DUSt3R Presentation

Algorithm Overview

Introduction

- Not need to know intrinsic and extrinsic parameters
- Single input image
- Based on **pointmaps**
- See the problem as a whole

Pointmap

- Dense 2D field of 3D points
- One-to-one mapping between image pixels and 3D scene points
- Assume no translucent surfaces case

$$X \in \mathbb{R}^{W \times H \times 3}$$

Training Part

Network Architecture

Relative Pose Estimation

- Compute pair-maps
- relative pose $P = [R|t]$
- Sensitive to noise and outliers

$$P, \sigma = \arg\min \sum_{i,j \in \mathcal{C}} w^{i,j} \left\| p^{i,j} - \sigma P \, \tilde{p}^{i,j} \right\|^2, \quad w^{i,j} > 0$$

Global Alignment

Recovering Intrinsics

- Estimate by solve optimization problem
- Assume pixel are squares AND principle point is centered
- Similarly, f2 will be calculate

$$f_1' = \arg\min_{f_1} \sum_{i} \sum_{j \in \mathcal{P}_i} w^{i,j} \left\| p'^{i,j} - f_1 \begin{bmatrix} X^{i,j}_x / X^{i,j}_z \\ Y^{i,j}_y / X^{i,j}_z \end{bmatrix} \right\|$$

Visual Localization

Result was not good

- Never trained for visual localization
- database images never seen during training

Result

Training detail

- Randomly sample an equal number of pairs during each epoch
- Train network sequentially
- Randomly select the image ratio for each batch

Multi-view Pose Estimation

- Select 10 frames
- achieves the best overall performance on the two datasets

Monocular Depth

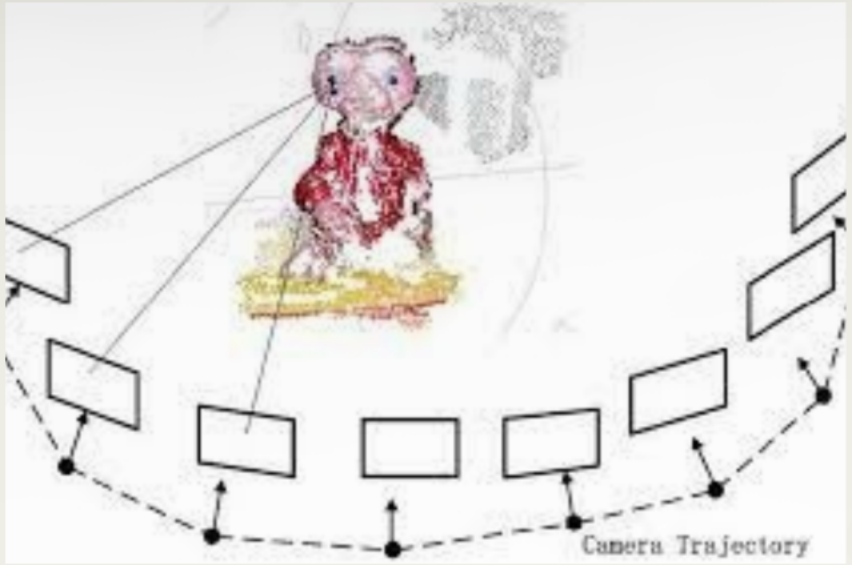
- Feed the same input meta network
- Each predicted is the x coordinate in the predicted 3D pointmap

Conclusion

In summary the DUSt3R algorithm solve not only 3D reconstruction without prior information about scene nor cameras, but a whole variety of 3D vision tasks as well

Introduction

- Not need to know intrinsic and extrinsic parameters
- Single input image
- Based on **pointmaps**
- See the problem as a whole

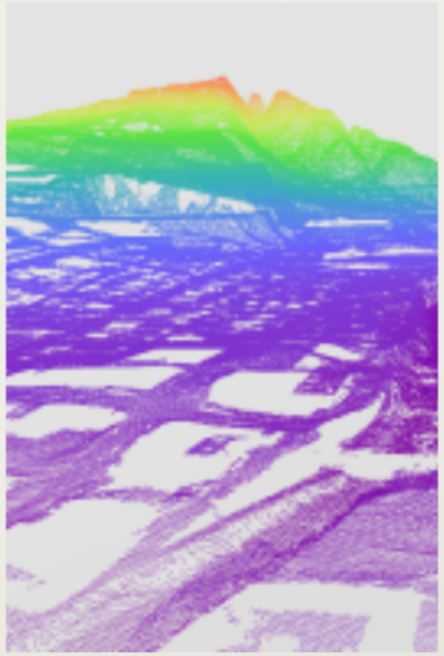


- Not need to know intrinsic and extrinsic parameters
- Single input image
- Based on **pointmaps**
- See the problem as a whole

Pointmap

- Dense 2D field of 3D points
- One-to-one mapping between image pixels and 3D scene points
- Assume no translucent surfaces case

$$X \in \mathbb{R}^{W \times H \times 3}$$



DUSt3R Presentation

Algorithm Overview

Introduction

- Not need to know intrinsic and extrinsic parameters
- Single input image
- Based on **pointmaps**
- See the problem as a whole

Pointmap

- Dense 2D field of 3D points
- One-to-one mapping between image pixels and 3D scene points
- Assume no translucent surfaces case

$$X \in \mathbb{R}^{W \times H \times 3}$$

Training Part

Network Architecture

Relative Pose Estimation

- Compute pair-maps
- relative pose $P = [R|t]$
- Sensitive to noise and outliers

$$P, \sigma = \arg\min \sum_{i,j \in \mathcal{C}} w^{i,j} \left\| p^{i,j} - \sigma P \, \tilde{p}^{i,j} \right\|^2, \quad w^{i,j} > 0$$

Global Alignment

Recovering Intrinsics

- Estimate by solve optimization problem
- Assume pixel are squares AND principle point is centered
- Similarly, f2 will be calculate

$$f_1' = \arg\min_{f_1} \sum_{i} \sum_{j \in \mathcal{P}_i} w^{i,j} \left\| p'^{i,j} - f_1 \begin{bmatrix} X^{i,j}_x / X^{i,j}_z \\ Y^{i,j}_y / X^{i,j}_z \end{bmatrix} \right\|$$

Visual Localization

Result was not good

- Never trained for visual localization
- database images never seen during training

Result

Training detail

- Randomly sample an equal number of pairs during each epoch
- Train network sequentially
- Randomly select the image ratio for each batch

Multi-view Pose Estimation

- Select 10 frames
- achieves the best overall performance on the two datasets

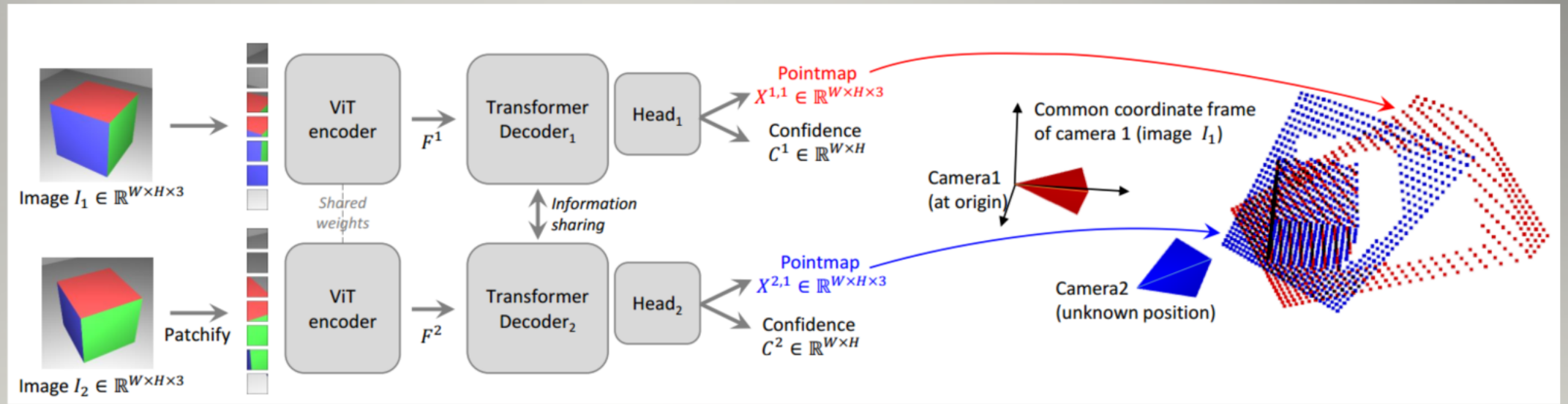
Monocular Depth

- Feed the same input meta network
- Each predicted is the x coordinate in the predicted 3D pointmap

Conclusion

In summary the DUSt3R algorithm solve not only 3D reconstruction without prior information about scene nor cameras, but a whole variety of 3D vision tasks as well

Network Architecture



DUSt3R Presentation

Algorithm Overview

Introduction

- Not need to know intrinsic and extrinsic parameters
- Single input image
- Based on **pointmaps**
- See the problem as a whole

Pointmap

- Dense 2D field of 3D points
- One-to-one mapping between image pixels and 3D scene points
- Assume no translucent surfaces case

$$X \in \mathbb{R}^{W \times H \times 3}$$

Training Part

Network Architecture

Relative Pose Estimation

- Compute pair-maps
- relative pose $P = [R|t]$
- Sensitive to noise and outliers

$$P, \sigma = \arg\min \sum_{i,j \in \mathcal{C}} w^{i,j} \left\| p^{i,j} - \sigma P \, \tilde{p}^{i,j} \right\|^2, \quad w^{i,j} > 0$$

Global Alignment

Recovering Intrinsics

- Estimate by solve optimization problem
- Assume pixel are squares AND principle point is centered
- Similarly, f2 will be calculate

$$f_1' = \arg\min_{f_1} \sum_{i} \sum_{j \in \mathcal{P}_i} w^{i,j} \left\| p'^{i,j} - f_1 \begin{bmatrix} X^{i,j}_x / X^{i,j}_z \\ Y^{i,j}_y / X^{i,j}_z \end{bmatrix} \right\|$$

Visual Localization

Result was not good

- Never trained for visual localization
- database images never seen during training

Result

Training detail

- Randomly sample an equal number of pairs during each epoch
- Train network sequentially
- Randomly select the image ratio for each batch

Multi-view Pose Estimation

- Select 10 frames
- achieves the best overall performance on the two datasets

Monocular Depth

- Feed the same input meta network
- Each predicted is the x coordinate in the predicted 3D pointmap

Conclusion

In summary the DUSt3R algorithm solve not only 3D reconstruction without prior information about scene nor cameras, but a whole variety of 3D vision tasks as well

Training Part

Regression loss:

- Need gound-truth pointmaps
- Normalize the predicted and ground-truth pointmaps for diffrences between datasets

$$\ell_{\text{regr}}(v,i) = \left\| \frac{1}{z} X_i^{v,1} - \frac{1}{\bar{z}} \bar{X}_i^{v,1} \right\|$$

Confidence loss:

- how much confidence network about particular pixel
- force extrapolate in hard area

$$\mathcal{L}_{\text{conf}} = \sum_{v \in \{1,2\}} \sum_{i \in \mathcal{D}^v} C_i^{v,1} \ell_{\text{regr}}(v,i) - \alpha \log C_i^{v,1},$$

Recovering Intrinsics

- Estimate by solve optimization problem
- Assume pixel are squares AND principle point is centered
- Similarly, f2 will be calculate

$$f_1^* = \arg\min_{f_1} \sum_{i=0}^{W} \sum_{j=0}^{H} C_{i,j}^{1,1} \left\| (i', j') - f_1 \frac{(X_{i,j,0}^{1,1}, X_{i,j,1}^{1,1})}{X_{i,j,2}^{1,1}} \right\|$$

Relative Pose Estimation

- Compare pointmaps
- relative pose P = [R |t]
- Sensitive to noise and outliers

$$R^*, t^* = \arg\min_{\sigma, R, t} \sum_i C_i^{1,1} C_i^{1,2} \left\| \sigma(R X_i^{1,1} + t) - X_i^{1,2} \right\|^2$$

Global Alignment

Pairwise graph

- Images are vertices
- Overlap between images are edges

For recover globally aligned pointmaps:

- For each pair of images predict pointmaps and confidence maps
- goal is rotate pairwise prediction in common cordinate frame
- Rigid transformation P align pointmaps with world coordinate pointmaps

Training detail

- Randomly sample an equal number of pairs during each epoch
- Train network sequentially
- Randomly select the image ratios for each batch

Result

Visual Localization

Result was not good

- Never trained for visual localisation
- database images were seen during training.

	Methods	7Scenes (Indoor) [113]							Cambridge (Outdoor) [48]				
		Chess	Fire	Heads	Office	Pumpkin	Kitchen	Stairs	S. Facade	O. Hospital	K. College	St.Mary's	G. Court
FM	AS [102]	4/1.96	3/1.53	2/1.45	9/3.61	8/3.10	7/3.37	**3**/2.22	**4**/0.21	20/0.36	13/0.22	8/0.25	24/**0.13**
FM	HLoc [100]	**2/0.79**	**2/0.87**	**2/0.92**	**3/0.91**	**5/1.12**	**4/1.25**	6/**1.62**	**4/0.2**	**15/0.3**	**12/0.20**	**7/0.21**	**11**/0.16
E2E	DSAC* [11]	**2**/1.10	**2**/1.24	**1**/1.82	**3**/1.15	**4**/1.34	4/1.68	**3**/1.16	**5**/0.3	**15/0.3**	15/0.3	13/0.4	49/0.3
E2E	HSCNet [54]	**2/0.7**	**2**/0.9	**1**/0.9	**3/0.8**	**4/1.0**	4/**1.2**	**3/0.8**	6/0.3	19/**0.3**	18/0.3	9/0.3	28/0.2
E2E	PixLoc [101]	**2**/0/80	**2/0.73**	**1/0.82**	**3**/0.82	**4**/1.21	**3/1.20**	5/1.30	**5/0.23**	16/0.32	14/0.24	10/0.34	30/0.14
E2E	SC-wLS [151]	3/0.76	5/1.09	3/1.92	6/0.86	8/1.27	9/1.43	12/2.80	11/0.7	42/1.7	14/0.6	39/1.3	164/0.9
E2E	NeuMaps [124]	**2**/0.81	3/1.11	2/1.17	**3**/0.98	**4**/1.11	4/1.33	4/1.12	6/0.25	19/0.36	14/0.19	17/0.53	**6/ 0.10**
	DUSt3R 224-NoCroCo	5/1.76	6/2.02	3/1.75	5/1.54	9/2.35	6/1.82	34/7.81	24/1.33	79/1.17	69/1.15	46/1.51	143/1.32
	DUSt3R 224	3/0.96	3/1.02	**1**/1.00	4/1.04	5/1.26	4/1.36	21/4.08	9/0.38	26/0.46	20/0.32	11/0.38	36/0.24
	DUSt3R 512	3/0.97	3/0.95	2/1.37	**3**/1.01	**4**/1.14	4/1.34	11/2.84	6/0.26	17/0.33	**11/0.20**	**7/0.24**	38/0.16

DUSt3R Presentation

Algorithm Overview

Introduction

- Not need to know intrinsic and extrinsic parameters
- Single input image
- Based on **pointmaps**
- See the problem as a whole

Pointmap

- Dense 2D field of 3D points
- One-to-one mapping between image pixels and 3D scene points
- Assume no translucent surfaces case

$$X \in \mathbb{R}^{W \times H \times 3}$$

Training Part

Network Architecture

Relative Pose Estimation

- Compute pair-maps
- relative pose $P = [R|t]$
- Sensitive to noise and outliers

$$P, \sigma = \arg\min \sum_{i,j \in \mathcal{C}} w^{i,j} \left\| p^{i,j} - \sigma P \, \tilde{p}^{i,j} \right\|^2, \quad w^{i,j} > 0$$

Global Alignment

Recovering Intrinsics

- Estimate by solve optimization problem
- Assume pixel are squares AND principle point is centered
- Similarly, f2 will be calculate

$$f_1' = \arg\min_{f_1} \sum_{i} \sum_{j \in \mathcal{P}_i} w^{i,j} \left\| p'^{i,j} - f_1 \begin{bmatrix} X^{i,j}_x / X^{i,j}_z \\ Y^{i,j}_y / X^{i,j}_z \end{bmatrix} \right\|$$

Visual Localization

Result was not good

- Never trained for visual localization
- database images never seen during training

Result

Training detail

- Randomly sample an equal number of pairs during each epoch
- Train network sequentially
- Randomly select the image ratio for each batch

Multi-view Pose Estimation

- Select 10 frames
- achieves the best overall performance on the two datasets

Monocular Depth

- Feed the same input meta network
- Each predicted is the x coordinate in the predicted 3D pointmap

Conclusion

In summary the DUSt3R algorithm solve not only 3D reconstruction without prior information about scene nor cameras, but a whole variety of 3D vision tasks as well

Multi-view Pose Estimation

- Select 10 frames
- achieves the best overall performance on the two datasets

Methods	Co3Dv2 [93]			RealEstate10K
	RRA@15	RTA@15	mAA(30)	mAA(30)
RelPose [176]	57.1	-	-	-
Colmap+SPSG [26, 99]	36.1	27.3	25.3	45.2
PixSfM [58]	33.7	32.9	30.1	49.4
PosReg [139]	53.2	49.1	45.0	-
PoseDiffusion [139]	80.5	79.8	66.5	48.0
DUSt3R 512 (w/ PnP)	94.3	**88.4**	**77.2**	61.2
DUSt3R 512 (w/ GA)	**96.2**	86.8	76.7	**67.7**

Monocular Depth

- Feed the same input to the network
- Depth prediction is the z coordinate in the predicted 3D pointmap

Methods	Train	Outdoor				Indoor					
		DDAD[40]		KITTI [35]		BONN [79]		NYUD-v2 [114]		TUM [118]	
		Rel↓	$\delta_{1.25}$ ↑	Rel↓	$\delta_{1.25}$ ↑	Rel↓	$\delta_{1.25}$ ↑	Rel↓	$\delta_{1.25}$ ↑	Rel ↓	$\delta_{1.25}$ ↑
DPT-BEiT[90]	D	10.70	**84.63**	9.45	89.27	-	-	**5.40**	**96.54**	**10.45**	**89.68**
NeWCRFs[173]	D	**9.59**	82.92	**5.43**	**91.54**	-	-	6.22	95.58	14.63	82.95
Monodepth2 [37]	SS	23.91	75.22	11.42	86.90	56.49	35.18	16.19	74.50	31.20	47.42
SC-SfM-Learners [6]	SS	16.92	77.28	11.83	86.61	21.11	71.40	13.79	79.57	22.29	64.30
SC-DepthV3 [120]	SS	**14.20**	**81.27**	11.79	86.39	**12.58**	**88.92**	**12.34**	**84.80**	**16.28**	**79.67**
MonoViT[181]	SS	-	-	**09.92**	**90.01**	-	-	-	-	-	
RobustMIX [91]	T	-	-	18.25	76.95	-	-	11.77	90.45	15.65	**86.59**
SlowTv [116]	T	**12.63**	79.34	(6.84)	(56.17)	-	-	11.59	87.23	15.02	80.86
DUSt3R 224-NoCroCo	T	19.63	70.03	20.10	71.21	14.44	86.00	14.51	81.06	22.14	66.26
DUSt3R 224	T	16.32	77.58	16.97	77.89	11.05	89.95	10.28	88.92	17.61	75.44
DUSt3R 512	T	13.88	81.17	**10.74**	**86.60**	**8.08**	**93.56**	**6.50**	94.09	**14.17**	79.89

AbsRel = $|y - \hat{y}|/y$

δ 1.25 = $\max(\hat{y}/y, y/\hat{y})$

3D Reconstruction

- It was not as accurate as the others

	Methods	GT cams	Acc.↓	Comp.↓	Overall↓
(a)	Camp [12]	✓	0.835	0.554	0.695
	Furu [33]	✓	0.613	0.941	0.777
	Tola [133]	✓	0.342	1.190	0.766
	Gipuma [34]	✓	**0.283**	0.873	0.578
(b)	MVSNet [160]	✓	0.396	0.527	0.462
	CVP-MVSNet [157]	✓	0.296	0.406	0.351
	UCS-Net [18]	✓	0.338	0.349	0.344
	CER-MVS [64]	✓	0.359	0.305	0.332
	CIDER [156]	✓	0.417	0.437	0.427
	PatchmatchNet [138]	✓	0.427	0.277	0.352
	GeoMVSNet [179]	✓	0.331	**0.259**	**0.295**
	DUSt3R 512	×	2.677	0.805	1.741

Conclusion

In summary, the DUSt3R algorithm solve not only 3D reconstruction without prior information about scene nor cameras, but a whole variety of 3D vision tasks as well.