FutureDepth: Learning to Predict the Future Improves Video Depth Estimation (ECCV 2024?)

2024.04.18

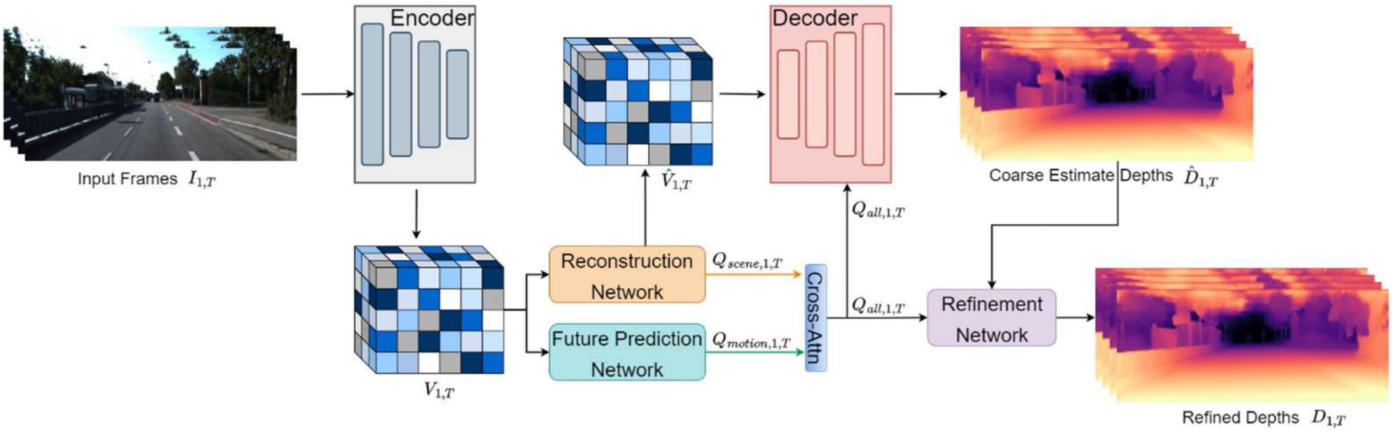
Motivation & Contribution

1. F-Net adopts a **multi-frame/time step future prediction loss based on auto-regressive sampling of future frames**. Extract **stronger motion, correspondence cues** at inference time for better temporally consistent depth prediction.

2. R-Net is trained to perform **MAE on features of a consecutive set of frames with a learnable, adaptive masking strategy.** This encourages R-Net to **leverage critical scene features distributed across frames for reconstruction and thus, understand the multi-view correspondences**.

每个子问题都没有得到完美解决，并且给下一步增加了噪音，增加了管道整体工作所需的复杂性和工程工作量。
在这方面，每个子问题之间缺乏沟通就很能说明问题：如果它们互相帮助似乎更合理，即密集重建自然应该受益于为恢复相机姿势而构建的稀疏场景，反之亦然。
最重要的是，该流程中的关键步骤很脆弱

Method: Pipeline



具体来说，为了预测某个像素位置的未来特征，F-Net 需要找到当前和之前时间步中可用的相应特征。这本质上使 F-Net 能够理解底层运动和多帧对应关系，以及较长上下文中的运动。

Method: F-Net

$$V_{1,T} = \{F_1, , F_2, ..., F_T\} \in \mathbb{R}^{H' \times W' \times (T \cdot C)}$$

$$\mathcal{L}_F = \sum_{i=1}^{L} ||\tilde{V}_{1+i,T+i} - V_{1+i,T+i}||_2,$$

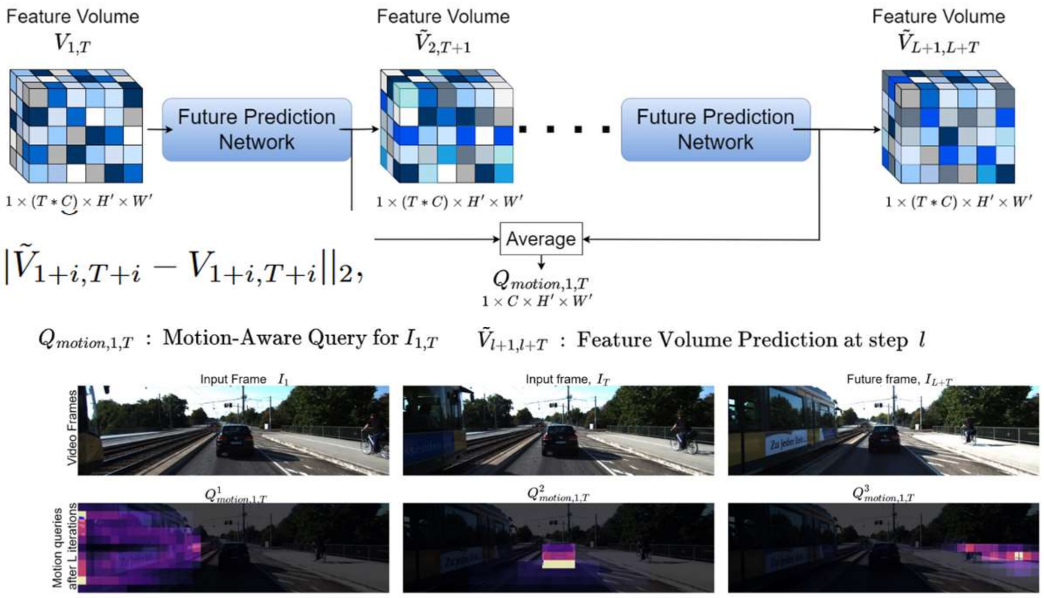
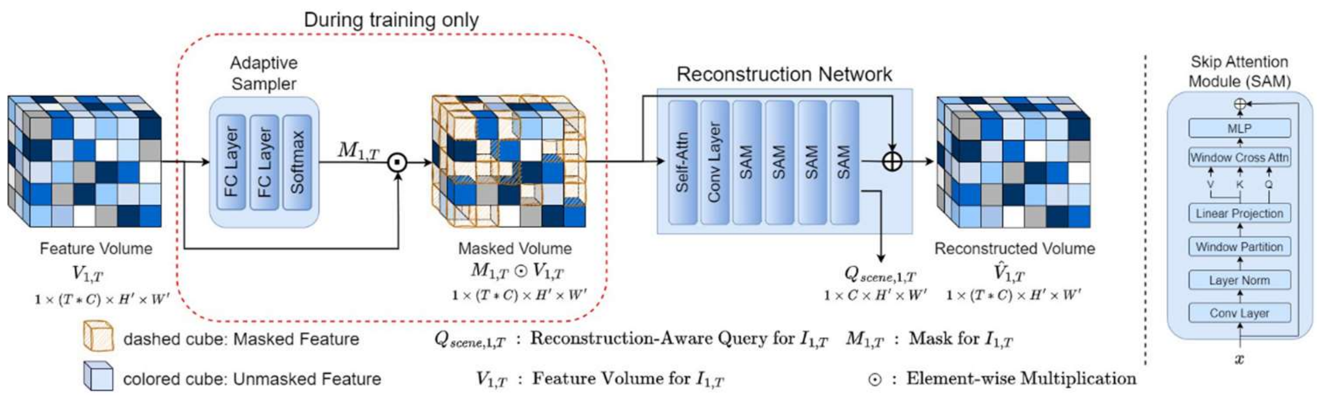


Fig. 4: Example motion query $Q_{motion,1,T}$ generated using future prediction network. Here we show three example channels in $Q_{motion,1,T}$, with $T = 4$, and $L = 6$.

具体来说，为了预测某个像素位置的未来特征，F-Net 需要找到当前和之前时间步中可用的相应特征。这本质上使 F-Net 能够理解底层运动和多帧对应关系，以及较长上下文中的运动。

Method: R-Net



我们观察到掩模生成器学习以鼓励 R-Net 利用多视图对应的方式对帧进行掩模。特别是，它跨帧屏蔽了同一对象的不同部分；例如，参见图 5 中的白色卡车及其上方的掩模。因此，R-Net 需要利用跨帧分布的信息来重建完整的特征量。

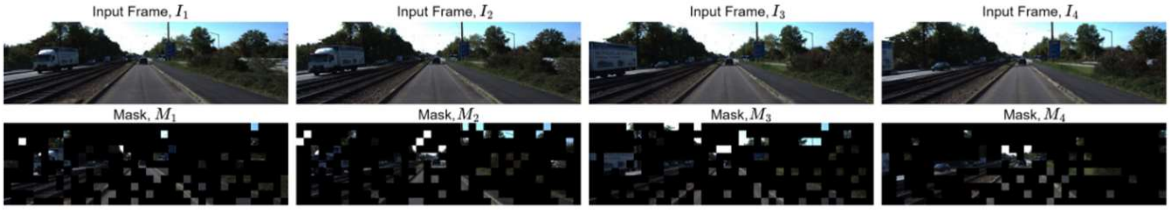
自适应采样器生成的蒙版侧重于跨帧的重要对象，例如货车、汽车、电车、公共汽车、铁路轨道和道路边界等。

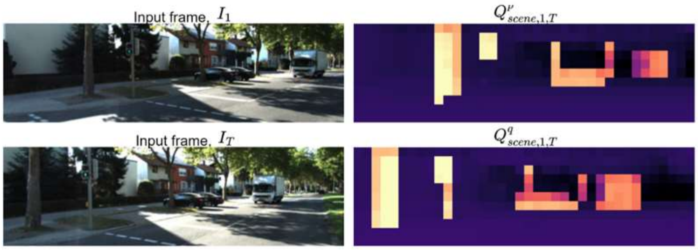
Qscene 在不同的时间步捕获重要的前景信息（例如卡车、停放的汽车、附近的树）

Method: R-Net (0为mask掉，1保留)

$$\mathcal{L}_R = ||(1 - M_{1,T}) \odot (\hat{V}_{1,T} - V_{1,T})||_2 + \mathcal{L}_D(D_{1,T}, D^{gt}_{1,T})$$

use the output features from the last layer as Q_{scene}.





我们观察到掩模生成器学习以鼓励 R-Net 利用多视图对应的方式对帧进行掩模。特别是，它跨帧屏蔽了同一对象的不同部分；例如，参见图 5 中的白色卡车及其上方的掩模。因此，R-Net 需要利用跨帧分布的信息来重建完整的特征量。

自适应采样器生成的蒙版侧重于跨帧的重要对象，例如货车、汽车、电车、公共汽车、铁路轨道和道路边界等。

Qscene 在不同的时间步捕获重要的前景信息（例如卡车、停放的汽车、附近的树）

1. First pretrain the FutureDepth encoder and decoder to perform depth prediction, without using F-Net, R-Net, and the refinement network. (5 epochs)

2. Train R-Net to perform feature volume reconstruction with random masking. In this step, we only use the L2 loss to train R-Net, and do not update the encoder or decoder. (3 epoch)

pretraining $en(\cdot)$, $de(\cdot)$
for epoch = 1→5 **do**
  for $I_{1,T}, D^{gt}_{1,T} \in \mathcal{D}$ **do**
    $D_{1,T} = de(en(I_{1,T}))$
    SILogLoss $(D_{1,T}, D^{gt}_{1,T})$
    Update parameters of $en(\cdot)$, $de(\cdot)$
  end for
end for
pretraining reconstruction network $g(\cdot)$######
for epoch = 1→3 **do**
  freeze $en(\cdot)$ weights
  for $I_{1,T}, D^{gt}_{1,T} \in \mathcal{D}$ **do**
    $V_{1,T} = en(I_{1,T})$
    generate random mask $M_{1,T}$
    $\hat{V}_{1,T} = g(M_{1,T} \odot V_{1,T})$
    L2-loss between $V_{1,T}$ and $\hat{V}_{1,T}$
    Update parameters of $g(\cdot)$
  end for
end for

预训练阶段，我们训练编码器和解码器 5 个 epoch，随后训练 R-Net 3 个 epoch。在主要训练部分，我们训练 FutureDepth 的所有组件 15 个 epoch。

3. Initialize both F-Net and R-Net with the pretrained R-Net weights, freeze encoder and decoder, and train the adaptive mask generator, F-Net, and R-Net simultaneously. We compute the LF , LA, and LR losses and learn the weights for F-Net, mask generator, and R-Net, respectively.

$$\mathcal{L}_F = \sum_{i=1}^{L} ||\tilde{V}_{1+i,T+i} - V_{1+i,T+i}||_2 \qquad \mathcal{L}_R = ||(1 - M_{1,T}) \odot (\hat{V}_{1,T} - V_{1,T})||_2 + \mathcal{L}_D(D_{1,T}, D^{gt}_{1,T})$$

4. Finally, freeze F-Net, mask generator, and R-Net, and train the encoder, decoder, and refinement network using the following loss:

$$\mathcal{L}_{D,final} = \frac{1}{NT}\sum_{i=0}^{N}\sum_{t=1}^{T}\mathcal{L}_D(D^i_t, D^{gt}_t)$$

for every epoch **do**
  for $I_{1,T}, D^{gt}_{1,T} \in \mathcal{D}$ **do**
    ### step-1 updating $h(\cdot)$, $s(\cdot)$, $g(\cdot)$ weights###
    freeze $en(\cdot)$, $de(\cdot)$ weights
    $V_{1,T} = en(I_{1,T})$; $M_{1,T} = s(V_{1,T})$
    $\tilde{V}_{1,T} = V_{1,T}$
    for i=1→ L **do**
      $\tilde{V}_{i+1,i+T} = h(\tilde{V}_{i,i+T-1})$
    end for
    $\hat{V}_{1,T}$, $Q_{scene,1,T} = g(M_{1,T} \odot V_{1,T})$
    $Q_{all,1,T} =$ cross-attn$(Q_{scene,1,T}, Q_{motion,1,T})$
    $D_{1,T} = de(\hat{V}_{1,T}, Q_{all,1,T})$
    compute loss $\mathcal{L}_F$ in Eq. 1 (main paper)
    compute loss $\mathcal{L}_A$ (refer section 2.2)
    compute loss $\mathcal{L}_R$ in Eq. 2 (main paper)
    Update parameters of $h(\cdot)$, $s(\cdot)$, $g(\cdot)$
    ### step-2 updating FutureDepth's $en(\cdot)$, $de(\cdot)$, $rf(\cdot)$ weights###
    freeze $s(\cdot)$, $g(\cdot)$, $h(\cdot)$ weights
    $V_{1,T} = en(I_{1,T})$
    $\hat{V}_{1,T}$, $Q_{scene,1,T} = g(V_{1,T})$
    get $Q_{motion,1,T}$ from future prediction F-Net $h(\cdot)$
    $Q_{all,1,T} =$ cross-attn$(Q_{scene,1,T}, Q_{motion,1,T})$
    $D^0_{1,T} = de(\hat{V}_{1,T}, Q_{all,1,T})$
    for i=1→ N **do**
      $D^i_{1,T} = rf(D^{i-1}_{1,T}, Q_{all,1,T})$
    end for
    compute loss $\mathcal{L}_{D,final}$ in Eq. 3 (main paper)
    Update parameters of FutureDepth 's $en()$, $de(\cdot)$, $rf(\cdot)$ weights
  end for
end for

预训练阶段，我们训练编码器和解码器 5 个 epoch，随后训练 R-Net 3 个 epoch。在主要训练部分，我们训练 FutureDepth 的所有组件 15 个 epoch。

Experiments

Method	NYUDV2			Sintel		
	$\delta < 1.25 \uparrow$	Abs Rel$\downarrow$	OPW$\downarrow$	$\delta < 1.25 \uparrow$	Abs Rel$\downarrow$	OPW$\downarrow$
ST-CLSTM [59]	0.833	0.131	0.645	0.351	0.517	0.585
FMNet [51]	0.832	0.134	0.387	0.357	0.513	0.521
R-CVD [21]	0.886	0.103	0.394	0.521	0.422	0.475
Many-Depth-FS [53]	0.865	0.096	0.428	0.492	0.487	0.540
NVDS [52]	0.950	0.072	0.364	0.591	0.335	0.424
MAMo [57]	0.942	0.074	0.388	0.579	0.358	0.493
Baseline (ours)	0.917	0.093	0.480	0.477	0.504	0.611
FutureDepth (ours)	**0.981**	**0.063**	**0.303**	**0.623**	**0.296**	**0.392**

我们最终扩展了训练集（2,400 到 14,410 帧），这显着减少了误差，表明大数据集是自监督深度训练中非常重要的元素

Experiments

Type	Method	Encoder	Abs Rel↓	Sq Rel↓	RMSE↓	RMSE_{log} ↓	$\delta < 1.25$ ↑
SF	AdaBins [4]	EfficientNet	0.058	0.190	2.360	0.088	0.964
	BinsFormer [25]	Swin-L	0.052	0.151	2.098	0.079	0.975
	NeWCRFs [58]	Swin-L	0.052	0.155	2.129	0.079	0.974
	PixelFormer [2]	Swin-L	0.051	0.149	2.081	0.077	0.976
	iDisc [35]	Swin-L	0.050	0.145	2.067	0.077	0.977
	GEDepth [56]	[24]	0.048	0.142	2.050	0.076	0.976
MF	FlowGRU [11]	[11]	0.112	0.700	4.260	0.184	0.881
	RDE-MV [34]	ResNet18†	0.111	0.821	4.650	0.187	0.821
	STAD [22]	[26]†	0.109	0.594	3.312	0.153	0.889
	Patil *et.al.* [34]	ConvLSTM†	0.102	–	4.148	–	0.884
	ST-CLSTM [59]	ResNet18	0.101	–	4.137	–	0.890
	NeuralRGB [26]	CNN-based†	0.100	–	2.829	–	0.931
	Cao *et.al.* [7]	–	0.099	–	3.832	–	0.886
	FMNet [51]	ResNeXt-101	0.099	–	3.744	0.129	0.888
	Flow2Depth [55]	[30]†	0.081	0.488	3.651	0.146	0.912
	TC-Depth-FS [38]	ResNet50	0.071	0.330	3.222	0.108	0.922
	ManyDepth-FS [53]	ResNet50	0.069	0.342	3.414	0.111	0.930
	ManyDepth-FS [53]	Swin-L	0.060	0.248	2.747	0.099	0.955
	NVDS [52]	DPT-L [36]	0.052	0.159	2.101	0.077	0.976
	MAMo [57]	Swin-L	0.049	0.130	1.989	0.072	0.977
MF (Ours)	Baseline	ResNet34	0.063	0.219	2.521	0.098	0.957
		Swin-B	0.055	0.162	2.163	0.082	0.973
		Swin-L	0.053	0.154	2.094	0.079	0.975
		Dinov2 (ViT-L)	0.051	0.141	2.064	0.076	0.979
	FutureDepth	ResNet34	0.054	0.179	2.016	0.087	0.965
		Swin-B	0.049	0.129	1.998	0.077	0.976
		Swin-L	0.044	0.119	1.920	0.068	0.983
		Dinov2 (ViT-L)	**0.041**	**0.117**	**1.856**	**0.066**	**0.984**

我们最终扩展了训练集（2,400 到 14,410 帧），这显着减少了误差，表明大数据集是自监督深度训练中非常重要的元素

Experiments

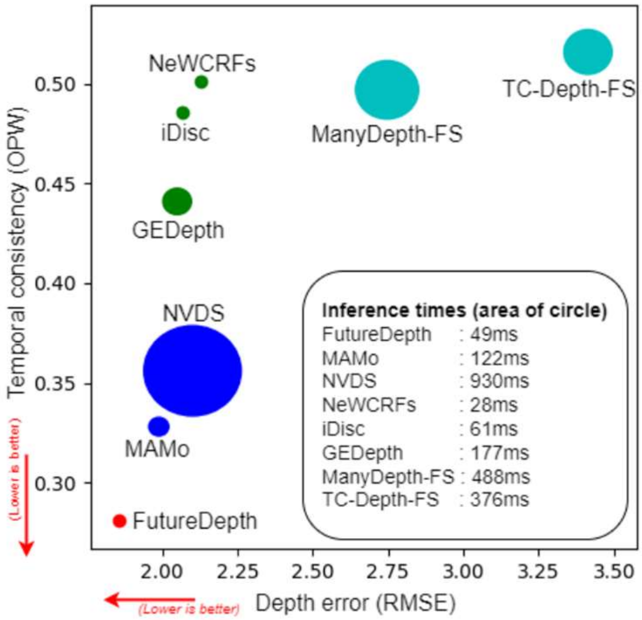
Model	Type	R-Net	AM	F-Net	Refine	Sq Rel↓	RMSE↓	$\delta < 1.25\uparrow$	OPW↓
Baseline	SF					0.156	2.098	0.974	0.544
Baseline	MF					0.154	2.094	0.975	0.540
				✓		0.129	1.978	0.981	0.311
		✓ (RM)				0.148	2.040	0.976	0.478
		✓	✓			0.136	1.999	0.980	0.416
		✓	✓	✓		0.122	1.931	**0.983**	0.284
FutureDepth	MF	✓	✓	✓	✓	**0.119**	**1.920**	**0.983**	**0.281**

Table 10: Ablation study on different masking ratios of adaptive sampler during inference on KITTI (Eigen split) dataset. We perform this experiment using Swin-L for FutureDepth encoder. Here we set $L = 4$, $T = 4$ and $N = 3$.

Metric	0.0	0.2	0.4	0.6	0.9
RMSE↓	1.920	1.906	1.892	1.911	1.956
Sq Rel↓	0.119	0.114	0.108	0.111	0.133

我们最终扩展了训练集（2,400 到 14,410 帧），这显着减少了误差，表明大数据集是自监督深度训练中非常重要的元素

Experiments



我们最终扩展了训练集（2,400 到 14,410 帧），这显着减少了误差，表明大数据集是自监督深度训练中非常重要的元素

DIG

Thanks