Supplementary Material
Unsupervised Learning for Robust Fitting: A Reinforcement Learning Approach

Giang Truong* Huu Le** David Suter* Erchuan Zhang* Syed Zulqarnain Gilani*
*School of Science, Edith Cowan University, Australia
**Department of Electrical Engineering, Chalmers University of Technology
{h.truong, d.suter, erchuan.zhang, s.gilani}@ecu.edu.au, huul@chalmers.se

March 25, 2021

1 Extension of the algorithm to quasi-convex residual cases

This experiment demonstrates that our proposed method performs favorably for fitting problems with *quasi-convex residuals*. One of the common applications is homography fitting. Given a set of putative correspondences $\mathcal{X} = \{(\mathbf{u}_i, \mathbf{v}_i)\}_{i=1}^N$, we estimate the homography matrix $\boldsymbol{\theta} \in \mathbb{R}^{3\times 3}$. The residual is of the form:

$$r_i(\boldsymbol{\theta}) = \frac{||(\boldsymbol{\theta}_{1:2} - \mathbf{v}_i\boldsymbol{\theta}_3)\hat{\mathbf{u}}_i||}{\boldsymbol{\theta}_3\hat{\mathbf{u}}_i},$$

where $\hat{\mathbf{u}}_i = [\mathbf{u}_i^T \; 1]^T$, $\boldsymbol{\theta}_{1:2}$ is the first 2 rows of $\boldsymbol{\theta}$, and $\boldsymbol{\theta}_3$ is the last row of $\boldsymbol{\theta}$. For this experiment, the state encoding $h(\mathbf{x}_i)$, where $\mathbf{x}_i = [\mathbf{u}_i, \mathbf{v}_i]$ discussion in Eq.(10) is chosen to be

$$h(\mathbf{x}_i) = [\mathbf{u}_i^T \;\; \mathbf{v}_i^T].$$

We compare our algorithm with other methods as described in the main paper. We conduct our experiment in one sequence of KITTI dataset [1]. We compute and match SIFT keypoints using VLFeat toolbox. The inlier threshold is chosen as $\epsilon = 0.03$. As shown in Fig. A2, our method generally achieves higher consensus size, compared to ULCM. When compared to RANSAC, despite the fact that our spread is larger, we have a smaller variance and higher median consensus size. This means that our method has much higher concentration on good results (with small tail to the distribution: *only rarely* producing poor results).

2 The role of Local Tree Refinement

Fig. A3 plots the consensus size in 2D Line Fitting experiment before and after applying refinement (Section 3.6 in the main paper). All results are obtained by running 100 times. Even though our RL based method performs competitively, the refinement process slightly improves the results. Note especially the performance of local refinement for outlier rates $> 25\%$.

3 Experiment on real data for 2D and 3D fitting

We have performed some experiments on real data: for 2D line fitting by capturing images containing line segments of interest (e.g., pens, rulers,...) in a scene surrounded by distracting objects (to generate outliers), and for 3D data by segmenting several walls containing a large fraction of outliers from the point clouds provided by the 7-Scene dataset [3]. Fig. A1 shows visualization of our experiment. Indeed, our method works equally well on challenging real world data.

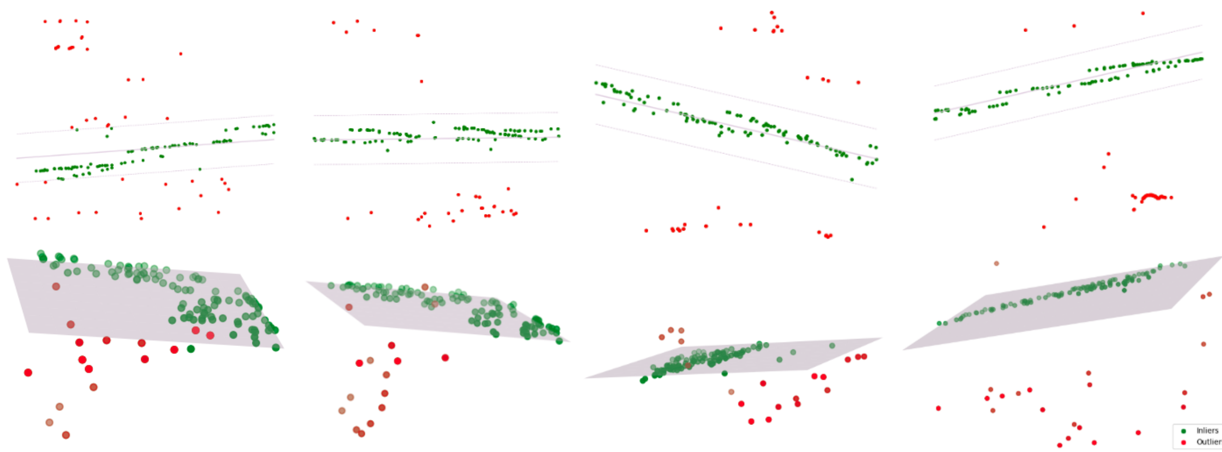

Figure A1: visualization of 2D Line Fitting and 3D Plane fitting on Real Data.

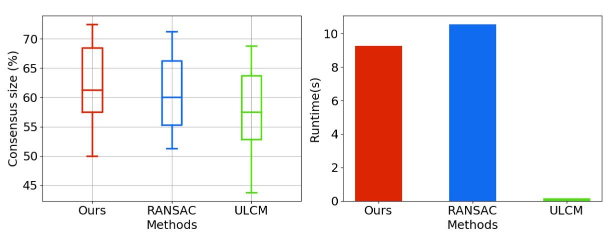

Figure A2: Homography ftting on KITTI dataset

Methods	Samson errors (mean)
Ours	0.17
RANSAC	0.18

Table 1: Samson error comparison between our method and RANSAC

4 Performance metric for Fundamental Matrix Estimation

To give readers a better understanding about the obtained solutions, real fundamental matrices are estimated by applying Least Square Fitting on the inlier sets. Table. 1 shows the comparison of Sampson error between our method and RANSAC. Our method achieved more inliers near the threshold, so that the error is slightly larger than RANSAC solution.

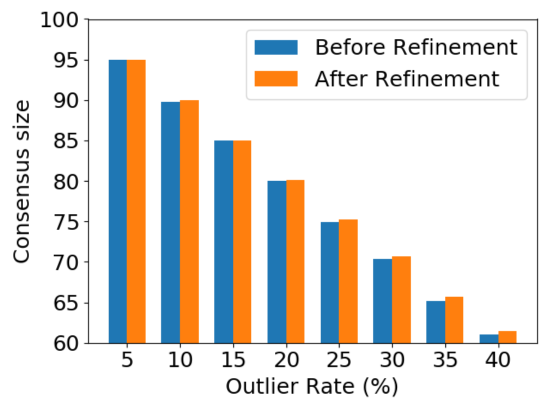

Figure A3: Consensus size of our proposed method, before and after refinement, in 2D line fitting experiment.

5 Implementation detail

Based on example implementation from Pytorch [2], we applied Huber loss when training. We use Adam optimizer with learning rate 0.001 and with batch size 32. The capacity of memory is set to 100,000.

References

[1] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012*

IEEE Conference on Computer Vision and Pattern Recognition, pages 3354–3361, 2012. 1

[2] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, pages 8026–8037, 2019. 2

[3] Jamie Shotton, Ben Glocker, and Christopher et al. Zach. Scene coordinate regression forests for camera relocalization in RGB-D images. In *CVPR*, 2013. 1