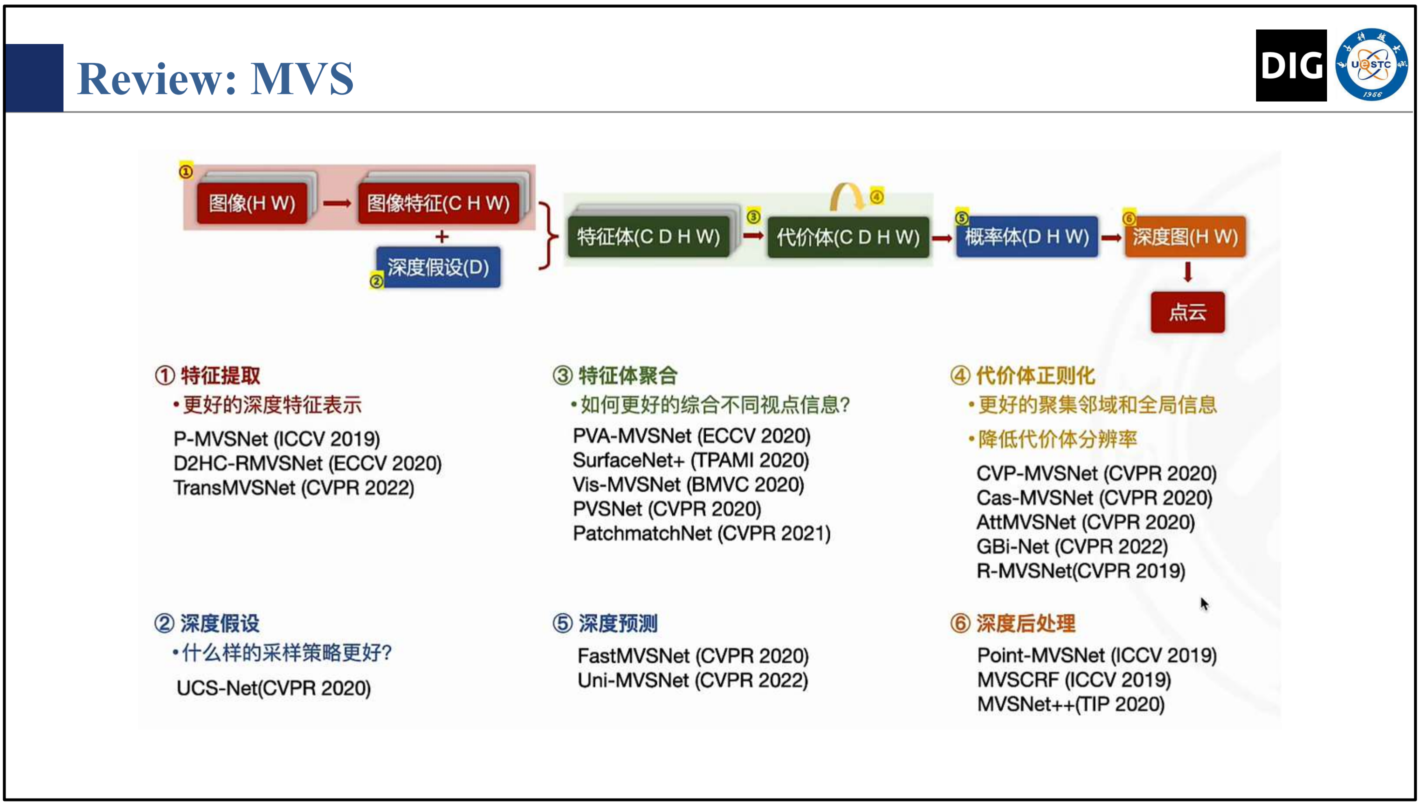
GeoMVSNet: Learning Multi-View Stereo with Geometry Perception (CVPR 2023)

2024.03.14



1.

2. 频域滤波策略来有效地减轻冗余的高频纹理，而无需产生更多的学习参数，并利用嵌入不同频率层次的几何结构来进行逐渐精细的深度估计

3.

Contribution

DIG

1. Geometric prior guided feature fusion and the probability volume geometry embedding approaches for **robust cost matching**.

2. Enhance geometry awareness via the frequency domain filtering strategy and adopt the idea of curriculum learning for progressively introducing geometric clues from easy to difficult.

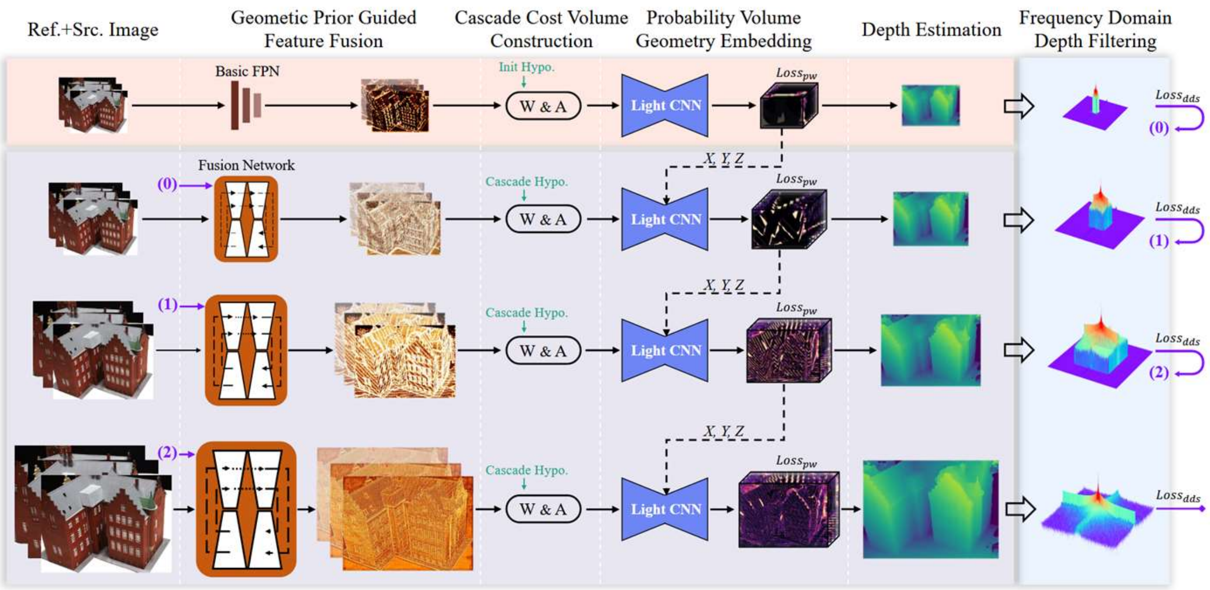
3. Gaussian-Mixture Model assumption and build the full-scene geometry perception loss function.

1.

2. 频域滤波策略来有效地减轻冗余的高频纹理，而无需产生更多的学习参数，并利用嵌入不同频率层次的几何结构来进行逐渐精细的深度估计

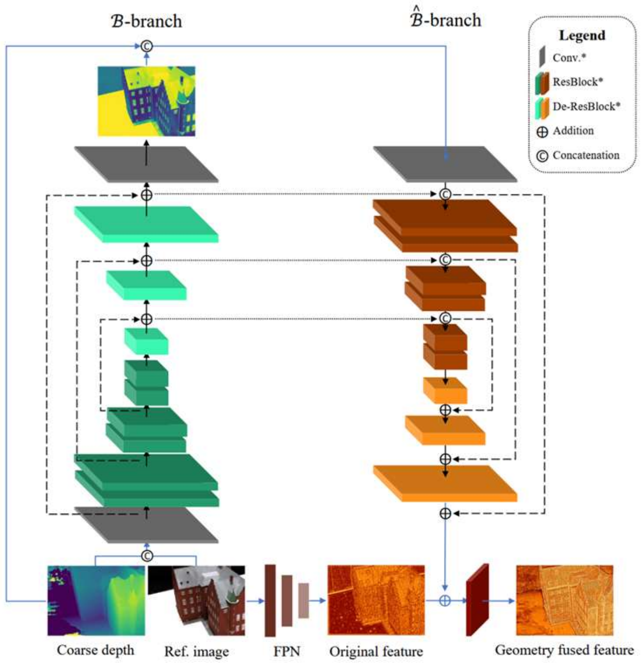
3.

Method: Pipeline



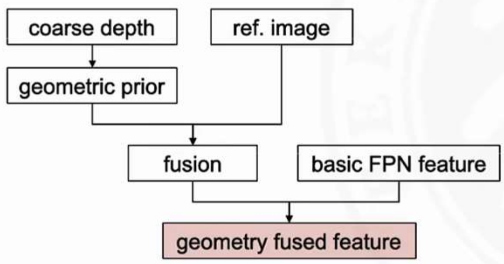
optical flow: instance tracking

Geometry Fusion



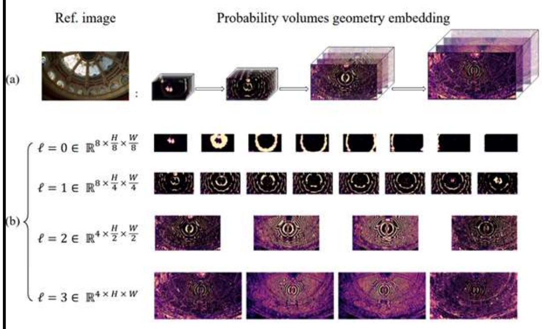
$$Branch(z) = \hat{\mathcal{B}}([D_{\uparrow}^{\ell}, \mathcal{B}([I_0^{\ell+1}, D_{\uparrow}^{\ell}])) ,$$

$$F_0^{\ell+1}(z) = Fusion\{\bar{F}_0^{\ell+1}(z) \oplus Branch(z)\}$$



✓ Strengthen the discrimination and structure of features.
✓ Solid foundation for robust aggregation.

Probability volume geometry embedding



3D ~~convolution~~ → 2D lightweight regularization network

\+

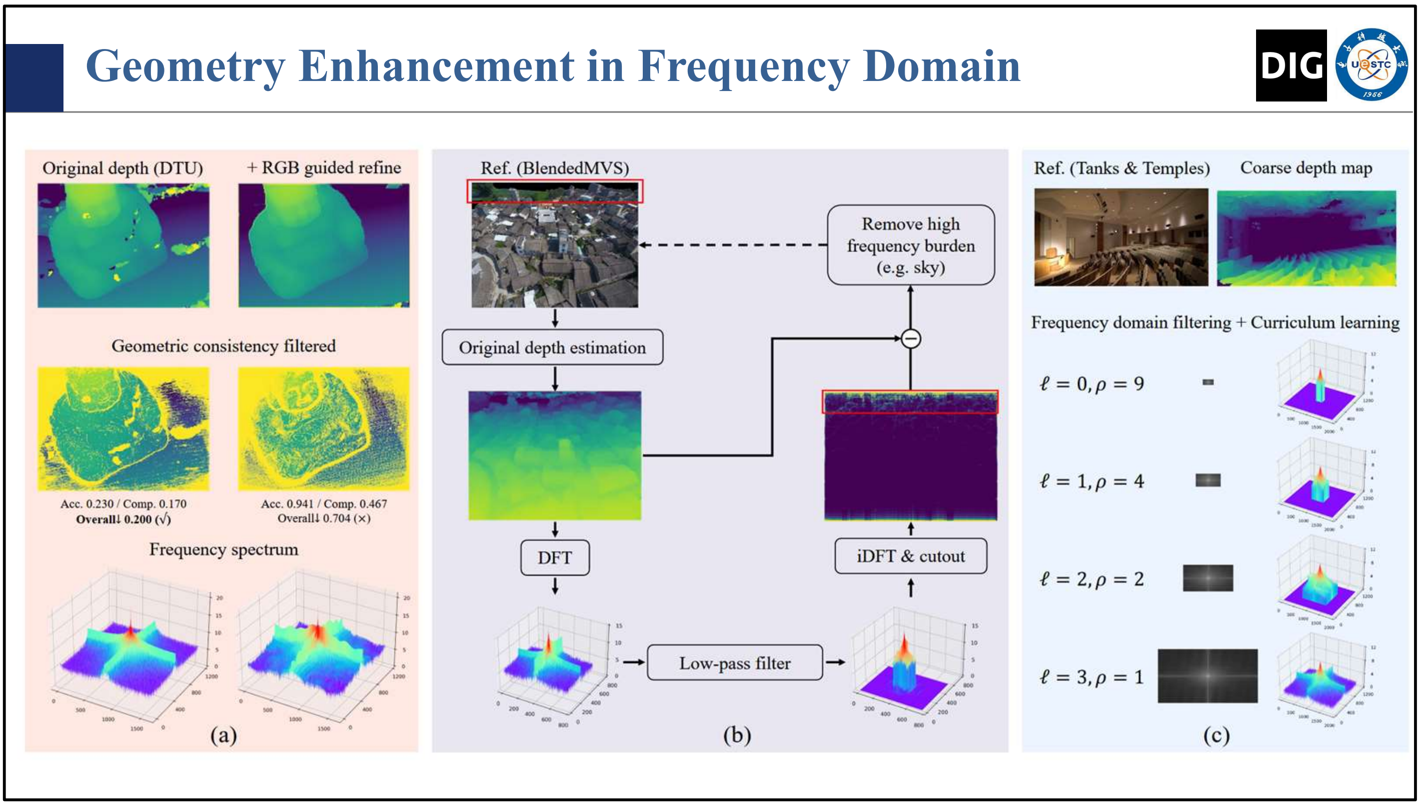
3D “position maps”

- ✓ Use depth-wise conv. instead of full 3D conv. to make the pipeline more efficient.
- ✓ Use geometric prior to compensate the reconstruction quality.
- ✓ W/o external overload.


```
kernel_size=(1,3,3),
```

	Cost volume (section 3.2)	
	Cost Volume	½D×½H×½W×2F
	Learning regularization (section 3.3)	
19	3-D conv, 3×3×3, 32 features	½D×½H×½W×F
20	3-D conv, 3×3×3, 32 features	½D×½H×½W×F
21	From Cost Volume: 3-D conv, 3×3×3, 64 features, stride 2	¼D×¼H×¼W×2F
22	3-D conv, 3×3×3, 64 features	¼D×¼H×¼W×2F
23	3-D conv, 3×3×3, 64 features	¼D×¼H×¼W×2F
24	From 21: 3-D conv, 3×3×3, 64 features, stride 2	⅛D×⅛H×⅛W×2F
25	3-D conv, 3×3×3, 64 features	⅛D×⅛H×⅛W×2F
26	3-D conv, 3×3×3, 64 features	⅛D×⅛H×⅛W×2F
27	From 24: 3-D conv, 3×3×3, 64 features, stride 2	1/16D×1/16H×1/16W×2F
28	3-D conv, 3×3×3, 64 features	1/16D×1/16H×1/16W×2F
29	3-D conv, 3×3×3, 64 features	1/16D×1/16H×1/16W×2F
30	From 27: 3-D conv, 3×3×3, 128 features, stride 2	1/32D×1/32H×1/32W×4F
31	3-D conv, 3×3×3, 128 features	1/32D×1/32H×1/32W×4F
32	3-D conv, 3×3×3, 128 features	1/32D×1/32H×1/32W×4F
33	3×3×3, 3-D transposed conv, 64 features, stride 2	1/16D×1/16H×1/16W×2F
	add layer 33 and 29 features (residual connection)	1/16D×1/16H×1/16W×2F
34	3×3×3, 3-D transposed conv, 64 features, stride 2	⅛D×⅛H×⅛W×2F
	add layer 34 and 26 features (residual connection)	⅛D×⅛H×⅛W×2F
35	3×3×3, 3-D transposed conv, 64 features, stride 2	¼D×¼H×¼W×2F
	add layer 35 and 23 features (residual connection)	¼D×¼H×¼W×2F
36	3×3×3, 3-D transposed conv, 32 features, stride 2	½D×½H×½W×F
	add layer 36 and 20 features (residual connection)	½D×½H×½W×F
37	3×3×3, 3-D trans conv, 1 feature (no ReLu or BN)	D×H×W×1

```
if g1 is not None:
    x = torch.cat((x, g1), 1)
```



那我们都知道神经网络，它对于高频信息的建模能力本来就比较差。这个房子跟这个云这个深度变化特别大，就是高频信息吗

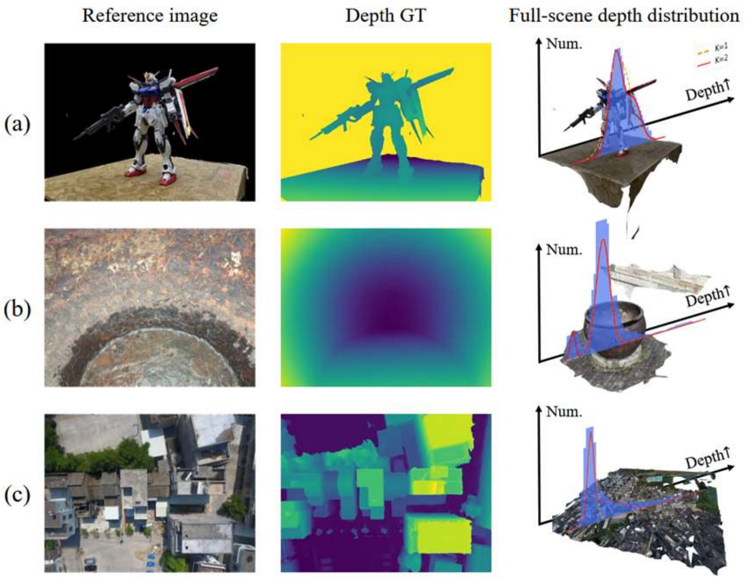
比如说这里面我们展示了一个航拍数据集的一个重建效果，那对于这样的一个多视点深度估计的深度图，通过这个离散的快速傅里叶变换，把它变换到频域。然后再通过低通滤波器把高频的信息扣掉

然后我们再通过反复的变换把它变换回来，把这个航拍时候的天空呃这些信息给抠掉。

在处理粗到精的过程中，首先在粗糙阶段进行更多的特征提取，因为在我们认为的粗糙阶段，高频信息不够准确。逐步提高网络分辨率和性能，逐渐将高频信息重新引入，直到最后一层，使得网络能够充分利用所有的高频信息。

普通方法增加了外部的复杂依赖嘛

Mixed-Gaussian Depth Distribution Model



$$Loss_{pw} = \sum_{z \in \Psi} (-P_{GT}(z)\, log[P(z)])\ ,$$

$$Loss_{dds} = \sum_{m=0, z \in \Upsilon}^{M'} \tilde{p}(z)\, (log\, \tilde{p}(z) - log\, \mathcal{N}_{GT}(z))\ , \quad (11)$$

$$\Upsilon = \Psi \cap \bigcup_{i=1}^{K} \{(\mu_i - 3\sigma_i, \mu_i + 3\sigma_i)\}\ , \quad (12)$$

$$Loss = \sum_{\ell=0}^{L} (\lambda_1^{\ell}\ Loss_{pw} + \lambda_2^{\ell}\ Loss_{dds})\ .$$

所以这里面我们就是把一个完全离散的这个约束转换到一个相对连续的一个约束
这里面我们用的是kl散度去度量这个全场景的一个这个深度分布的相似性

M' = 48

Experiments

	Method	Acc. (*mm*)	Comp. (*mm*)	Overall↓ (*mm*)
	Gipuma [12]	**0.283**	0.873	0.578
	COLMAP [36]	0.400	0.664	0.532
	R-MVSNet [57]	0.383	0.452	0.417
	CasMVSNet [14]	0.325	0.385	0.355
	CVP-MVSNet [54]	0.296	0.406	0.351
Post-pyramid Era	EPP-MVSNet [27]	0.413	0.296	0.355
	CER-MVS [28]	0.359	0.305	0.332
	RayMVSNet [48]	0.341	0.319	0.330
	Effi-MVSNet [45]	0.321	0.313	0.317
	CDS-MVSNet [13]	0.352	0.280	0.316
	NP-CVP-MVSNet [53]	0.356	0.275	0.315
	UniMVSNet [32]	0.352	0.278	0.315
	TransMVSNet [8]	0.321	0.289	0.305
	GBi-Net* [29]	0.312	0.293	0.303
	MVSTER* [46]	0.340	0.266	0.303
	GeoMVSNet (Ours)	0.331	**0.259**	**0.295**

Method	Intermediate									Advanced						
	Mean↑	Family	Francis	Horse	L.H.	M60	Panther	P.G.	Train	**Mean↑**	Aud.	Bal.	Cou.	Mus.	Pal.	Tem.
COLMAP [36]	42.14	50.41	22.25	25.63	56.43	44.83	46.97	48.53	42.04	27.24	16.02	25.23	34.70	41.51	18.05	27.94
CasMVSNet [14]	56.42	76.36	58.45	46.20	55.53	56.11	54.02	58.17	46.56	31.12	19.81	38.46	29.10	43.87	27.36	28.11
PatchmatchNet [44]	53.15	66.99	52.64	43.24	54.87	52.87	49.54	54.21	50.81	32.31	23.69	37.73	30.04	41.80	28.31	32.29
CER-MVS [28]	64.82	81.16	64.21	50.43	**70.73**	63.85	63.99	**65.90**	58.25	40.19	25.95	45.75	39.65	51.75	35.08	42.97
Effi-MVSNet [45]	56.88	72.21	51.02	51.78	58.63	58.71	56.21	57.07	49.38	34.39	20.22	42.39	33.73	45.08	29.81	35.09
UniMVSNet [32]	64.36	81.20	66.43	53.11	63.46	**66.09**	64.84	62.23	57.53	38.96	28.33	44.36	39.74	52.89	33.80	34.63
TransMVSNet [8]	63.52	80.92	65.83	**56.94**	62.54	63.06	60.00	60.20	**58.67**	37.00	24.84	44.59	34.77	46.49	34.69	36.62
GBi-Net [29]	61.42	79.77	**67.69**	51.81	61.25	60.37	55.87	60.67	53.89	37.32	29.77	42.12	36.30	47.69	31.11	36.93
MVSTER [46]	60.92	80.21	63.51	52.30	61.38	61.47	58.16	58.98	51.38	37.53	26.68	42.14	35.65	49.37	32.16	39.19
GeoMVSNet (Ours)	**65.89**	**81.64**	67.53	55.78	68.02	65.49	**67.19**	63.27	58.22	**41.52**	**30.23**	**46.53**	**39.98**	**53.05**	**35.98**	**43.34**

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

Ablation

Method	Sec. 3.1		Sec. 3.2		Sec. 3.4		Acc.	Comp.	Overall↓
	GFN	PVE	FDF	CL	$Loss_{pw}$	$Loss_{dds}$			
baseline (L=4, N=5)					✓		0.3629	0.3016	0.3323
+ geometry fusion network	✓				✓		0.3520	0.2893	0.3207
+ prob. volume embedding		✓			✓		0.3705	0.3053	0.3379
+ fusion & embedding	✓	✓			✓		0.3404	0.2922	0.3163
+ frequency domain filtering	✓		✓		✓		0.3663	0.2707	0.3185
+ curriculum learning	✓		✓	✓	✓		0.3650	0.2634	0.3142
+ distribution similarity loss	✓	✓			✓	✓	0.3346	0.2832	0.3089
proposed	✓	✓	✓	✓	✓	✓	**0.3309**	**0.2593**	**0.2951**

embedding单加是副作用，