DualRefine: Self-Supervised Depth and Pose Estimation Through Iterative Epipolar Sampling and Refinement Toward Equilibrium

Antyanta Bangunharcana[1], Ahmed Magd[2], Kyung-Soo Kim[1]

[1]Mechatronics, Systems, and Control Laboratory, [2]Vehicular Systems Design and Control Lab

Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea

The IEEE/CVF Conference on Computer Vision and Pattern Recognition 2023

Mechatronics, Systems and Control

KAIST

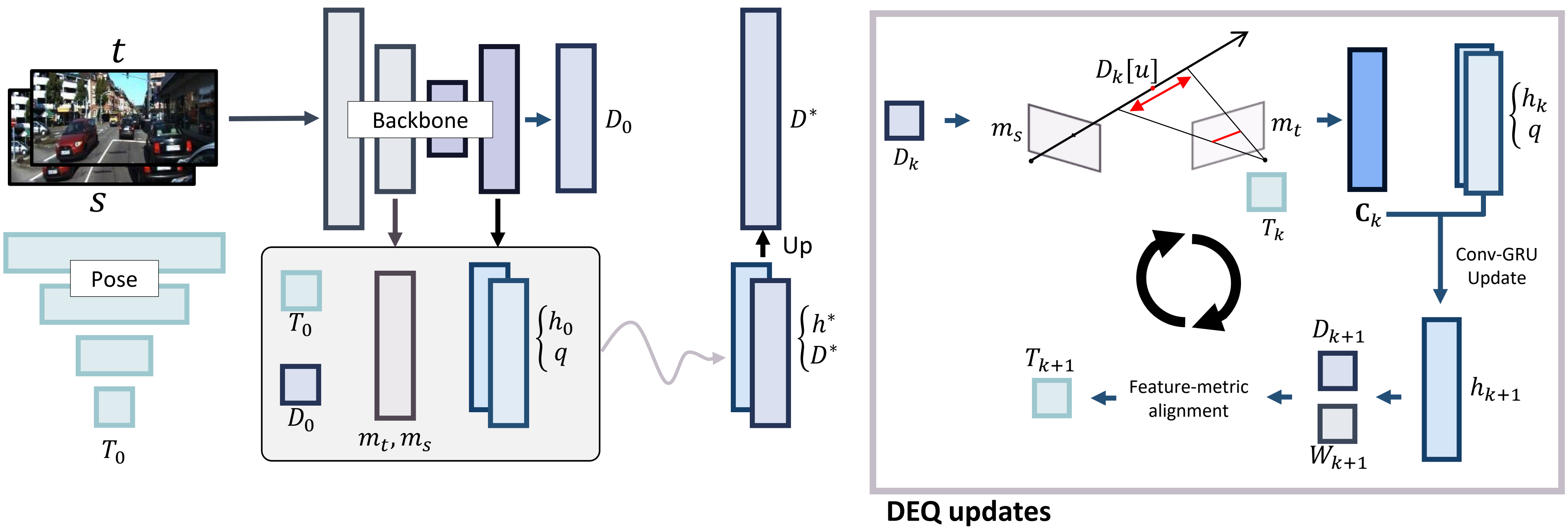
KAIST VDC LAB Vehicular Systems Design and Control Lab

JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

Overview

❖ DualRefine

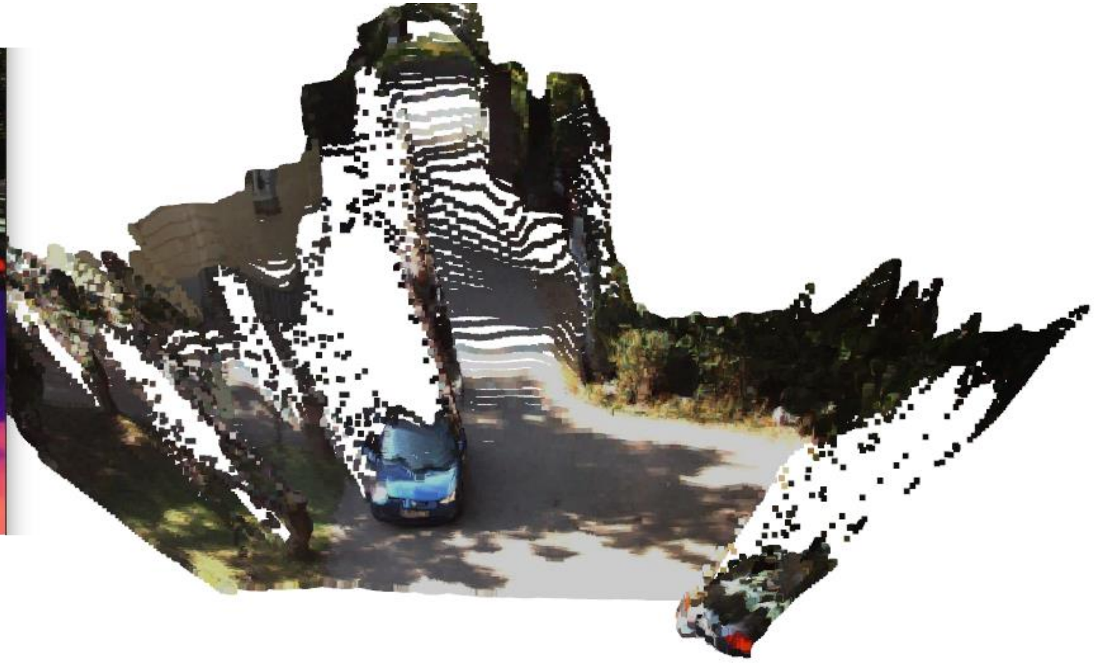
- Train a network that refines *both* the depth and pose estimates toward an equilibrium



Introduction : Background and Motivation

❖ Background

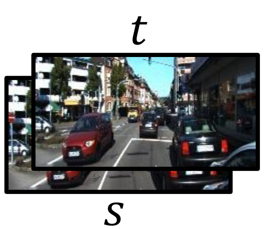
- Depth and pose estimation for robotics, autonomous driving, and AR/VR

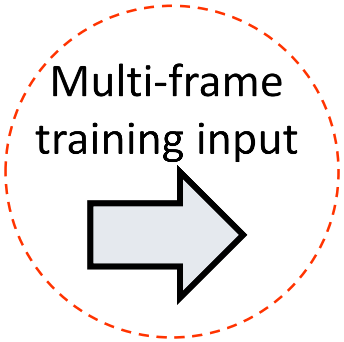


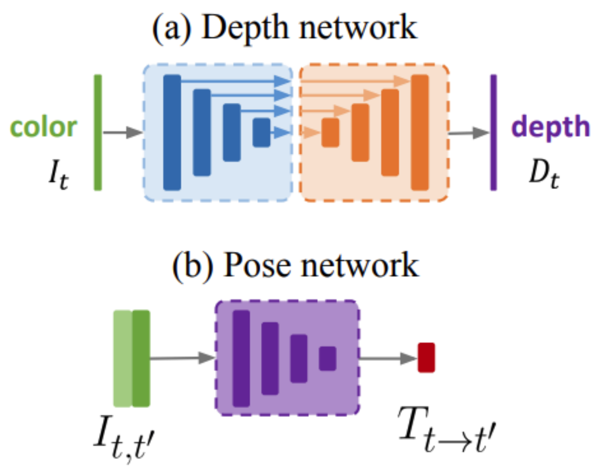
Introduction : Background and Motivation

❖ Background

- Self-supervised depth and pose estimation







$$z'u' = z'\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = KT_{t\to s}\left(D[u]K^{-1}\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}\right)$$

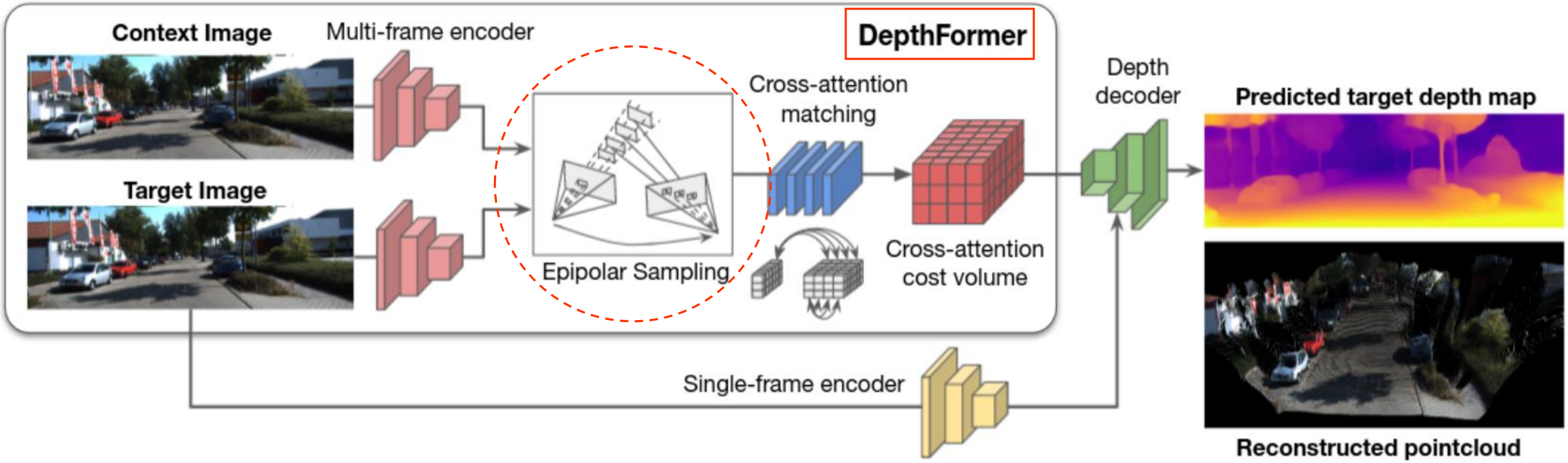
Warp & training loss

MonoDepth2

Godard, Clément, et al. "Digging into self-supervised monocular depth esti mation." *Proceedings of the IEEE/CVF International Conference on Compu ter Vision*. 2019.

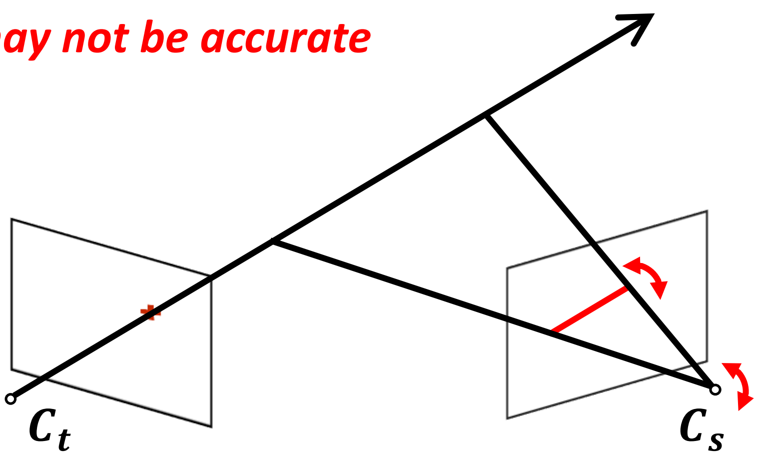
Introduction : Background and Motivation

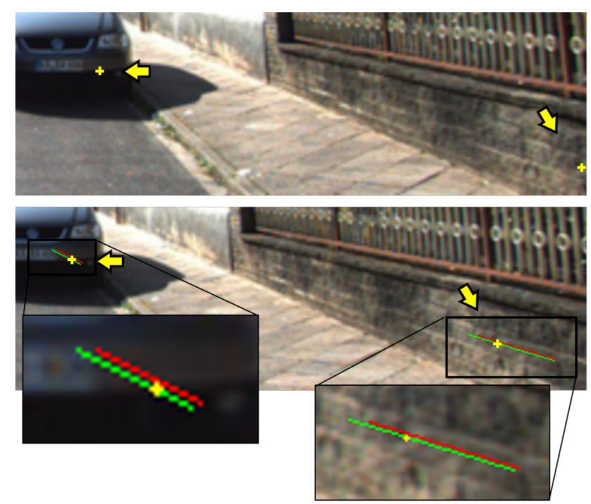
❖ DepthFormer



Guizilini, Vitor, et al. "Multi-frame self-supervised depth with transformers." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022.

Poses may not be accurate

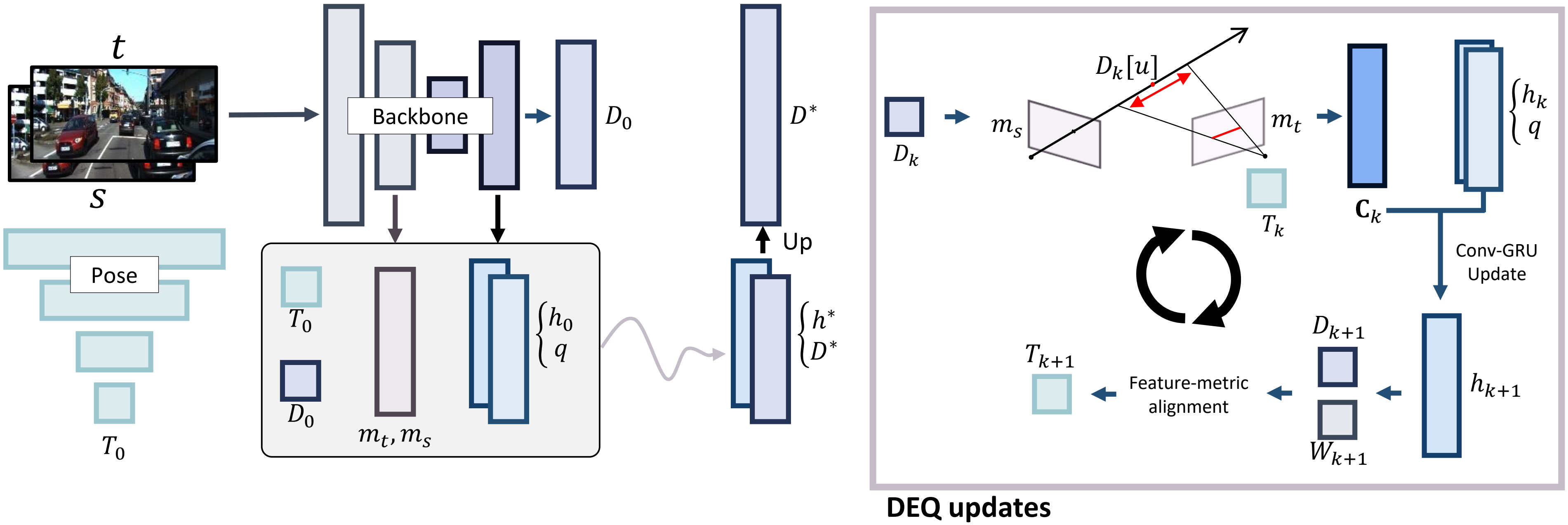




Methodology : Overview

❖ DualRefine

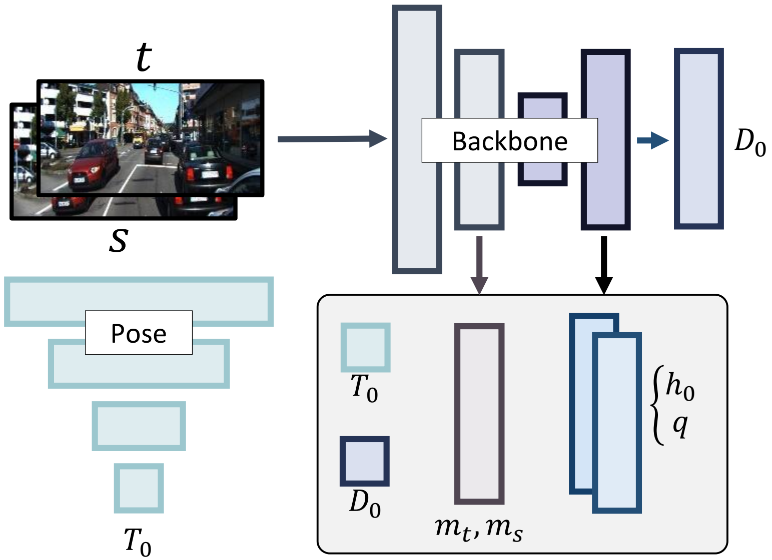
- Train a network that refines *both* the depth and pose estimates toward an equilibrium



Methodology : DualRefine

❖ Initial estimates

- Estimate initial D_0 and T_0 using baseline models (*e.g.*, MonoDepth2)
  - We use DIFFNet[†] for its SoTA accuracy



The initial estimates serve two purposes:

1. Initial states for the refinement steps
2. Teacher constraint for the refined estimates[‡]

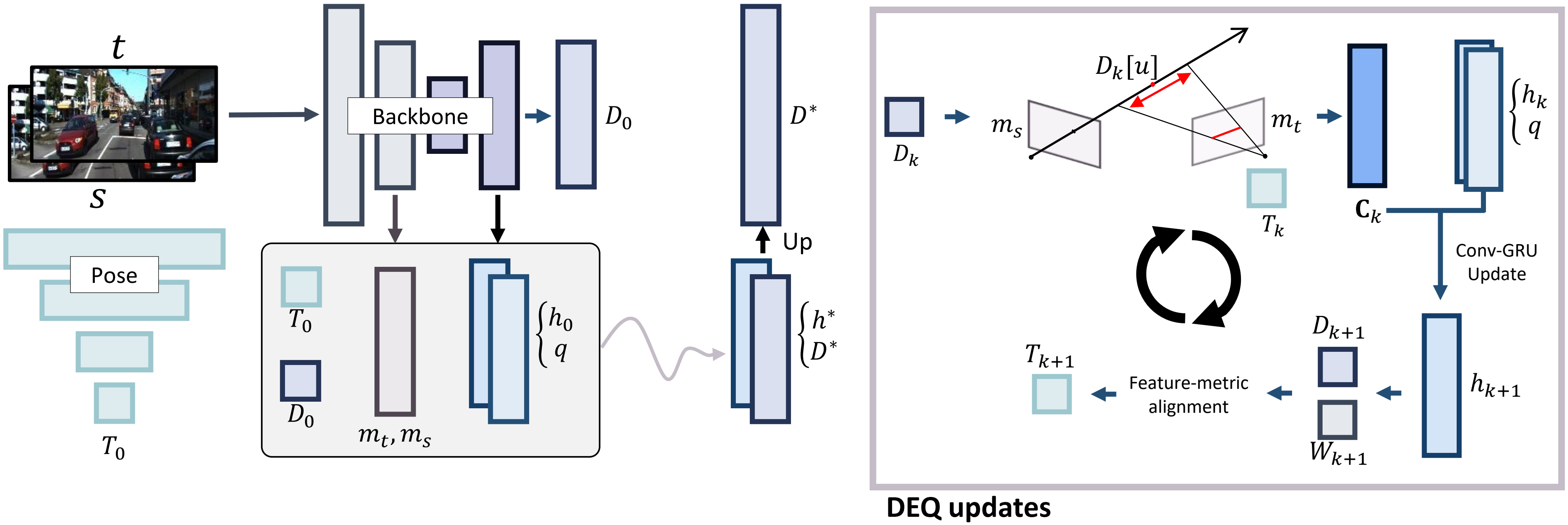
- Extract hidden feature maps

[†]Zhou, Hang, David Greenwood, and Sarah Taylor. "Self-supervised monocular depth estimation with internal feature fusion." *arXiv preprint arXiv:2110.09482* (2021).
[‡]Watson, Jamie, et al. "The temporal opportunist: Self-supervised multi-frame monocular depth." *Proceedings of the IEEE/CVF Conference on Computer Vision and Patt ern Recognition*. 2021.

Methodology : DualRefine

❖ Refinements

- Perform refinement updates towards equilibrium depth and pose
  This is implemented as a DEQ → efficient training memory



Methodology : DualRefine

❖ Refinements

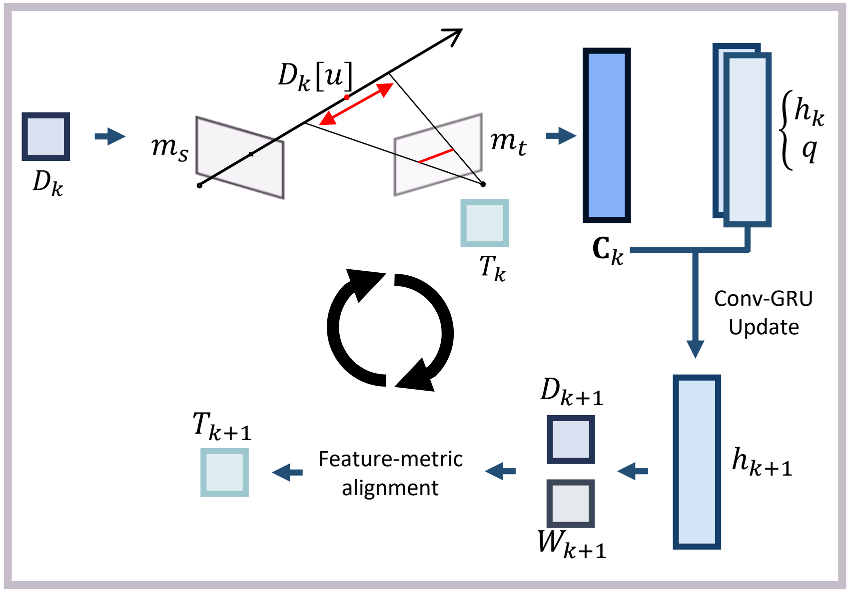
- Perform refinement updates towards equilibrium depth and pose

▪ Update at step k

Input: D_k and T_k (h_k)

Output: D_{k+1} and T_{k+1} (h_{k+1})

1. Sample matching costs C_k locally
2. Conv-GRU to update the hidden states
3. Compute depth updates ΔD_k and per-pixel matching confidence
4. Compute updated pose T_{k+1}



DEQ updates

Methodology : DualRefine

❖ Refinements

- Perform refinement updates towards equilibrium depth and pose

Monocular depth and relative poses

❖ Monocular estimates

- KITTI dataset
- Losses based on Monodepth2 and Manydepth
  with modifications to train both depth and pose refinements

	Method	Test frames	Semantics	$W \times H$	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	δ_1 ↑	δ_2 ↑	δ_3 ↑
Low & mid res	Ranjan *et al.* [73]	1		832 × 256	0.148	1.149	5.464	0.226	0.815	0.935	0.973
	EPC++ [62]	1		832 × 256	0.141	1.029	5.350	0.216	0.816	0.941	0.976
	Struct2depth (M) [11]	1	•	416 × 128	0.141	1.026	5.291	0.215	0.816	0.945	0.979
	Videos in the wild [29]	1	•	416 × 128	0.128	0.959	5.230	0.212	0.845	0.947	0.976
	Guizilini *et al.* [33]	1	•	640 × 192	0.102	0.698	4.381	0.178	0.896	0.964	**0.984**
	Johnston *et al.* [45]	1		640 × 192	0.106	0.861	4.699	0.185	0.889	0.962	0.982
	Monodepth2 [26]	1		640 × 192	0.115	0.903	4.863	0.193	0.877	0.959	0.981
	Packnet-SFM [31]	1		640 × 192	0.111	0.785	4.601	0.189	0.878	0.960	0.982
	Li *et al.* [54]	1		416 × 128	0.130	0.950	5.138	0.209	0.843	0.948	0.978
	DIFFNet [109]	1		640 × 192	0.102	0.764	4.483	0.180	0.896	0.965	0.983
	DualRefine-initial (D_0)	1		640 × 192	0.103	0.721	4.476	0.180	0.891	0.965	**0.984**
	Patil *et al.* [70]	N†		640 × 192	0.111	0.821	4.650	0.187	0.883	0.961	0.982
	Wang *et al.* [93]	2 (-1, 0)		640 × 192	0.106	0.799	4.662	0.187	0.889	0.961	0.982
	ManyDepth (MR) [95]	2 (-1, 0)	2021	640 × 192	0.098	0.770	4.459	0.176	0.900	0.965	0.983
	DepthFormer [32]	2 (-1, 0)	2022	640 × 192	0.090	**0.661**	**4.149**	0.175	0.905	**0.967**	**0.984**
	DualRefine-refined (D^*)	2 (-1, 0)		640 × 192	**0.087**	0.698	4.234	**0.170**	**0.914**	**0.967**	0.983
High res	DRO [30]	2 (-1, 0)		960 × 320	0.088	0.797	4.464	0.212	0.899	0.959	0.980
	Wang *et al.* [93]	2 (-1, 0)		1024 × 320	0.106	0.773	4.491	0.185	0.890	0.962	0.982
	ManyDepth (HR ResNet50) [95]	2 (-1, 0)		1024 × 320	0.091	0.694	4.245	0.171	0.911	0.968	0.983
	DualRefine-refined (HR) (D^*)	2 (-1, 0)		960 × 288	**0.087**	**0.674**	**4.130**	**0.167**	**0.915**	**0.969**	**0.984**

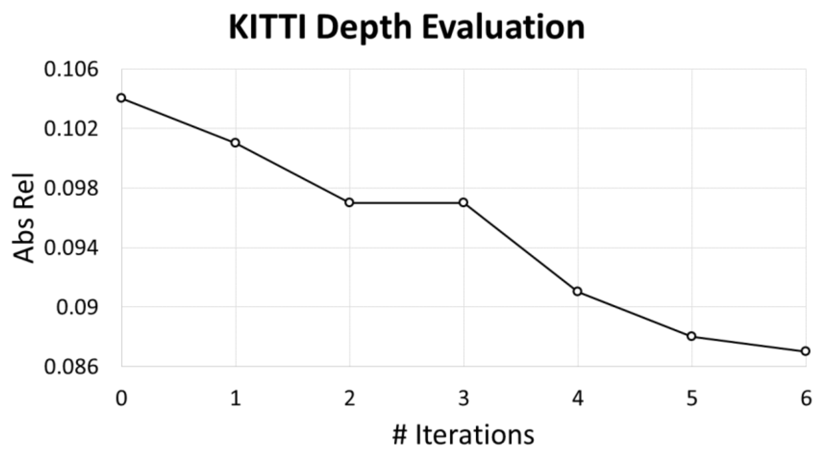
- The refinement procedure massively improves the initial estimates
- Competitive with SoTA DepthFormer that is based on heavy Transformer-based architecture, while requiring less than 1/8 their memory (2GB vs 16GB per batch)
- The current proposed model can run between 15~25 fps

Monocular depth and relative poses

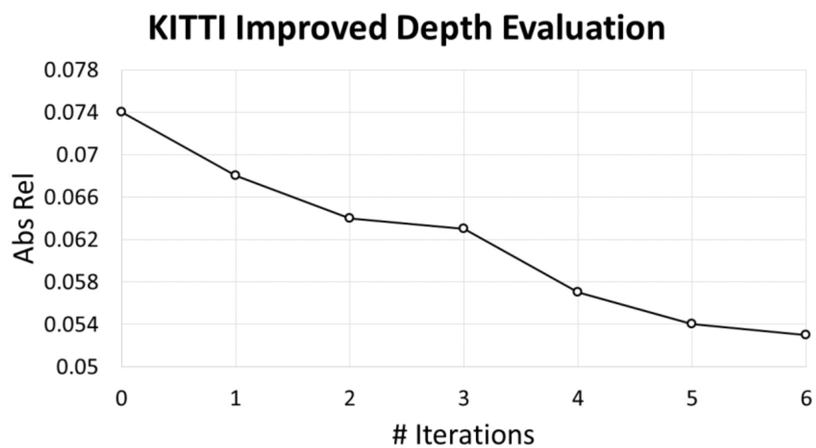
❖ Accuracy/error vs num iterations

- KITTI depth and *improved* depth data
- Abs Rel indicates the absolute relative error to ground truth

# iters	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	δ_1 ↑	δ_2 ↑	δ_3 ↑
0	0.104	0.778	4.495	0.181	0.894	0.965	0.983
1	0.101	0.743	4.405	0.179	0.902	0.966	0.983
2	0.097	0.708	4.302	0.176	0.909	0.967	0.983
3	0.097	0.711	4.312	0.176	0.908	0.967	0.983
4	0.091	0.700	4.259	0.172	0.913	0.967	0.983
5	0.088	0.697	4.239	0.170	0.914	0.967	0.983
6	0.087	0.698	4.234	0.170	0.914	0.967	0.983
7	0.088	0.696	4.230	0.171	0.913	0.967	0.983
8	0.088	0.695	4.229	0.172	0.912	0.966	0.983
9	0.089	0.693	4.234	0.173	0.911	0.966	0.983



# iters	Abs Rel ↓	Sq Rel ↓	RMSE ↓	RMSE log ↓	δ_1 ↑	δ_2 ↑	δ_3 ↑
0	0.074	0.389	3.390	0.115	0.940	0.990	0.997
1	0.068	0.344	3.201	0.106	0.950	0.991	0.997
2	0.064	0.311	3.100	0.101	0.956	0.992	0.998
3	0.063	0.314	3.105	0.101	0.956	0.992	0.998
4	0.057	0.299	3.029	0.096	0.960	0.992	0.998
5	0.054	0.293	2.995	0.093	0.961	0.992	0.998
6	0.053	0.290	2.974	0.092	0.962	0.992	0.998
7	0.052	0.287	2.962	0.092	0.962	0.992	0.998
8	0.053	0.285	2.963	0.093	0.961	0.992	0.998
9	0.054	0.286	2.979	0.094	0.960	0.992	0.998



Monocular depth and relative poses

❖ Ablation

- Importance of pose updates
- What effects the pixel weighting for the pose updates towards the depth accuracy
- The impact of pose refinement for computing accurate consistency masks

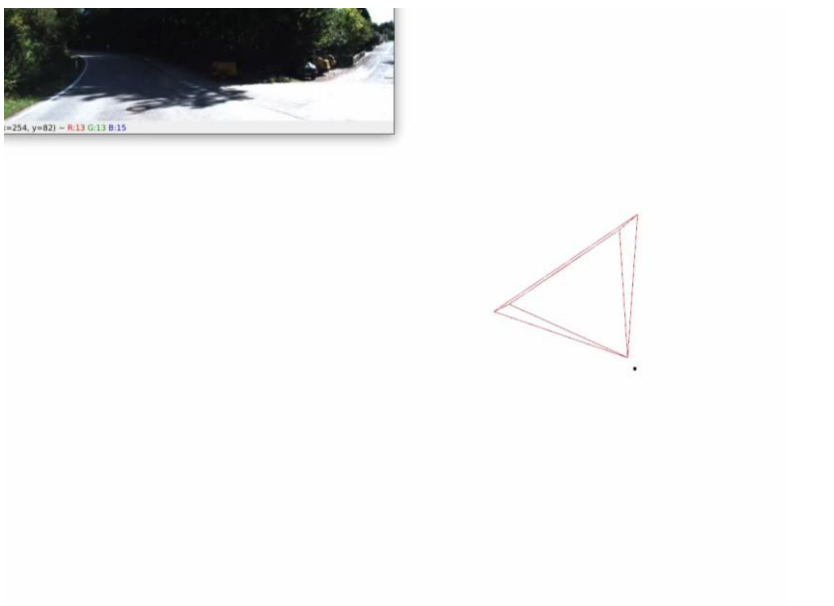
Pose Updates	Consistency mask	Abs Rel	Sq Rel	RMSE	δ_1	δ_2
no update	T_0	0.097	0.713	4.462	0.898	0.964
no weights	T_0	0.091	0.694	4.271	0.909	**0.967**
no $W_{h,k}$	T_0	0.090	0.667	4.252	0.909	**0.967**
no W_q	T_0	0.093	0.686	4.258	0.908	**0.967**
W_q and $W_{h,k}$	T_0	0.090	0.669	4.293	0.910	**0.967**
no weights	T^*	0.092	0.667	4.257	0.908	**0.967**
no $W_{h,k}$	T^*	0.091	**0.666**	4.243	0.909	**0.967**
no W_q	T^*	0.088	0.674	4.251	0.911	0.966
W_q and $W_{h,k}$	T^*	**0.087**	0.698	**4.234**	**0.914**	**0.967**

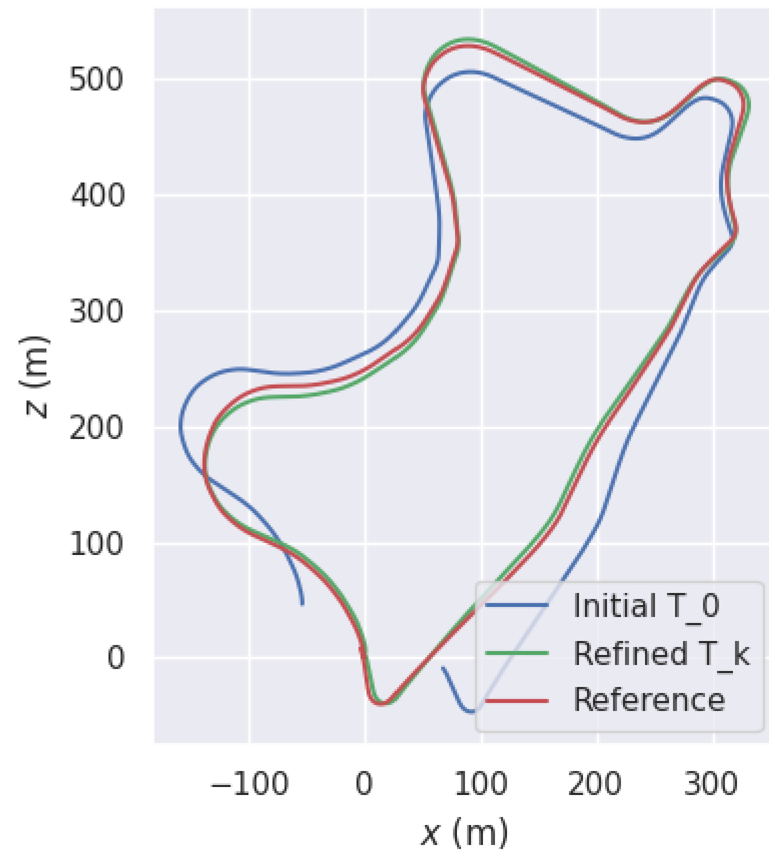
Table 2. Ablation experiment for the effect of pose updates

Monocular depth and relative poses

❖ Monocular estimates

- KITTI Visual odometry KITTI sequence 9





Monocular depth and relative poses

❖ Quantitative results

- KITTI Visual odometry Sequence 9 and 10 (common evaluation sequence)

	Methods	Seq 9			Seq 10		
		$t_{err}(\%)\downarrow$	$r_{err}(^\circ/100m)\downarrow$	ATE $(m)\downarrow$	$t_{err}(\%)\downarrow$	$r_{err}(^\circ/100m)\downarrow$	ATE $(m)\downarrow$
	ORB-SLAM2 [68] (w/o LC)	9.67	0.3	44.10	4.04	0.3	6.43
2017	ORB-SLAM2 [68]	3.22	0.4	8.84	4.25	0.3	8.51
	SfMLearner [110]	19.15	6.82	77.79	40.40	17.69	67.34
	GeoNet [102]	28.72	9.8	158.45	23.90	9.0	43.04
	DeepMatchVO [76]	9.91	3.8	27.08	12.18	5.9	24.44
	Monodepth2 [26]	17.17	3.85	76.22	11.68	5.31	20.35
	DW [29]-Learned	-	-	20.91	-	-	17.88
	DW [29]-Corrected	-	-	19.01	-	-	14.85
	SC-Depth [8]	7.31	3.05	23.56	7.79	4.90	12.00
2020	Zou *et al.* [111]	3.49	**1.00**	11.30	**5.81**	1.8	11.80
2021	P-RGBD SLAM [9]	5.08	1.05	13.40	4.32	2.34	**7.99**
	DualRefine-initial (T_0)	9.06	2.59	39.31	9.45	4.05	15.13
	DualRefine-refined (T^*)	**3.43**	1.04	**5.18**	6.80	**1.13**	10.85

- Competitive odometry accuracy to SoTA
  - Currently the proposed method only utilizes frames at $t-1$ and t,
  - while ORB-SLAM2 (full) performs local bundle adjustments and global loop closure optimization.
  - Zou et al. perform training with global optimization,
  - and P-RGBD SLAM also integrates global loop closure optimization

Conclusion

❖ What's next?

- Current limitations:
  - Dynamic objects, non-Lambertian surfaces due to higher reliance on geometry

❖ Further details

SCAN ME

Github Code

https://github.com/antabangun/DualRefine

SCAN ME

Project Page

https://antabangun.github.io/projects/DualRefine/index.html