Benchmarking MDE: The Design Decisions that Matter

Jaime Spencer
@jaimespencer06 @depthchallenge

CVSSP | Centre for Vision, Speech and Signal Processing

– Recent efforts in benchmarking monocular depth estimation

– **Failure points**

- Incorrect ground-truth
- Uninformative metrics
- Ablating/hyperparam tuning on test set
- Non-comparable methodologies

Deconstructing Monocular Depth Reconstruction: The Design Decisions that Matter, Spencer et al, arXiv22

Code

Paper

MDEC @ WACV23

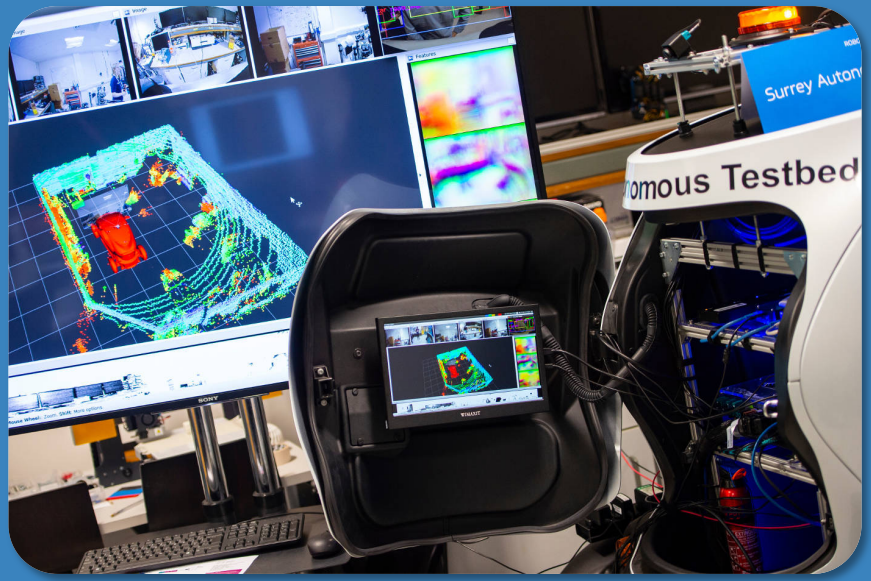
– Benchmarking is a **core component** of every research field

IS

- **Proxy** for real-world performance
- **Introspective**
- **Comparison** between methods

ISN'T

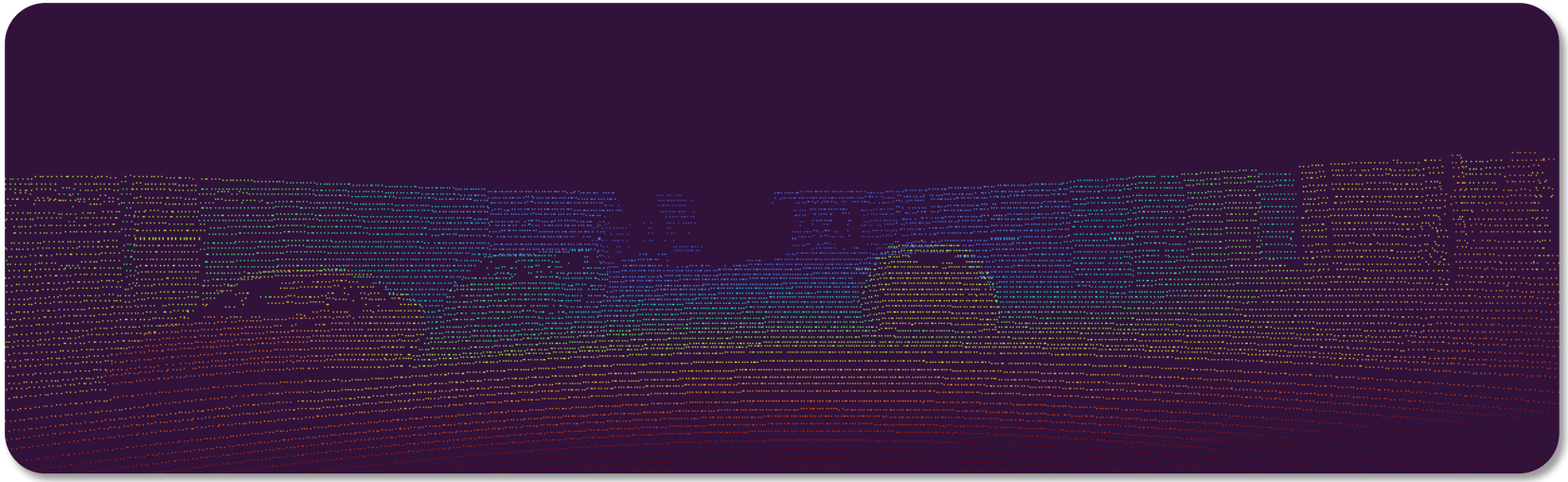
- Real-world performance
- Every metric ever invented
- Holy grail for accepting papers



Ground-truth

– Depth estimation on **Kitti** uses vehicle **LiDAR**

- Viewpoints not perfectly aligned ⟶ **Different occlusions**
- Combined with LiDAR sparsity ⟶ **Inaccurate at boundaries**

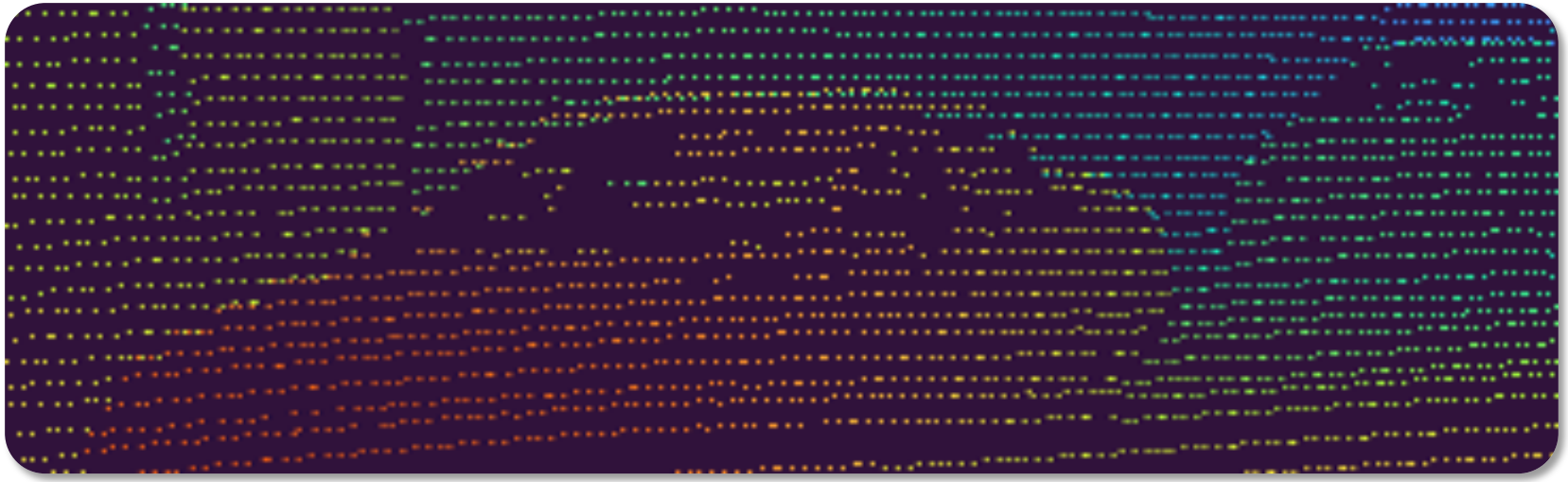


– Depth estimation on **Kitti** uses vehicle **LiDAR**

- Viewpoints not perfectly aligned ⟶ **Different occlusions**
- Combined with LiDAR sparsity ⟶ **Inaccurate at boundaries**

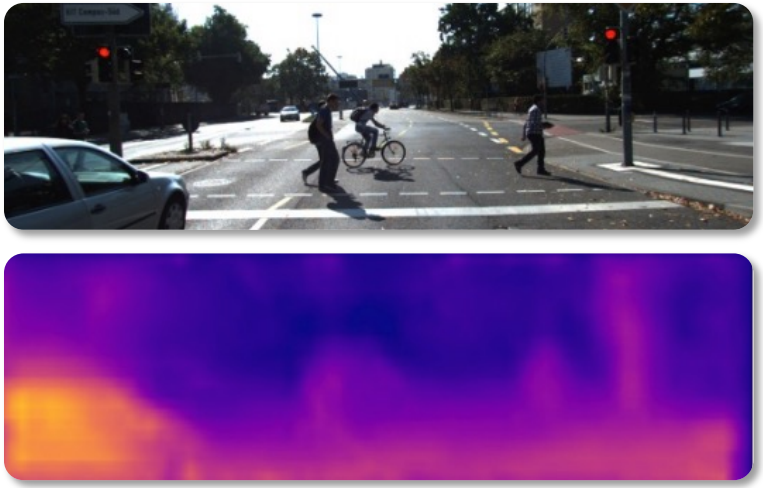
– Depth estimation on **Kitti** uses vehicle **LiDAR**

- Viewpoints not perfectly aligned ⟶ **Different occlusions**
- Combined with LiDAR sparsity ⟶ **Inaccurate at boundaries**



- Does it mean this benchmark was useless? **No!**
- As **research progresses**, the detail required changes

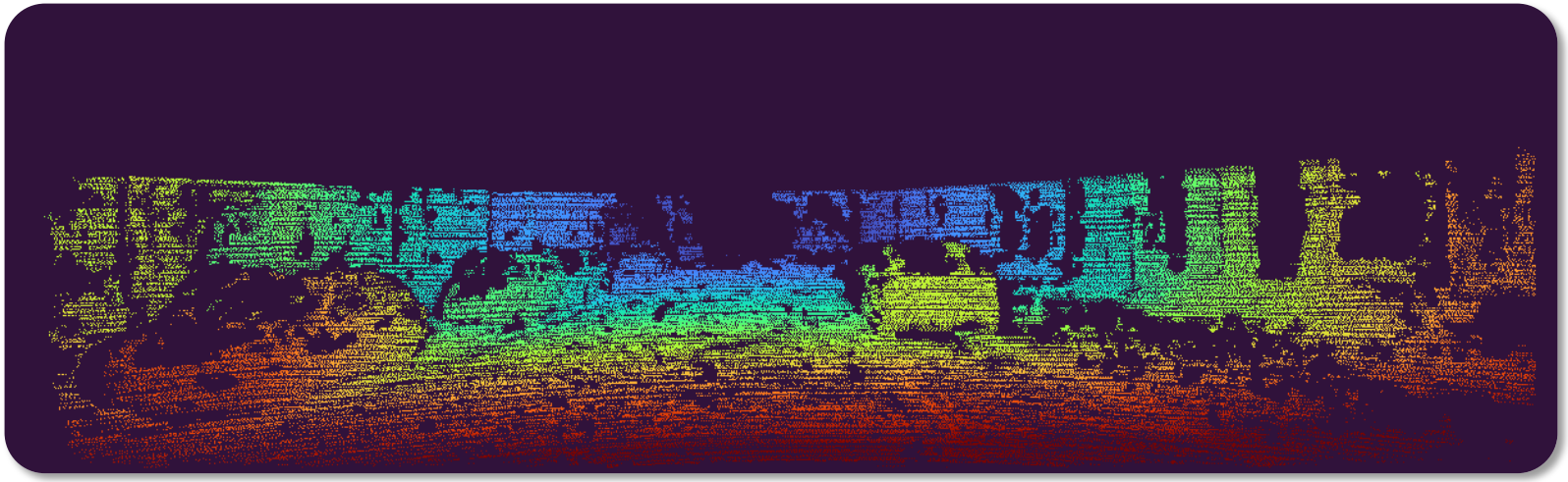
Eigen & Fergus, 2014



Garg et al, 2016

– **Accumulate LiDAR** over multiple frames + **consistency checks**

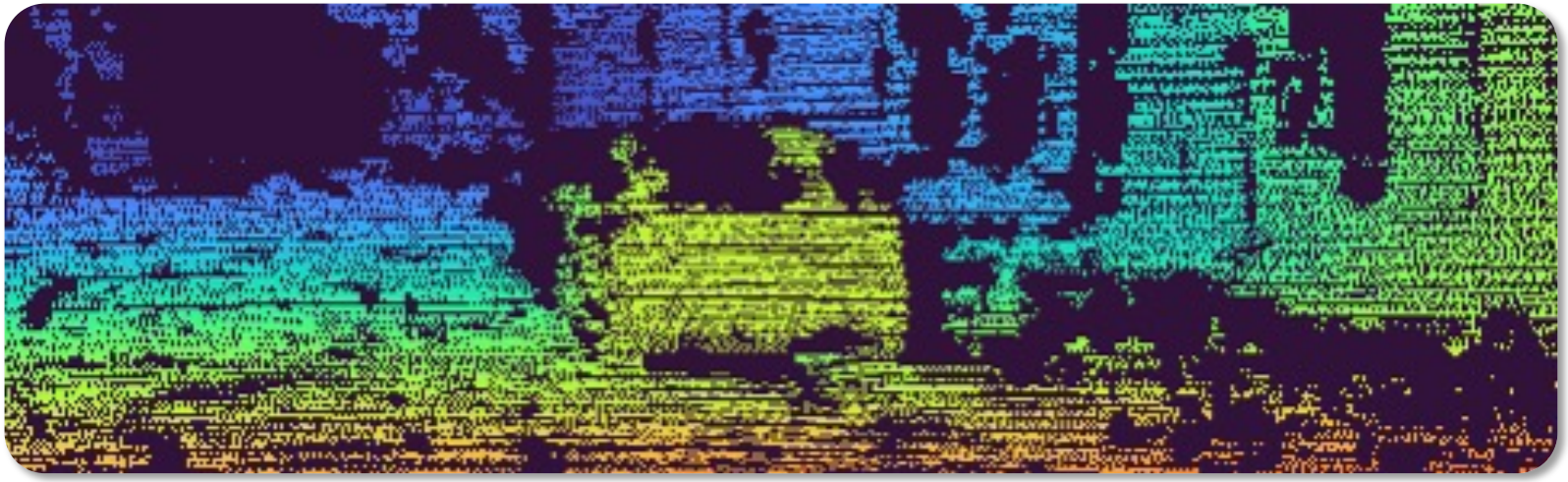
➢ Removes points close to **object boundaries**



Sparsity Invariant CNNs, Uhrig et al, 3DV17

– **Accumulate LiDAR** over multiple frames + **consistency checks**

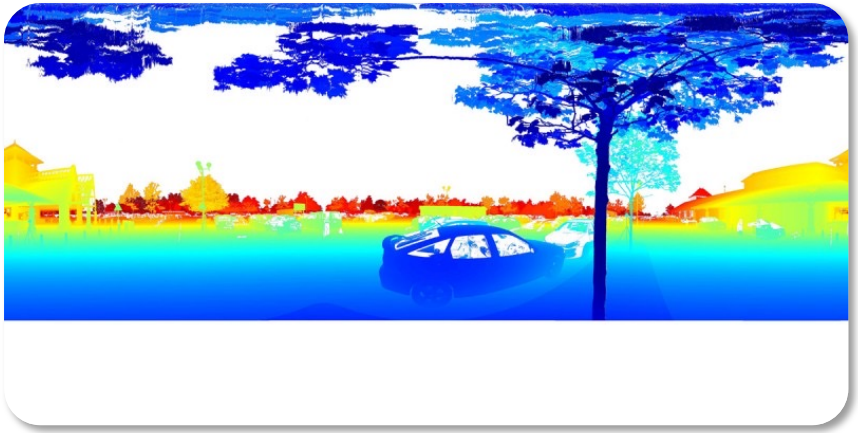
➢ Removes points close to **object boundaries**



Sparsity Invariant CNNs, Uhrig et al, 3DV17

– **SYNS** consists of aligned image/LiDAR panoramas

➢ **Outdoor & dense!** But highly sensitive to **dynamic objects**



The Southampton-York Natural Scenes (SYNS) dataset: Statistics of surface attitude, Adams et al, Scientific Reports 16

– Generate **SYNS-Patches** by sampling every 20° at eye level

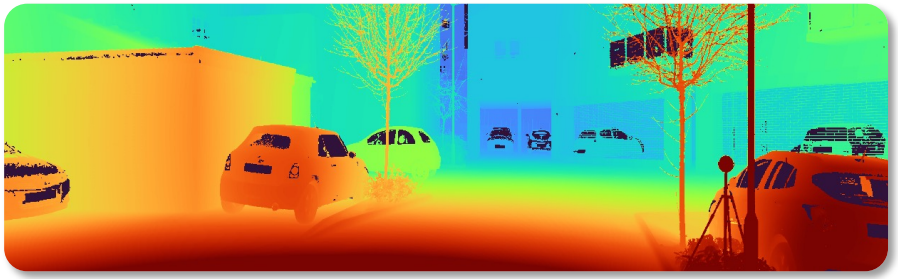
➢ **Manual filtering** of dynamic objects and inconsistencies

– Generate **SYNS-Patches** by sampling every 20° at eye level

➢ **Manual filtering** of dynamic objects and inconsistencies

– Generate **SYNS-Patches** by sampling every 20° at eye level

➢ Dense ground-truth allows for accurate **depth boundaries**



Metrics

– A few Kitti Eigen metrics are **saturated**, **incorrect** or **not informative**

Squared Relative difference: $\frac{1}{|T|}\sum_{y\in T}||y-y^*||^2/y^*$

higher is better		
δ_1	δ_2	δ_3
0.879	0.961	0.982
0.887	0.964	0.983
0.884	0.965	**0.984**
0.892	0.966	**0.984**
0.893	0.965	**0.984**
0.888	0.965	**0.984**
0.898	0.966	**0.984**
0.907	0.967	**0.984**
0.908	**0.968**	**0.984**
0.895	0.964	0.982
0.891	0.963	0.982
0.894	0.966	**0.984**
0.900	0.968	**0.984**
0.912	**0.969**	**0.984**

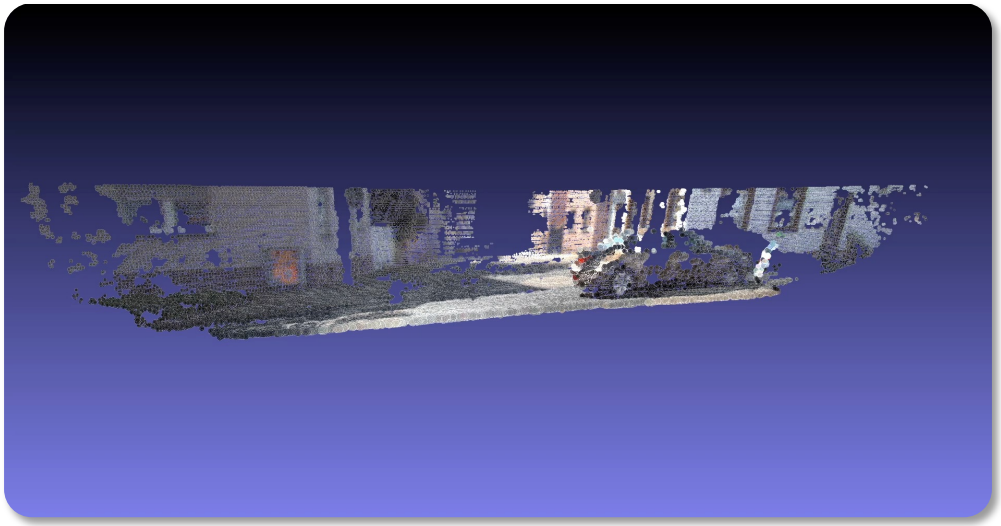
- We incorporate metrics from **Kitti Benchmark**
- Favour those that are more **interpretable**: MAE, RMSE, AbsRel…
- ➢ Forgotten **Scale Invariant Log**

$$e = \sqrt{\sum |\log(\hat{y}) - \log(y)|^2 - \left(\sum \log(\hat{y}) - \log(y)\right)^2}$$

Directional Term!

– Metrics so far focus on depth **accuracy along each ray**

➢ **Recall:** Objective is to recover **3D structure** of the scene!



– Örnek et al. propose to instead use **pointcloud-based** reconstruction metrics

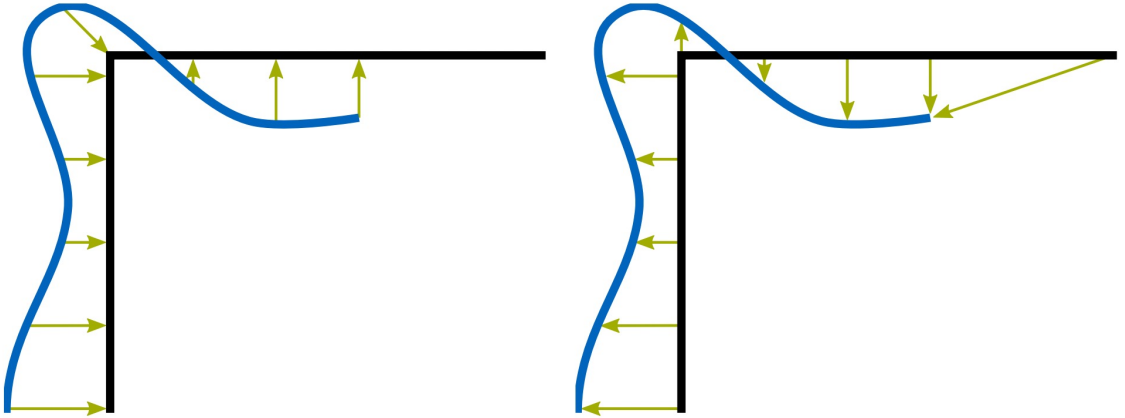
➢ **F-Score** & **IoU** based on Precision and Recall at **10cm accuracy**

Method	2D Metrics			3D Metrics				
	absrel↓	rmse↓	δ_1 ↑	CD↓	EMD↓	Comp.↓	IoU↑	F-score↑
Oracle NN	**0.097**	**0.225**	**0.891**	**0.257**	**0.130**	0.128	0.328	0.335
Median Plane	0.211	0.577	0.668	0.677	0.636	**0.042**	**0.347**	0.369
Eigen [3]	0.217	0.712	0.637	0.584	0.529	0.055	0.254	0.405
FCRN [4]	0.217	0.703	0.647	0.491	0.430	0.061	0.273	0.428
BTS [6]	0.190	0.657	0.694	0.454	0.400	0.055	0.336	**0.500**
VNL [36]	0.258	0.638	0.534	0.764	0.686	0.078	0.219	0.313

From 2D to 3D: Re-thinking Benchmarking of Monocular Depth Prediction, Örnek et al, arXiv22

– SYNS-Patches **depth boundary** metrics based on **IBims-1**

➢ **Chamfer distance** between predicted/ground-truth boundaries



Edge Accuracy

Edge Completeness

Evaluation of CNN-based Single-Image Depth Estimation Methods, Koch et al, ECCV-W18

Design Decisions

– Common **discrepancies** in training procedure

  - Input image size
  - Pretraining
  - Hyperparams
  - Architecture
  - Cherry picking

– We aim to **minimize changes** between approaches

– Another common source of error is **hyperparam tuning** on test set

➢ Can lead to overfitting over the course of optimization cycles

Test Time Refinement Model

One advantage of having a single-frame depth estimator is its wide applicability. However, this comes at a cost when running continuous depth estimation on image sequences as consecutive predictions are often misaligned or discontinuous. These are caused by two major issues 1) scaling inconsistencies between neighboring frames, since both our and related models have no sense of global scale, and 2) low temporal consistency of depth predictions. In this work we contend that fixing the model weights during inference is not required or needed and being able to *adapt* the model in an online fashion is advantageous, especially for practical autonomous systems. More specifically, we propose to keep the model training while performing inference, addressing these concerns by effectively performing online optimization. In doing that, we also show that even with very limited temporal resolution (i.e., three-frame sequences), we can significantly increase the quality of depth predictions both qualitatively and quantitatively. Having this low temporal resolution allows our method to still run on-line in real-time, with a typically negligible delay of a single frame. The online refinement is run for N steps ($N = 20$ for all experiments) which are effectively fine-tuning the model on-the-fly; N determines a good compromise between exploiting the online tuning sufficiently and preventing over-training which can cause artifacts. The online refinement approach can be seamlessly applied to any model including the motion model described above.

– We hope to mitigate these inconsistencies by **standardizing benchmarking procedure**

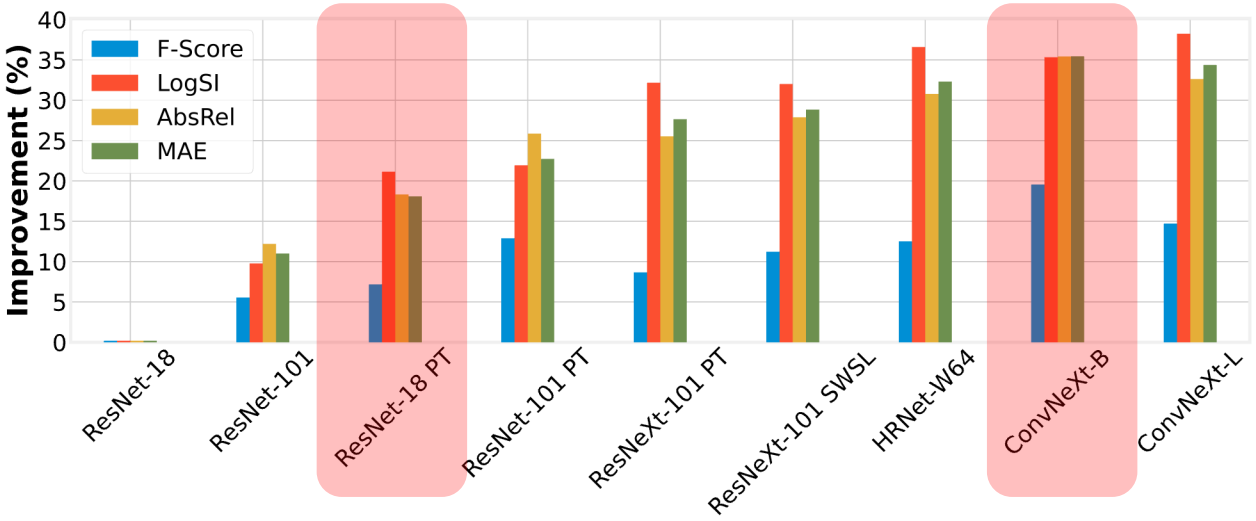
– Each model is trained with three different **random seeds**

➢ Report mean performance across seeds

– First ablation based on changing **backbone**

*Made simple by wonderful `timm` library!

➢ Simple* engineering changes vs. complex contributions



– Next ablation based on **regularization** techniques

➢ Edge-aware smoothness, second-order & occlusion

KE (test)	#	AbsRel↓	SqRel↓	RMSE↓	LogRMSE↓	$\delta < 1.25^1$↑	$\delta < 1.25^2$↑	$\delta < 1.25^3$↑
No Regularization	2	0.1002	0.7580	4.5833	0.1905	0.8865	0.9589	0.9798
First-order	1	**0.0993**	**0.7386**	4.5274	0.1873	0.8870	0.9608	0.9810
Fist-order Blur	7	0.1013	0.7600	4.5431	0.1886	0.8849	0.9603	0.9807
Second-order	4	0.1004	0.7567	**4.5260**	0.1877	**0.8871**	0.9609	0.9808
Second-order Blur	3	0.1003	0.7661	4.5512	0.1881	0.8867	0.9607	0.9806
Occlusion (BG)	6	0.1004	0.7573	4.5454	0.1872	0.8850	0.9607	**0.9811**
Occlusion (FG)	5	0.1004	0.7623	4.5326	**0.1872**	0.8870	**0.9611**	0.9810

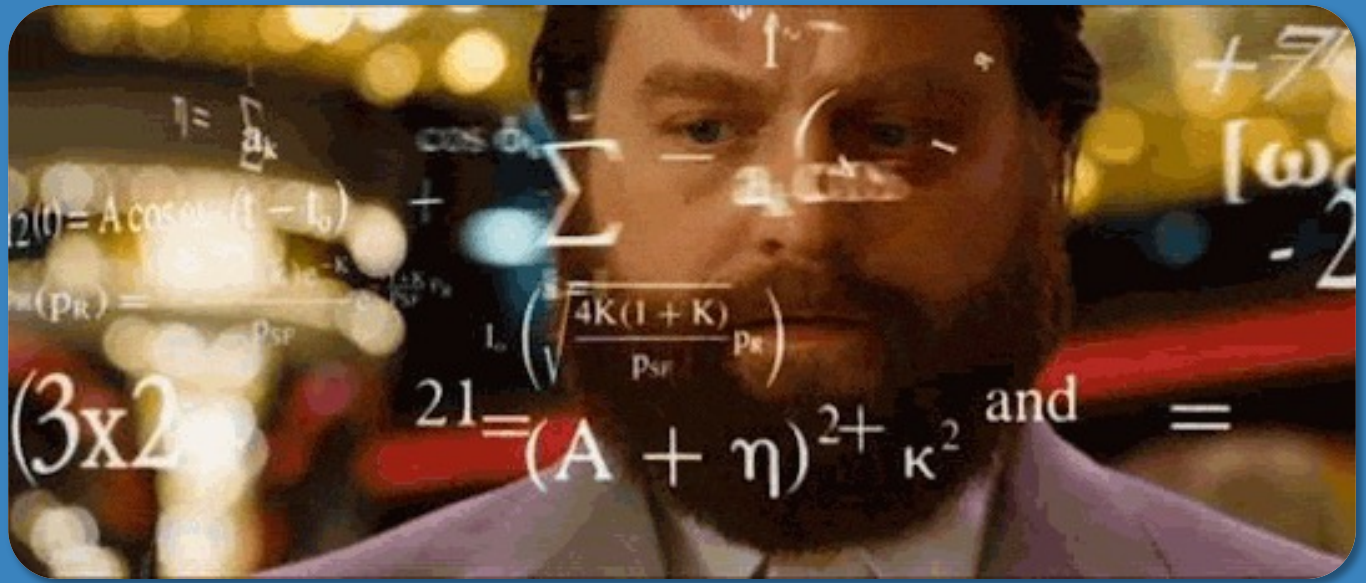
Kitti Eigen Test Split

– Next ablation based on **regularization** techniques

➢ Edge-aware smoothness, second-order & occlusion

	Image-based					Pointcloud-based			
KEZ (val)	#	MAE↓	RMSE↓	AbsRel↓	LogSI↓	#	Chamfer↓	F-Score↑	IoU↑
No Regularization	**1**	**1.63**	3.68	**7.86**	11.35	**1**	**0.66**	**50.25**	**34.68**
First-order	3	1.65	**3.64**	8.12	11.32	6	0.67	49.30	33.90
Fist-order Blur	5	1.65	3.66	8.19	11.36	4	0.67	49.39	33.96
Second-order	2	1.64	3.64	8.10	11.25	5	0.67	49.32	33.93
Second-order Blur	4	1.65	3.66	8.14	11.27	2	0.67	49.46	34.05
Occlusion (BG)	7	1.66	3.65	8.27	11.38	7	0.68	48.86	33.52
Occlusion (FG)	6	1.65	3.65	8.19	**11.23**	3	0.67	49.40	34.02

Kitti Eigen-Zhou Val Split



Results

MONO (M)
- SfM-Learner
- Klodt
- Johnston

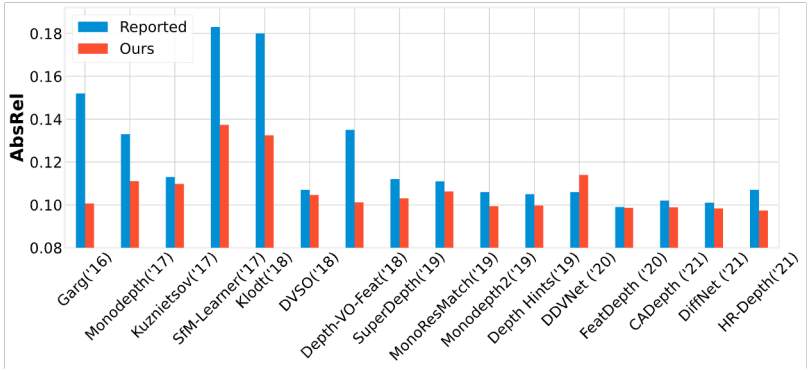
STEREO (S)
- Garg
- Monodepth
- SuperDepth

MS
- Depth-VO-Feat
- Monodepth2
- FeatDepth
- CADepth
- DiffNet
- HR-Depth

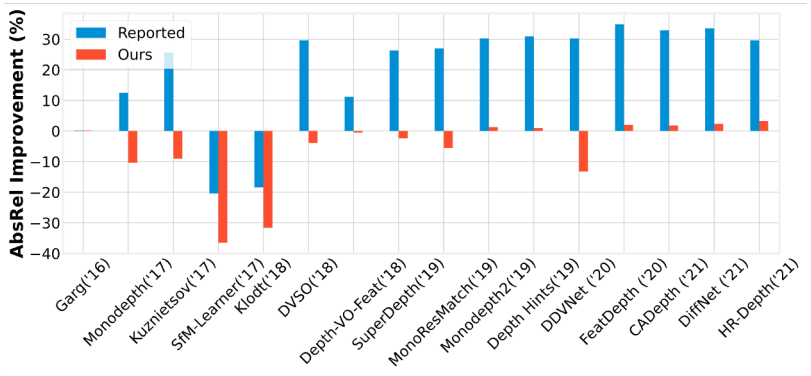
PROXY (D*)
- Kuznietsov
- DVSO
- MonoResMatch
- DepthHints

– Evaluation on **Kitti Eigen**

➢ Performance vs. original is better, but relative w.r.t. Garg is much lower



Absolute – Lower is Better



Relative – Higher is Better

		Image-based				Pointcloud-based		
Kitti Eigen-Benchmark (test)	Train	MAE↓	RMSE↓	AbsRel↓	LogSI↓	Chamfer↓	F-Score↑	IoU↑
SfM-Learner	M	1.98	4.57	10.69	15.80	0.73	44.77	30.03
Klodt	M	1.96	4.54	10.49	15.86	0.72	45.26	30.40
Monodepth2	M	1.84	4.11	8.82	13.10	0.71	46.64	31.62
Johnston	M	1.80	4.04	8.65	12.75	0.69	47.35	32.10
HR-Depth	M	1.83	3.99	8.85	12.89	0.71	45.72	30.78
Garg	S	1.60	3.75	**7.65**	11.39	**0.60**	**53.28**	**37.33**
Monodepth	S	1.76	3.99	8.65	13.08	0.67	49.80	34.09
SuperDepth	S	1.64	3.77	7.81	11.63	0.63	52.30	36.40
Depth-VO-Feat	MS	1.63	3.72	7.70	11.64	0.62	52.01	36.15
Monodepth2	MS	1.61	3.62	7.90	10.99	0.64	50.50	34.98
FeatDepth	MS	1.60	3.60	7.80	11.01	0.65	49.99	34.51
CADepth	MS	1.63	3.60	8.09	10.84	0.66	49.32	34.06
DiffNet	MS	1.62	3.63	7.97	10.93	0.65	49.63	34.23
HR-Depth	MS	**1.58**	**3.56**	7.70	**10.68**	0.62	51.49	35.93
Kuznietsov	SD*	1.82	3.98	9.32	11.80	0.71	45.80	30.63
DVSO	SD*	1.71	3.89	8.38	11.38	0.68	48.50	32.97
MonoResMatch	SD*	1.65	3.79	7.90	11.74	0.66	50.70	34.92
DepthHints	MSD*	1.63	3.62	8.10	10.94	0.66	49.30	33.80

Kitti Eigen-Benchmark Test Split

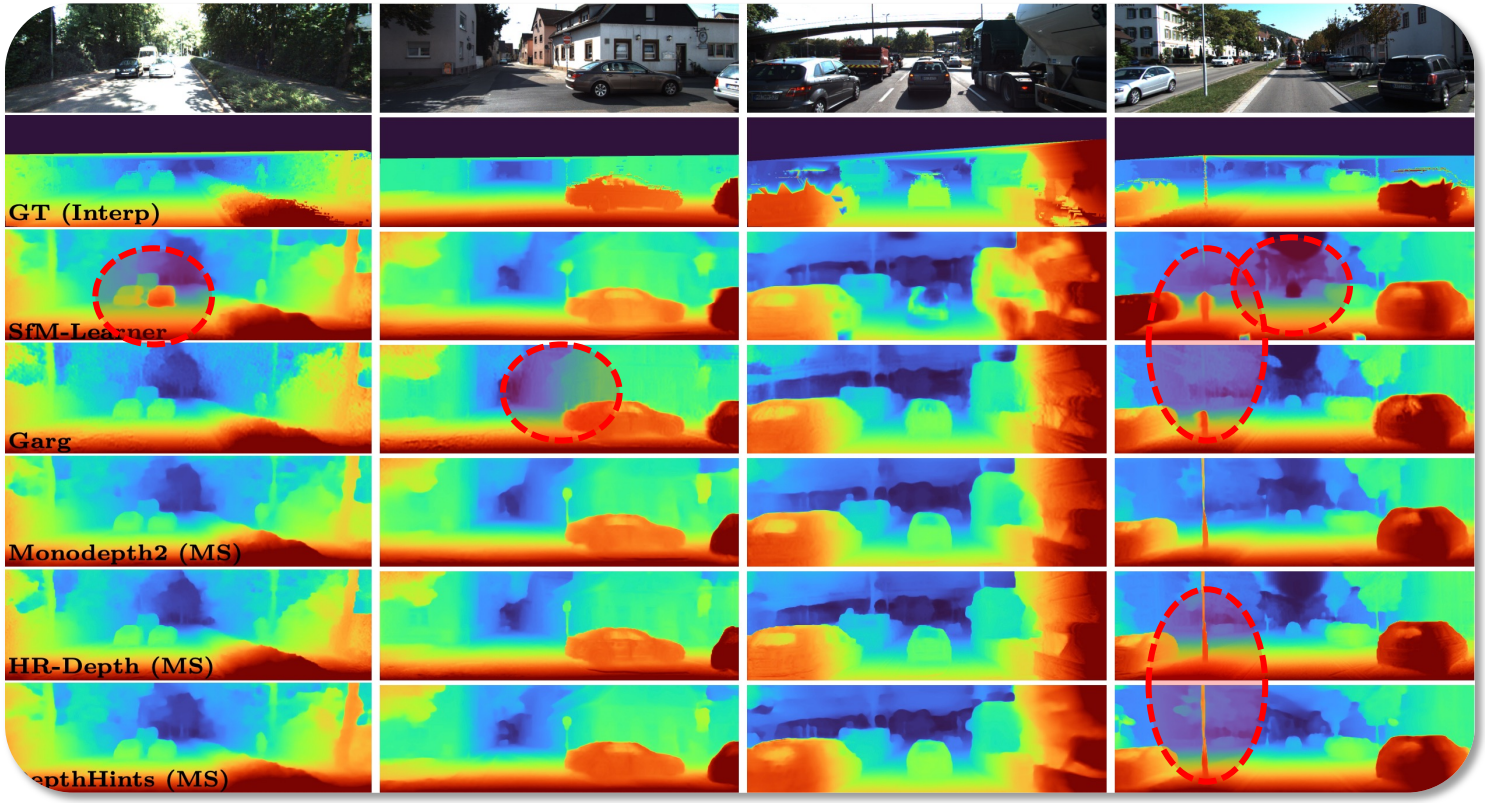
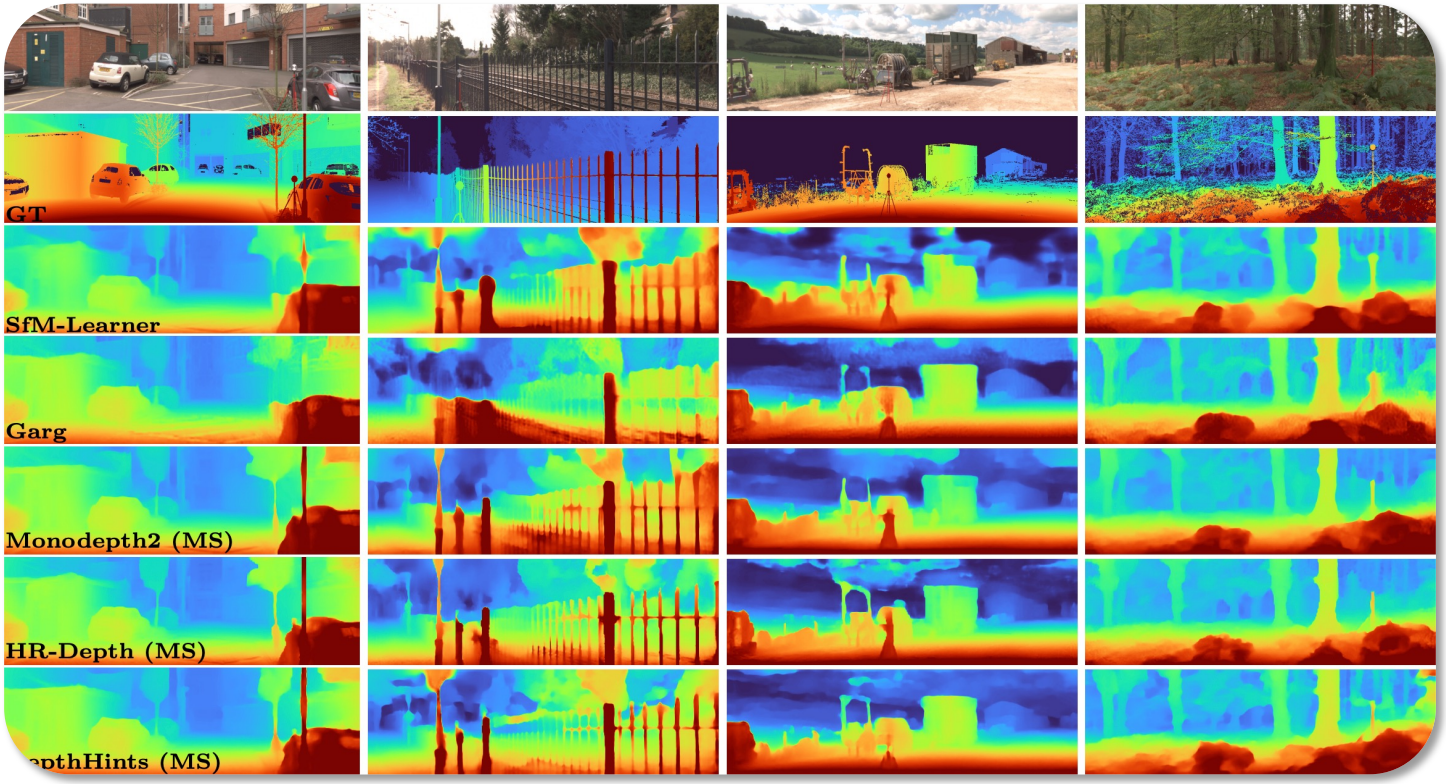

GT (Interp)
SfM-Learner
Garg
Monodepth2 (MS)
HR-Depth (MS)
epthHints (MS)

		Image-based				Pointcloud-based			Edge-based		
SYNS-Patches	Train	MAE↓	RMSE↓	LogSI↓	AbsRel↓	Chamfer↓	F-Score↑	IoU↑	Acc↓	Comp↓	F-Score↑
SfM-Learner	M	5.43	9.25	36.91	31.58	2.66	11.79	6.43	3.46	36.12	8.47
Klodt	M	5.40	9.20	36.40	31.20	2.57	12.00	6.57	3.44	35.22	8.48
Monodepth2	M	5.33	9.02	35.62	30.05	2.78	12.08	6.62	3.30	37.01	8.46
Johnston	M	5.26	8.95	34.90	29.53	2.59	13.37	7.40	**3.07**	30.03	9.16
HR-Depth	M	5.24	8.92	35.28	29.72	2.74	12.16	6.66	3.23	42.82	8.60
Garg	S	5.29	9.20	35.81	30.73	2.41	13.48	7.45	3.37	26.79	**9.53**
Monodepth	S	5.36	9.14	36.50	31.32	2.92	12.04	6.62	3.62	68.31	8.43
SuperDepth	S	5.26	9.08	35.82	30.83	2.72	12.87	7.10	3.40	40.40	9.01
Depth-VO-Feat	MS	5.30	9.17	35.95	30.83	2.52	12.43	6.82	3.50	38.49	8.77
Monodepth2	MS	5.18	8.91	35.05	29.04	2.63	13.18	7.27	3.38	32.69	8.95
FeatDepth	MS	5.16	**8.80**	34.94	29.12	2.68	12.27	6.73	3.50	44.09	8.41
CADepth	MS	5.22	8.97	34.99	29.80	2.45	12.83	7.06	3.42	35.89	8.70
DiffNet	MS	5.16	8.91	**34.66**	**28.80**	2.55	13.16	7.26	3.45	39.46	8.81
HR-Depth	MS	**5.13**	8.85	34.79	28.94	2.43	**13.79**	**7.65**	3.25	28.33	9.21
Kuznietsov	SD*	5.47	9.50	35.56	31.08	2.44	13.15	7.26	3.39	47.13	9.11
DVSO	SD*	5.21	8.98	35.04	29.79	2.62	12.68	7.00	3.48	57.57	9.03
MonoResMatch	SD*	5.17	8.92	35.24	29.76	2.81	12.34	6.80	3.56	67.71	8.76
DepthHints	MSD*	5.33	9.07	35.70	30.90	**2.37**	12.91	7.11	3.24	**26.21**	9.01

SYNS-Patches Test Split


GT
SfM-Learner
Garg
Monodepth2 (MS)
HR-Depth (MS)
pthHints (MS)

Conclusions

– **Classical methods** are still highly competitive

➢ Be wary of taking benchmarking rankings as absolute

– **Monocular** supervision still sensitive to **dynamic objects**

➢ **Monodepth2** contributions are great for mitigating this!

– Check out our **workshop/challenge** at WACV23!

➢ Submissions will likely close **early November**

Questions?

Code

Paper

MDEC @ WACV23

CVSSP | Centre for Vision, Speech and Signal Processing