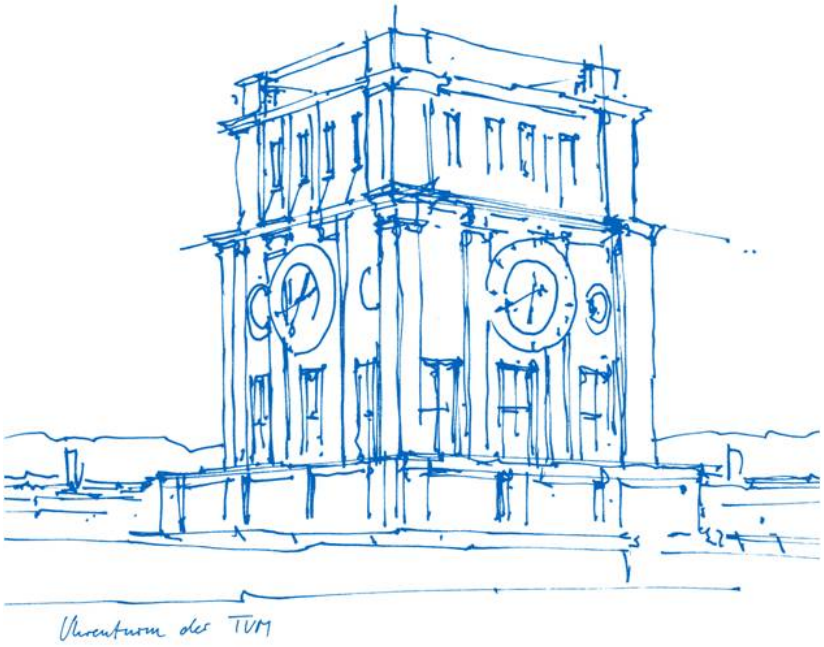
Seminar: The Evolution of Motion Estimation and Real-time 3D Reconstruction

Referent: Jingpei Wu

Technische Universität München

Fakultät für Informatik

München, 16 April 2020



Presentation Paper [ECCV 2018]

DeepTAM: Deep Tracking and Mapping

Huizhong Zhou, Benjamin Ummenhofer, Thomas Brox

Outline

- Introduction
- Method description
- Experiments and results
- Personal comments
- Summary

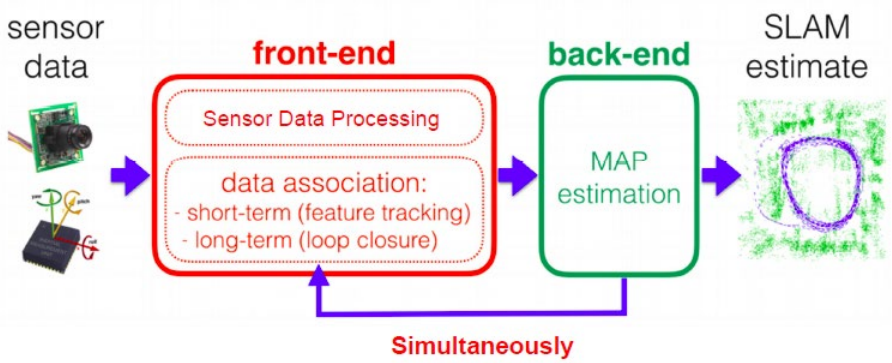
Outline

- Introduction
- Method description
  - Tracking
  - Mapping
- Experiments and results
  - Tracking
  - Mapping
- Personal comments
- Summary

Introduction

- Good solutions from traditional methods (filter-based and optimization-based) and a long history of research
- Development of deep learning and its amazing performance in computer vision tasks, e.g. classification, object detection, semantic segmentation
- Better performance with deep learning in camera tracking and mapping field than classical approaches
- Comparison: better representation of features, robustness but likely to overfit
- Categories: (1) Non-end-to-end learning (2) End-to-end learning (3) Unsupervised learning

- Here: keyframe-based dense camera tracking and depth map estimation with end-to-end learning approach



[Cadena et al. 2016]

Outline

- Introduction
- Method description
  - Tracking
  - Mapping
- Experiments and results
  - Tracking
  - Mapping
- Personal comments
- Summary

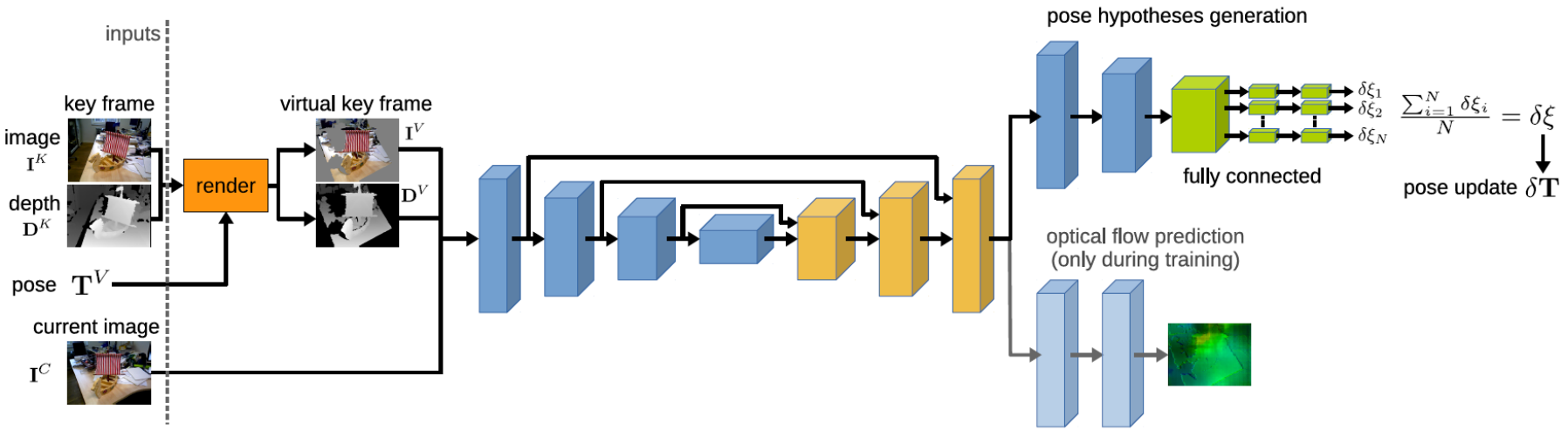
Tracking

Relation between keyframe pose T^K and current camera pose T^C

$$\mathbf{T}^C = \mathbf{T}^K \mathbf{T}^{KC}, \text{ with } \mathbf{T}^C, \mathbf{T}^K, \mathbf{T}^{KC} \in SE(3)$$

With virtual keyframe (I^V, D^V) showing the keyframe from the viewpoint of T^V

$$\mathbf{T}^C = \mathbf{T}^V \delta\mathbf{T} \qquad \delta\mathbf{T} = f(\mathbf{I}^C, \mathbf{I}^V, \mathbf{D}^V)$$

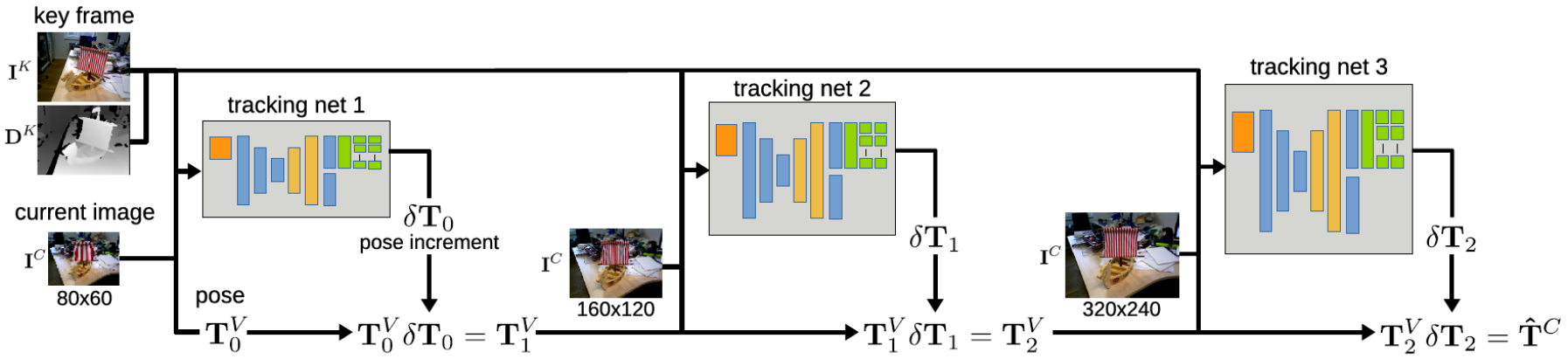


- Optical flow prediction as an auxiliary task
- Pose generation: fully connected layers

$$\delta\xi_i = (\mathbf{r}_i, \mathbf{t}_i)^T \qquad \delta\xi = \frac{1}{N} \sum_{i=1}^{N=64} \delta\xi_i$$

Tracking Network

- Coarse-to-fine strategy
  - Coarse camera motions visible at small image resolutions
  - Small motions require higher image resolutions



Training of tracking network

- Objective function

$$\mathcal{L}_{\text{tracking}} = \mathcal{L}_{\text{flow}}(w) + \mathcal{L}_{\text{motion}}(\delta\xi) + \mathcal{L}_{\text{uncertainty}}(\delta\xi_i)$$

w is the predicted optical flow and $\delta\xi_i$ is the predicted poses

$$\mathcal{L}_{\text{flow}} = \sum_{i,j} \| w(i,j) - w_{gt}(i,j) \|_2$$

endpoint error, metric for optical flow

$$\mathcal{L}_{\text{motion}} = \alpha \| \mathbf{r} - \mathbf{r}_{gt} \|_2 + \| \mathbf{t} - \mathbf{t}_{gt} \|_2$$

transformation error (rotation and translation)

$$\mathcal{L}_{\text{uncertainty}} = \frac{1}{2}\log(|\boldsymbol{\Sigma}|) - 2\log\left(\frac{\mathbf{x}^T\boldsymbol{\Sigma}^{-1}\mathbf{x}}{2}\right) - \log(K_v(\sqrt{2\mathbf{x}^T\boldsymbol{\Sigma}^{-1}\mathbf{x}}))$$

negative log-likelihood of the multivariate Laplace distribution

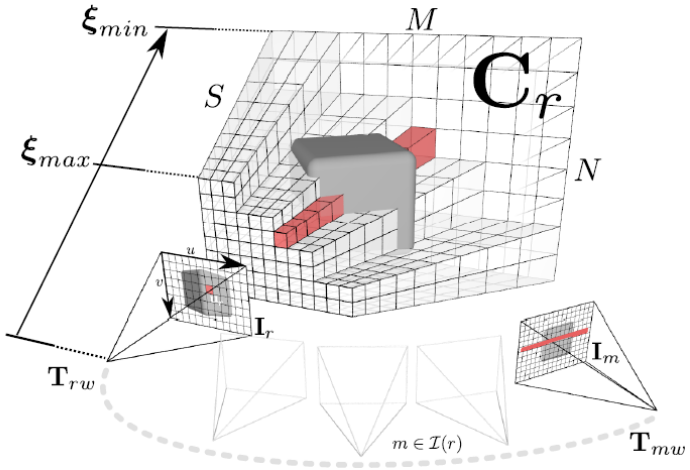
- Datasets: SUN3D and SUNCG
- Solutions for overfitting
  - incremental formulation, which reduces the magnitude of motion
  - rendered images and depth maps as a proxy for real keyframes, normal distribution sampling simulates all possible 6 DOF motions

Outline

- Introduction
- Method description
  - Tracking
  - Mapping
- Experiments and results
  - Tracking
  - Mapping
- Personal comments
- Summary

Mapping

- Geometry of a scene: a set of depth maps for every keyframe
- Cost volume: achieved by accumulating information from multiple images



Let: C the cost volume, $C(x,d)$ the photo-consistency cost for a pixel x at depth label $d \in B_{fb}$

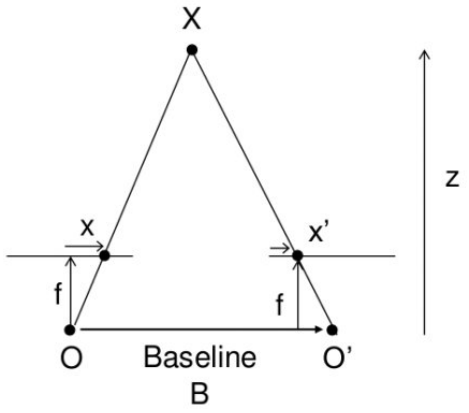
$$B_{\text{fb}} = \left\{ b_i \mid b_i = d_{\min} + i \cdot \frac{d_{\max} - d_{\min}}{N-1}, i = 0,1,\ldots,N-1 \right\}$$

given m images $I_1, \ldots, I_m$ and their poses $T_1, \ldots, T_m$, the photo-consistency costs as

$$\mathbf{C}(\mathbf{x},d) = \sum_{i \in \{1,\ldots,m\}} \rho_i(\mathbf{x},d) \cdot w_i(\mathbf{x})$$

$\rho_i(x,d)$: the sum of absolute differences of 3×3 patches between I^K and the warped image $\tilde{I}_i$

$w_i(x)$: weighting factor (confidence)



disparity

Newcombe et al., "DTAM: Dense Tracking and Mapping in Real-Time", 2011

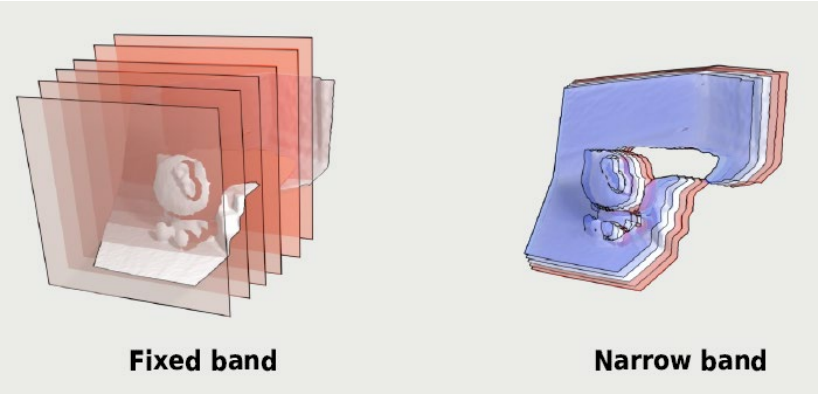
OpenCV: Depth Map from Stereo Images

Mapping

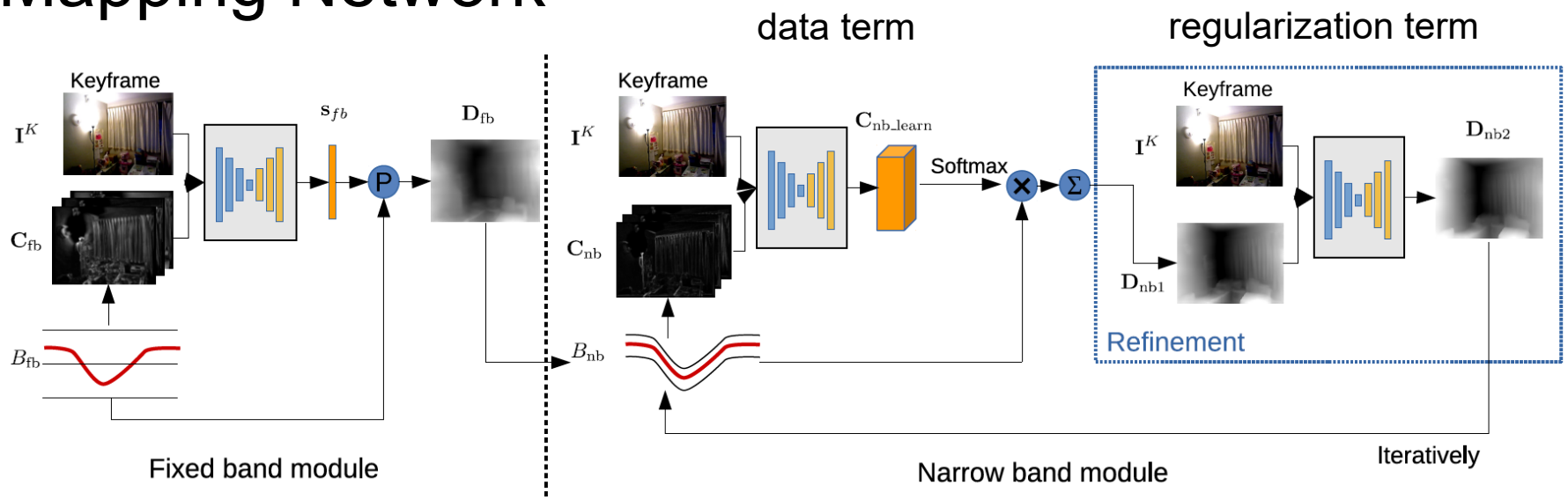
- depth map: extracted from the cost volume using CNN
- combines matching cost information in the cost volume and image-based scene priors
- accuracy OR computational efficiency?

narrow band of depth labels centered at the previous depth estimate d_{prev} as

$$B_{\text{nb}} = \left\{ b_i \mid b_i = d_{\text{prev}} + i \cdot \sigma_{\text{nb}} \cdot d_{\text{prev}}, i = -\frac{N}{2}, \ldots, \frac{N-2}{2} \right\}$$



Mapping Network



I^K
$(320 \times 240 \times 3)$

C_{fb}
$(320 \times 240 \times 32)$

s_{fb}
$(320 \times 240 \times 1)$

$$D_{fb} = (1 - s_{fb}) \cdot d_{min} + s_{fb} \cdot d_{max}$$

I^K
$(320 \times 240 \times 3)$

C_{nb_learn}
$(320 \times 240 \times 32)$

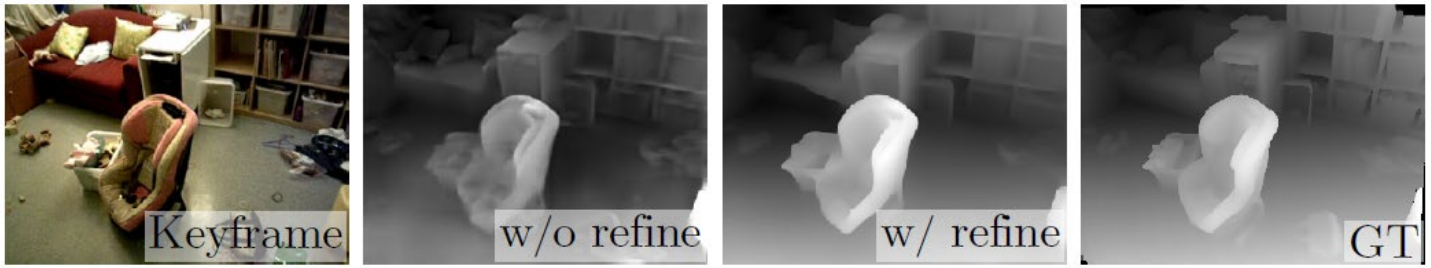
C_{nb}
$(320 \times 240 \times 32)$
$\sigma_{nb} = 0.0125$

$$D_{nb1} = \Sigma_d \, B_{nb} \times softmax(-C_{nb\_learn})$$

I^k → D_{nb2}

Training of mapping network

- Effect of narrow band refinement: focus on depth regularization (15 iterations)



- Objective function
  - L1 loss on the inverse depth maps

$$\mathcal{L}_{\text{depth}} = \left| \mathbf{D} - \mathbf{D}_{gt} \right|$$

  - The scale-invariant gradient loss

$$\mathcal{L}_{\text{sc-inv-grad}} = \sum_{h \in \{1,2,4\}} \sum_{i,j} \left\| \mathbf{g}_h[\mathbf{D}](i,j) - \mathbf{g}_h[\mathbf{D}_{\text{gt}}](i,j) \right\|_2$$

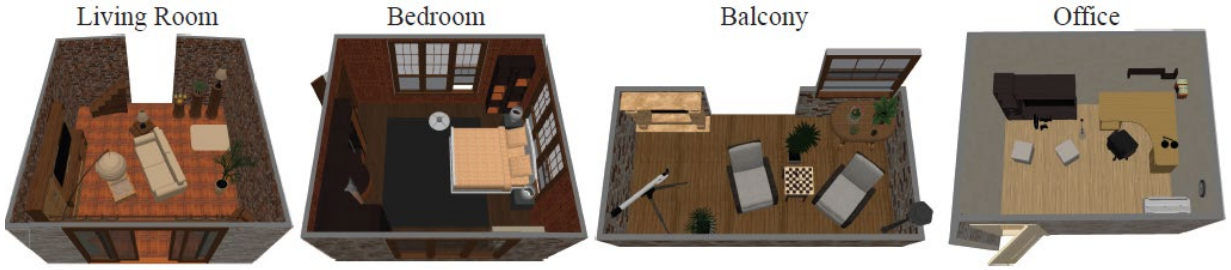
$g_h[D](i,j)$ is gradient images of the predicted depth map that emphasize discontinuities

$$\mathbf{g}_h[\mathbf{D}](i,j) = \left(\frac{\mathbf{D}(i+h,j) - \mathbf{D}(i,j)}{|\mathbf{D}(i+h,j)| + |\mathbf{D}(i,j)|}, \frac{\mathbf{D}(i,j+h) - \mathbf{D}(i,j)}{|\mathbf{D}(i,j+h)| + |\mathbf{D}(i,j)|} \right)^{\top}$$

Training datasets

- SUN3D: large variety of indoor scenes

- SUNCG: synthetic dataset of 3D scenes with realistic scene scale



- MVG: contains both indoor and outdoor scenes

Xiao et al., “SUN3D: A Database of Big Spaces Reconstructed using SfM and Object Labels”, 2013
Song et al., “Semantic Scene Completion from a Single Depth Image”, 2016 (SUNCG)

Outline

- Introduction
- Method description
  - Tracking
  - Mapping
- Experiments and results
  - Tracking
  - Mapping
- Personal comments
- Summary

Tracking evaluation

- RGB-D benchmark: images and depth maps with accurate ground truth poses
- Effects of optical flow prediction and multiple pose hypotheses prediction
- Generalization: not train or finetune on this benchmark

	Tracking			
Sequence	RGB-D SLAM Kerl *et al.* [17]	Ours (w/o flow)	Ours (w/o hypotheses)	Ours
fr1/360	0.125	0.069	0.065	**0.054**
fr1/desk	0.037	0.042	0.031	**0.027**
fr1/desk2	0.020	0.025	0.020	**0.017**
fr1/plant	0.062	0.063	0.060	**0.057**
fr1/room	0.042	0.051	0.041	**0.039**
fr1/rpy	0.082	0.070	**0.063**	0.065
fr1/xzy	0.051	0.030	0.021	**0.019**
average	0.060	0.050	0.043	**0.040**

Values describe the translational RMSE in [m/s]
Ours (w/o flow): not learn optical flow
Ours (w/o hypotheses): just a single pose

Sturm et al., “A benchmark: for the evaluation of RGB-D SLAM systems”, 2012
Kerl et al., “Dense visual SLAM for RGB-D cameras”, 2013

Outline

- Introduction
- Method description
  - Tracking
  - Mapping
- Experiments and results
  - Tracking
  - Mapping
- Personal comments
- Summary

Mapping evaluation

- Error metrics
  - scale invariant metric [1]

$$\text{sc-inv}\left(\mathbf{D}, \mathbf{D}_{\text{gt}}\right) = \sqrt{\frac{1}{n}\sum_{i,j}\mathbf{E}(i,j)^2 - \frac{1}{n^2}\left(\sum_{i,j}\mathbf{E}(i,j)\right)^2}$$

where $E(i,j) = \log D(i,j) - \log D_{gt}(i,j)$ and n the number of pixels

  - L1-rec: normalizes the depth error with respect to the ground truth depth value

$$\text{L1-rel}\left(\mathbf{D}, \mathbf{D}_{\text{gt}}\right) = \frac{1}{n}\sum_{i}\frac{\left|\mathbf{D}(i,j) - \mathbf{D}_{\text{gt}}(i,j)\right|}{\mathbf{D}_{\text{gt}}(i,j)}$$

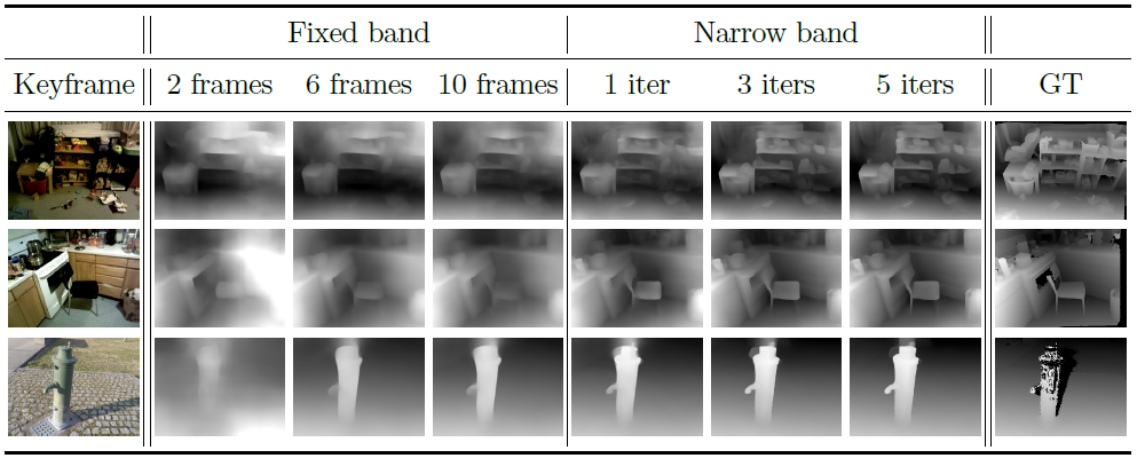
  - L1-inv: gives more importance to close depth values

$$\text{L1-inv}(\mathbf{D}, \mathbf{D}_{\text{gt}}) = \frac{1}{n}\sum_{\text{i}}\left|\frac{1}{\mathbf{D}(\text{i},\text{j})} - \frac{1}{\mathbf{D}_{\text{gt}}(\text{i},\text{j})}\right|$$

[1] Eigen et al., “Depth Map Prediction from a Single Image using a Multi-Scale Deep Network”, 2014

Fixed band and narrow band module

- qualitatively



- quantitatively

		Fixed band			Narrow band		
		2frames	6frames	10frames	1iter	3iters	5iters
MVS	L1-inv	0.117	0.085	**0.083**	0.076	0.065	**0.064**
	L1-rel	0.239	0.163	**0.159**	0.142	0.113	**0.111**
	sc-inv	0.193	0.160	**0.159**	0.156	0.132	**0.130**
SUNCG	L1-inv	0.075	**0.065**	0.067	0.049	0.039	**0.036**
	L1-rel	0.439	**0.418**	0.423	0.304	0.213	**0.171**
	sc-inv	0.213	**0.199**	0.200	0.174	0.152	**0.146**
SUN3D	L1-inv	0.097	0.067	**0.065**	0.050	**0.035**	0.036
	L1-rel	0.288	0.198	**0.193**	0.141	**0.082**	0.083
	sc-inv	0.206	0.174	**0.172**	0.155	**0.125**	0.128

keyframe depth map errors on the test split of training data sets

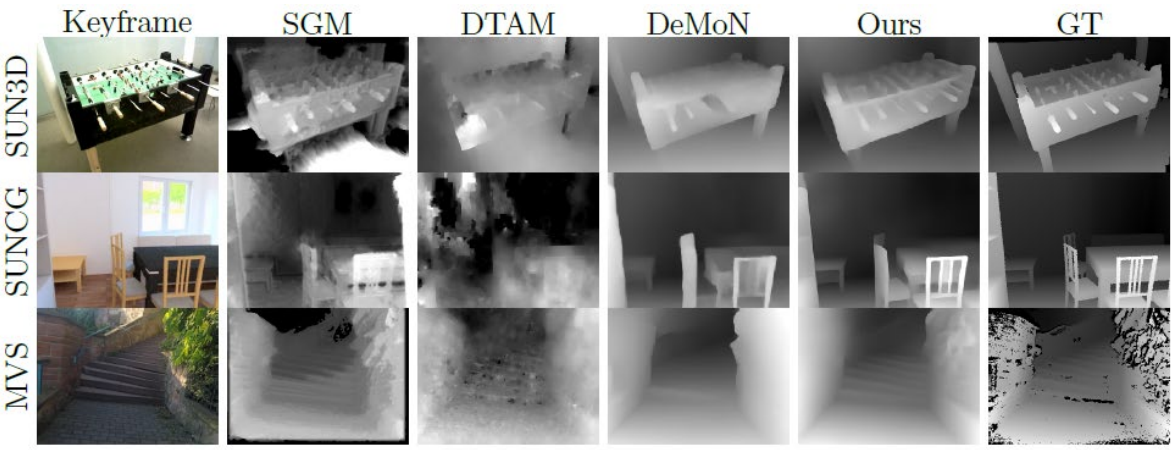
Comparison

- quantitatively

		Mapping comparison			
		SGM	DTAM	DeMoN	Ours
MVS	L1-inv	-	0.086	0.059	**0.036**
	L1-rel	-	0.557	0.240	**0.171**
	sc-inv	0.251	0.305	0.246	**0.146**
SUNCG	L1-inv	-	0.142	0.169	**0.036**
	L1-rel	-	0.380	0.533	**0.083**
	sc-inv	0.248	0.343	0.383	**0.128**
SUN3D	L1-inv	-	0.210	0.197	**0.064**
	L1-rel	-	0.423	0.412	**0.111**
	sc-inv	0.146	0.374	0.340	**0.130**

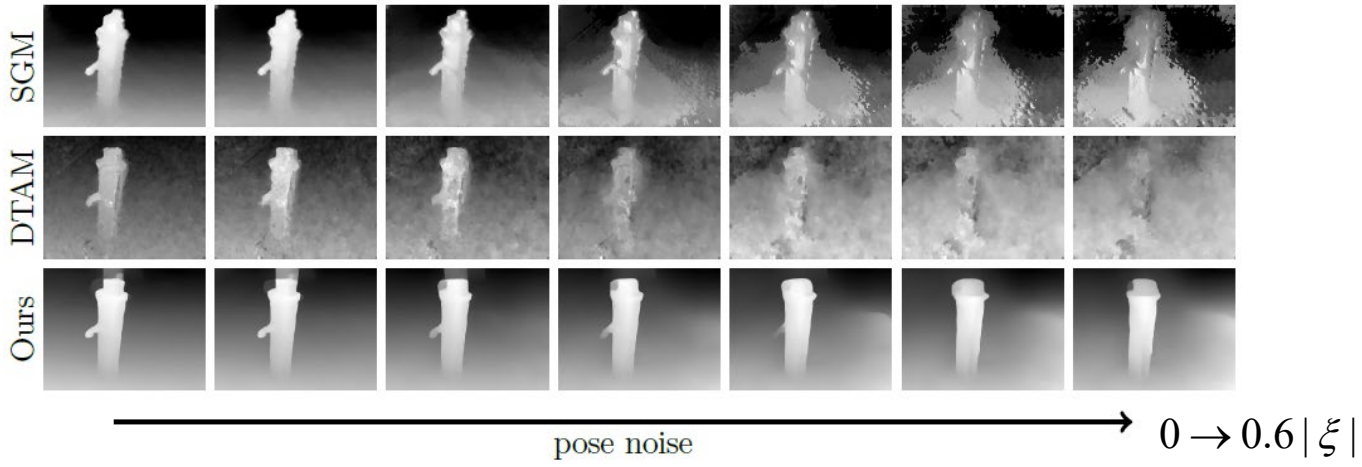
	Tracking and mapping	
Sequence	CNN-SLAM* Tateno *et al.* [29]	Ours
fr1/360	0.500	**0.116**
fr1/desk	0.095	**0.078**
fr1/desk2	0.115	**0.055**
fr1/plant	**0.150**	0.165
fr1/room	0.445	**0.084**
fr1/rpy	0.261	**0.052**
fr1/xzy	0.206	**0.054**
average	0.253	**0.086**

- qualitatively

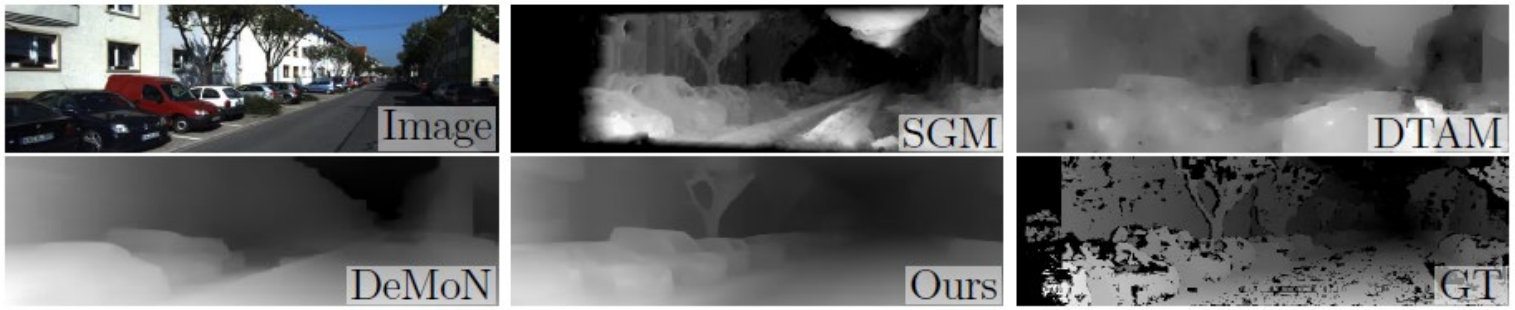


Robustness & Generalization

- Robustness: apply the same normal-distributed noise vectors to the camera poses



- Generalization experiment on KITTI



Outline

- Introduction
- Method description
  - Tracking
  - Mapping
- Experiments and results
  - Tracking
  - Mapping
- **Personal comments**
- Summary

Personal comments

- The method showed the great performance, robustness and generalization capabilities, the way to handle the problem seemed to be more straightforward than classic methods which incorporate several modules and need much mathematics

- The neural network design involved many useful tricks: virtual keyframe synthetics, multiple pose hypotheses, optical flow prediction (enforce useful learning), coarse-to-fine strategy, appending refinement network etc.

- The detailed architecture design should be difficult: hyperparameters, skip connections, branches, etc.? → available backbone architecture from deep learning field

- Deep learning methods lack explainability relatively, combining traditional modules and deep learning is still powerful and combining with other subtasks like segmentation could provide more progress

- Incorporate with loop closure

Outline

- Introduction
- Method description
  - Tracking
  - Mapping
- Experiments and results
  - Tracking
  - Mapping
- Personal comments
- Summary

Summary

- This paper propose a deep learning architecture for real-time dense mapping and tracking

- Tracking
  - generating synthetic viewpoints allows to track incrementally with respect to a keyframe
  - a multiple hypothesis approach for camera poses leads to more accurate pose estimation

- Mapping
  - neural networks combining cost volume information and image-based priors lead to accurate and robust dense depth estimation
  - an efficient depth refinement strategy combining a network with the narrow band technique helps for finer details

- runtime: 44Hz for tracking
  (on NVIDIA GTX 1070)

	Tracking	Cost volume	Fixed band	Narrow band
Mean	0.0227	0.0164	0.0181	0.0359
Min	0.0203	0.0153	0.0171	0.0347
Max	0.0251	0.0168	0.0190	0.0393

Thanks for listening

Questions ?

Xiao et al., “SUN3D: A Database of Big Spaces Reconstructed using SfM and Object Labels”, 2013
Song et al., “Semantic Scene Completion from a Single Depth Image”, 2016 (SUNCG)
Newcombe et al., “DTAM: Dense tracking and mapping in real-time”, 2011
Hirschmüller et al., “Accurate and efficient stereo processing by semi-global matching and mutual information”, 2005 (SGM)
Tateno et al. “CNN-SLAM: Real-Time Dense Monocular SLAM with Learned Depth Prediction”, 2017
Ummenhofer et al., “DeMoN: Depth and Motion Network for Learning Monocular Stereo”, 2017

Supplementary Formulations

multivariate Laplace distribution, if $\boldsymbol{\mu} = 0$

$$\frac{2}{(2\pi)^{k/2} \mid \boldsymbol{\Sigma} \mid^{0.5}} \left(\frac{\mathbf{x}' \boldsymbol{\Sigma}^{-1} \mathbf{x}}{2} \right)^{v/2} K_v(\sqrt{2\mathbf{x}' \boldsymbol{\Sigma}^{-1} \mathbf{x}})$$

where $v = (2-k)/2$ and K_v is the modified Bessel function of the second kind

$$\boldsymbol{\Sigma} = \frac{1}{N} \sum_{i}^{N} (\delta\xi_i - \delta\xi)(\delta\xi_i - \delta\xi)^T, \mathbf{x} = \delta\xi - \delta\xi_{gt}$$

Cost Volume

$$\mathbf{C}(\mathbf{x}, d) = \sum_{i \in \{1,\ldots,m\}} \rho_i(\mathbf{x}, d) \cdot w_i(\mathbf{x})$$

$w_i(x)$: weighting factor, with $d^* = \arg\min_{\mathrm{d}} \rho_i(x, d)$

$$w_i(\mathbf{x}) = 1 - \frac{1}{N-1} \sum\nolimits_{d \in B_{\mathrm{fb}} \setminus \{d^*\}} \exp\left(-\alpha \cdot \left(\rho_i(\mathbf{x}, d) - \rho_i\left(\mathbf{x}, d^*\right) \right)^2 \right)$$