represents depth value for the center point of a certain box in the six frames. Our FMNet shows higher depth accuracy and better temporal consistency than ST-CLSTM [55] and our baseline.

Table 1: Comparisons with state-of-the-art methods on the NYU Depth V2 dataset. δ_i means $\delta < 1.25^i$. We show our results in the last row. Best performance is in boldface.

Method	Rel	RMSE	log 10	δ_1	δ_2	δ_3
Laina *et al.* [25]	0.127	0.573	0.055	0.811	0.953	0.988
Pad-net [54]	0.120	0.582	0.055	0.817	0.954	0.987
Cao *et al.* [6]	0.141	0.540	0.060	0.819	0.965	0.992
DORN [14]	**0.115**	0.509	**0.051**	0.828	0.965	0.992
ST-CLSTM [55]	0.131	0.571	0.056	0.833	0.965	0.991
Cao *et al.* [5]	0.131	0.574	0.056	**0.835**	0.965	0.990
Ours	0.134	**0.452**	0.056	0.832	**0.968**	**0.992**

Table 2: Comparisons with state-of-the-art methods on the KITTI dataset. The last row shows our results. Best performance is in boldface.

Method	Rel	RMSE	log 10	δ_1	δ_2	δ_3
Mahjourian *et al.* [36]	0.159	5.912	–	0.784	0.923	0.970
Zhou *et al.* [58]	0.143	5.370	–	0.824	0.937	0.974
ST-CLSTM [55]	0.101	4.137	0.043	**0.890**	**0.970**	0.989
Patil *et al.* [37]	0.111	4.650	–	0.883	0.961	0.982
CVD [35]	0.130	4.876	–	0.878	0.946	0.970
Cao *et al.* [5]	0.109	4.366	0.047	0.872	0.962	0.986
Ours	**0.099**	**3.832**	**0.042**	0.886	0.968	**0.989**

4.5 Comparisons with state-of-the-art results

In this section, we evaluate our FMNet on the NYU Depth V2 dataset and the KITTI dataset. We compare our FMNet with some state-of-the-art results. The depth estimation results are reported in Table 1 and Table 2. Our approach achieves comparable depth estimation accuracy on the NYU Depth V2 dataset and the KITTI dataset. For some metrics, we outperform state-of-the-art methods based on generative adversarial networks, optical flow, or knowledge distillation. We only use the scale-invariant loss [12] as supervision, which shows the effectiveness of our design.

To further prove the effectiveness of our FMNet in temporal consistency, we compare the consistency metric *OPW* with previous state-of-the-art method [55] in Table 3. Our approach outperforms the ST-CLSTM [55] by a large margin. The ST-CLSTM causes obvious flickering and inconsistency between adjacent sequences due to its seriality and locality. It can only produce depth results frame by frame. By contrast, our FMNet has better characteristic of parallelism and globality. It can produce depth results of all input frames in one time. In Fig. 4, we show OPW_t values frame by frame. Our approach achieves better consistency whether inside or between input sequences. We also show some visual results of the NYU Depth V2 dataset in Fig. 5. The visualization results and the depth curves show that our FMNet achieves video depth results with higher accuracy and better consistency.

CVD [35] is one of the most famous methods in consistent video depth estimation. Based on the trained Midas [42] for single image depth estimation, CVD is in the test-time training paradigm. Their method highly relies on camera poses [43, 44] and optical flow [22]. When these information cannot be accurate in videos with dynamic scenes such as the KITTI dataset, CVD inevitably fails and causes large errors in depth results. They even need to finetune the optical flow model [22] to get the depth metrics reported in Table 2. On the KITTI dataset, our method outperforms CVD in depth estimation metrics. We also achieve more than 30% improvement of *OPW* as shown in Table 4. While CVD fails on the KITTI dataset, our method remains highly effective for consistent video depth estimation.

Table 3: Effectiveness of our method in temporal consistency. We compare depth accuracy and temporal consistency of ST-CLSTM [55] and our FMNet on the NYU depth V2 dataset. Our method outperforms the ST-CLSTM [55] by a large margin in video depth consistency.

Method	Rel	RMSE	log 10	δ_1	δ_2	δ_3	*OPW*
ST-CLSTM [55]	**0.131**	0.571	0.056	**0.833**	0.965	0.991	12.159
Ours	0.134	**0.452**	**0.056**	0.832	**0.968**	**0.992**	**6.425**

Table 4: Ablation study on the transformer and masking. We report the depth accuracy and temporal consistency on the KITTI dataset. We test the inference time on one GTX 1080Ti GPU with eight 640×480 frames as input.

Method	RMSE	log 10	δ_1	δ_2	δ_3	*OPW*	Time(s)
baseline	3.905	0.044	0.875	0.965	0.988	44.179	**3.22**
baseline+transformer	3.877	0.044	0.881	0.966	0.989	39.164	4.77
baseline+transformer+masking	**3.832**	**0.042**	**0.886**	**0.968**	**0.989**	**30.596**	3.36

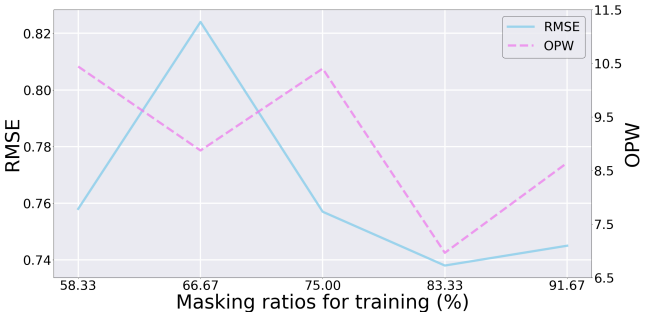

Figure 6: Ablation study on masking ratios. The X-axis represents masking ratios and Y-axis means RMSE and OPW. To reduce the experimental cost, we randomly choose 40 videos for training and 10 videos for OPW evaluation on the NYU Depth V2 dataset. The RMSE is evaluated on the public test dataset and cannot be compared with results in Table 1.

Some structure-from-motion (SFM) methods [47, 48] could achieve higher depth accuracy metrics. Those methods predict depth maps by feature matching over multiple frames. This idea benefits static scenes but does not account for dynamically moving objects. They need to mask the moving cars or people for pose estimation on the KITTI dataset. Those methods inevitably fail for videos with natural scenes or objects motion. By contrast, our method is not limited by camera poses. Our FMNet is also significantly faster than SFM-based methods due to the time-consuming pose estimation.

4.6 Ablation studies

4.6.1 Transformer and masking. We further ablate the design of transformer and masking in Table 4. The baseline is a CNN model without transformer and masking. If we add the transformer to the baseline and directly input N frames without masking, the depth accuracy and temporal consistency both improve but the inference speed decreases. With our masked frames predicting strategy, we force our model to learn the inter-frame temporal correlations. Our FMNet achieves better temporal consistency, higher depth accuracy, and faster inference speed than the model without masking. The inference speed improves because the heavy computational burden of full sequences is only sustained by the lightweight one-layer temporal decoder. Compared with the baseline, our FMNet achieves 30.75% temporal consistency improvement on the KITTI dataset.

4.6.2 Masking ratios for training. The masking ratios significantly influence the temporal consistency. We change the masking ratios for training. The experimental results are shown in Fig. 6. The masking ratio of 83.33% achieves the best accuracy and consistency. With 12 frames input and only 2 frames retained, the redundancy can be minimized while still preserving inter-frame correlations. If we use a higher masking ratio of 91.67% with only one frame remaining, the redundancy is lower, however, the temporal correlations are lost. As for lower masking ratios, the redundancy is higher and the consistency is worse. We adopt the very high masking ratio with only 2 frames remaining in our approach.

4.7 Inference speed comparison

We test the inference time of DeepV2D [48], CVD [35], and Robust-CVD [24] (officially published code). In Table 5, we make a comparison with those methods on one GTX 1080Ti GPU with eight 640×480 frames as input. We can see that our FMNet, whether with ResNet18 [19] or ResNext101 [53] as the backbone, has significantly faster inference speeds than methods using pose estimation or optical flow such as DeepV2D [48], CVD [35], and Robust-CVD [24].

Table 5: Inference speed comparison. We test the speed using one GTX 1080Ti GPU with eight 640×480 frames as input.

Method	Camera pose	Optical flow	Time(s)
Ours (ResNet18)			1.32
Ours (ResNext101)			3.36
DeepV2D (pose pre-computed) [48]	✓		11.65
DeepV2D (pose estimation) [48]	✓		55.62
CVD [35]	✓	✓	376.32
Robust-CVD [24]	✓	✓	252.17

5 CONCLUSION

In this work, we provide a new perspective for consistent video depth estimation. Observing the redundancy of videos, we propose the frame masking network (FMNet) to mine consistency with the masking-and-reconstructing strategy. Randomly masking some input frames, we compel our model to estimate the depth of masked frames based on unmasked ones. The implicit inter-frame correlations and the larger temporal receptive field lead to better temporal consistency compared with previous approaches. Further analysis shows that with a very high masking ratio, the FMNet generates the most accurate and consistent results on the NYU Depth V2 dataset. The phenomenon reveals that the consistency can be directly modeled from the videos. There are also some limitations of our method. For example, we only use the scale-invariant loss as supervision. If ground truth exists flickering, it might lead to a measure of inconsistency in depth results. We can further introduce supervisions such as geometric constraints for this situation.

6 ACKNOWLEDGMENTS

This work was funded by Adobe.

REFERENCES

[1] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. 2021. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 6836–6846.

[2] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. 2022. BEiT: BERT Pre-Training of Image Transformers. In *International Conference on Learning Representations*.

[3] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. 2021. Adabins: Depth estimation using adaptive bins. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 4009–4018.

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *Advances in neural information processing systems*, Vol. 33. 1877–1901.

[5] Yuanzhouhan Cao, Yidong Li, Haokui Zhang, Chao Ren, and Yifan Liu. 2021. Learning Structure Affinity for Video Depth Estimation. In *Proceedings of the 29th ACM International Conference on Multimedia*. 190–198.

[6] Yuanzhouhan Cao, Zifeng Wu, and Chunhua Shen. 2017. Estimating depth from monocular images as classification using deep fully convolutional residual networks. *IEEE Transactions on Circuits and Systems for Video Technology* 28, 11 (2017), 3174–3182.

[7] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *European Conference on Computer Vision (ECCV)*, Vol. 12346. Springer, 213–229.

[8] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. 2020. Generative pretraining from pixels. In *International conference on machine learning*. PMLR, 1691–1703.

[9] Xinjing Cheng, Peng Wang, and Ruigang Yang. 2018. Depth estimation via affinity learned with convolutional spatial propagation network. In *European Conference on Computer Vision (ECCV)*, Vol. 11220. 108–125.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. 4171–4186.

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*.

[12] David Eigen, Christian Puhrsch, and Rob Fergus. 2014. Depth map prediction from a single image using a multi-scale deep network. In *Advances in neural information processing systems*, Vol. 27. 2366–2374.

[13] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. 2021. Multiscale vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 6824–6835.

[14] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. 2018. Deep ordinal regression network for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2002–2011.

[15] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. 2013. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research* 32, 11 (2013), 1231–1237.

[16] C. Godard, O. Aodha, M. Firman, and G. Brostow. 2019. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 3828–3838.

[17] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*, Vol. 27.

[18] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2021. Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377* (2021).

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778.

[20] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* 2, 7 (2015).

[21] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[22] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. 2017. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2462–2470.

[23] Kevin Karsch, Ce Liu, and Sing Bing Kang. 2014. Depth transfer: Depth extraction from video using non-parametric sampling. *IEEE transactions on pattern analysis and machine intelligence* 36, 11 (2014), 2144–2158.

[24] Johannes Kopf, Xuejian Rong, and Jia-Bin Huang. 2021. Robust consistent video depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1611–1621.

[25] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. 2016. Deeper depth prediction with fully convolutional residual networks. In *2016 Fourth international conference on 3D vision (3DV)*. IEEE, 239–248.

[26] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.

[27] Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. 2019. From big to small: Multi-scale local planar guidance for monocular depth estimation. *arXiv preprint arXiv:1907.10326* (2019).

[28] Ruibo Li, Ke Xian, Chunhua Shen, Zhiguo Cao, Hao Lu, and Lingxiao Hang. 2018. Deep attention-based classification network for robust depth prediction. In *Asian Conference on Computer Vision (ACCV)*. 663–678.

[29] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. 2017. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1925–1934.

[30] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2117–2125.

[31] Yifan Liu, Ke Chen, Chris Liu, Zengchang Qin, Zhenbo Luo, and Jingdong Wang. 2019. Structured knowledge distillation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2604–2613.

[32] Yifan Liu, Changyong Shu, Jingdong Wang, and Chunhua Shen. 2020. Structured knowledge distillation for dense prediction. *IEEE transactions on pattern analysis and machine intelligence* (2020).

[33] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 10012–10022.

[34] Zhouyong Liu, Shun Luo, Wubin Li, Jingben Lu, Yufan Wu, Shilei Sun, Chunguo Li, and Luxi Yang. 2020. Convtransformer: A convolutional transformer network for video frame synthesis. *arXiv preprint arXiv:2011.10185* (2020).

[35] Xuan Luo, Jia-Bin Huang, Richard Szeliski, Kevin Matzen, and Johannes Kopf. 2020. Consistent video depth estimation. *ACM Transactions on Graphics (ToG)* 39, 4 (2020), 71–1.

[36] Reza Mahjourian, Martin Wicke, and Anelia Angelova. 2018. Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5667–5675.

[37] Vaishakh Patil, Wouter Van Gansbeke, Dengxin Dai, and Luc Van Gool. 2020. Don't forget the past: Recurrent depth estimation from monocular video. *IEEE Robotics and Automation Letters* 5, 4 (2020), 6813–6820.

[38] Juewen Peng, Zhiguo Cao, Xianrui Luo, Hao Lu, Ke Xian, and Jianming Zhang. 2022. BokehMe: When Neural Rendering Meets Classical Rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 16283–16292.

[39] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *OpenAI blog* (2018).

[40] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog* (2019).

[41] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. 2021. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 12179–12188.

[42] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. 2020. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence* 44, 03 (2020), 1623–1637.

[43] Johannes Lutz Schönberger and Jan-Michael Frahm. 2016. Structure-from-Motion Revisited. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 4104–4113.

[44] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. 2016. Pixelwise View Selection for Unstructured Multi-View Stereo. In *European Conference on Computer Vision (ECCV)*, Vol. 9907. 501–518.

[45] Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing* 45, 11 (1997), 2673–2681.

[46] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. 2012. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision (ECCV)*. Springer, 746–760.

[47] Chengzhou Tang and Ping Tan. 2018. BA-Net: Dense Bundle Adjustment Networks. In *International Conference on Learning Representations*.

[48] Zachary Teed and Jia Deng. 2019. DeepV2D: Video to Depth with Differentiable Structure from Motion. In *International Conference on Learning Representations*.
[49] Zachary Teed and Jia Deng. 2020. Raft: Recurrent all-pairs field transforms for optical flow. In *European Conference on Computer Vision (ECCV)*. Springer, 402–419.
[50] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, Vol. 30.
[51] Ke Xian, Chunhua Shen, Zhiguo Cao, Hao Lu, Yang Xiao, Ruibo Li, and Zhenbo Luo. 2018. Monocular Relative Depth Perception With Web Stereo Data Supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 311–320.
[52] Ke Xian, Jianming Zhang, Oliver Wang, Long Mai, Zhe Lin, and Zhiguo Cao. 2020. Structure-Guided Ranking Loss for Single Image Depth Prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 608–617.
[53] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. 2017. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 1492–1500.
[54] Dan Xu, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. 2018. Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 675–684.
[55] Haokui Zhang, Chunhua Shen, Ying Li, Yuanzhouhan Cao, Yu Liu, and Youliang Yan. 2019. Exploiting temporal consistency for real-time video depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 1725–1734.
[56] Xuaner Zhang, Kevin Matzen, Vivien Nguyen, Dillon Yao, You Zhang, and Ren Ng. 2019. Synthetic defocus and look-ahead autofocus for casual videography. *ACM Transactions on Graphics (TOG)* 38, 4 (2019).
[57] Zhoutong Zhang, Forrester Cole, Richard Tucker, William T Freeman, and Tali Dekel. 2021. Consistent depth of moving objects in video. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–12.
[58] Lipu Zhou, Jiamin Ye, Montiel Abello, Shengze Wang, and Michael Kaess. 2018. Unsupervised learning of monocular depth estimation with bundle adjustment, super-resolution and clip loss. *arXiv preprint arXiv:1812.03368* (2018).
[59] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. 2021. Deformable detr: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*.