Supplementary Material of Learnable Motion Coherence for Correspondence Pruning

Yuan Liu[1] Lingjie Liu[2] Cheng Lin[1] Zhen Dong[3] Wenping Wang[1,4]
[1]The University of Hong Kong [2]MPI Informatics, Saarland Informatics Campus
[3]Wuhan University [4]Texas A&M University

1. Proof of Proposition 1

The problem is

$$\underset{\boldsymbol{s}}{minimize}\, \mathrm{Tr}((\boldsymbol{s}-\boldsymbol{m})^\intercal(\boldsymbol{s}-\boldsymbol{m})) + \eta\, \mathrm{Tr}(\boldsymbol{s}^\intercal \boldsymbol{L}\boldsymbol{s}), \quad (1)$$

where $\boldsymbol{m}_i = (m_{i,x}, m_{i,y})$ is an input motion and $\boldsymbol{L}$ is the Laplacian matrix of the graph. We have $\boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^\intercal$ where $\boldsymbol{\Lambda} = diag([\lambda_i])$ is a diagonal matrix of all eigenvalues λ_i and columns of $\boldsymbol{U}$ are eigenvectors. We will prove that the solution is given by $\boldsymbol{s} = \boldsymbol{U}diag([1/(1+\eta\lambda_i)])\boldsymbol{U}^\intercal\boldsymbol{m}$.

Proof. To simplify the discussion, let us only consider fitting the X-direction motion $\boldsymbol{m}_x = [m_{i,x}]$ first, which is,

$$\underset{\boldsymbol{s}_x}{minimize}\, (\boldsymbol{s}_x-\boldsymbol{m}_x)^\intercal(\boldsymbol{s}_x-\boldsymbol{m}_x) + \eta\boldsymbol{s}_x^\intercal\boldsymbol{L}\boldsymbol{s}_x. \quad (2)$$

Replacing $\boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^\intercal$, we have,

$$\underset{\boldsymbol{s}_x}{minimize}\, (\boldsymbol{s}_x-\boldsymbol{m}_x)^\intercal(\boldsymbol{s}_x-\boldsymbol{m}_x) + \eta(\boldsymbol{U}^\intercal\boldsymbol{s})^\intercal\boldsymbol{\Lambda}\boldsymbol{U}^\intercal\boldsymbol{s}_x. \quad (3)$$

Then, replacing $\boldsymbol{s}_x$ with $\boldsymbol{s}_x = \boldsymbol{U}\boldsymbol{t}$ where $\boldsymbol{t} \in \mathbb{R}^N$, we have,

$$\underset{\boldsymbol{t}}{minimize}\, (\boldsymbol{U}\boldsymbol{t}-\boldsymbol{m}_x)^\intercal(\boldsymbol{U}\boldsymbol{t}-\boldsymbol{x}_x) + \eta\boldsymbol{t}^\intercal\boldsymbol{\Lambda}\boldsymbol{t}. \quad (4)$$

Since $\boldsymbol{U}$ is orthogonal, we can further replace $\boldsymbol{m}_x$ with $\boldsymbol{U}\boldsymbol{U}^\intercal\boldsymbol{m}_x$ to get,

$$\underset{\boldsymbol{t}}{minimize}\, (\boldsymbol{t}-\boldsymbol{U}^\intercal\boldsymbol{m}_x)^\intercal(\boldsymbol{t}-\boldsymbol{U}^\intercal\boldsymbol{m}_x) + \eta\boldsymbol{t}^\intercal\boldsymbol{\Lambda}\boldsymbol{t}. \quad (5)$$

Merging two terms, we have,

$$\underset{\boldsymbol{t}}{minimize}\, \boldsymbol{t}^\intercal(\boldsymbol{I}+\eta\boldsymbol{\Lambda})\boldsymbol{t} - 2(\boldsymbol{U}^\intercal\boldsymbol{m}_x)^\intercal\boldsymbol{t} + C, \quad (6)$$

where C is a constant independent from $\boldsymbol{t}$. Obviously, the solution of $\boldsymbol{t}$ in Problem 6 is given by $(\boldsymbol{I}+\eta\boldsymbol{\Lambda})^{-1}\boldsymbol{U}^\intercal\boldsymbol{m}_x$. Then, we replace it back to get $\boldsymbol{s}_x = \boldsymbol{U}(\boldsymbol{I}+\eta\boldsymbol{\Lambda})^{-1}\boldsymbol{U}^\intercal\boldsymbol{m}_x = \boldsymbol{U}diag([1/(1+\eta\lambda_i)])\boldsymbol{U}^\intercal\boldsymbol{m}_x$. Similarly, the solution for the Y-direction motions $\boldsymbol{s}_y = \boldsymbol{U}diag([1/(1+\eta\lambda_i)])\boldsymbol{U}^\intercal\boldsymbol{m}_y$ following the same procedure. Combining both, we have $\boldsymbol{s} = \boldsymbol{U}diag([1/(1+\eta\lambda_i)])\boldsymbol{U}^\intercal\boldsymbol{m}$. □

2. Connections to Motion Coherence Theory

In Motion Coherence Theory [11], the smoothness regularization ϕ of a function $f(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}$ is,

$$\phi(f) = \int_{\mathbb{R}^d} \sum_l^\infty \frac{\beta^{2l}}{l!2^l}\|D^l f(\boldsymbol{x})\|^2 d\boldsymbol{x}, \quad (7)$$

where D is a derivative operator such that $D^{2l}f = \nabla^{2l}f$, $D^{2l+1} = \nabla(\nabla^{2l}f)$. ∇ is the gradient operator, and ∇^2 is the laplacian operator. According to [1], the Equation 7 is equivalent to the regularization term used in [6, 4, 3] as follows,

$$\psi(f) = \int_{\mathbb{R}^d} \frac{\|\tilde{f}(\boldsymbol{s})\|^2}{G(\boldsymbol{s})} d\boldsymbol{s}, \quad (8)$$

where $\tilde{f}$ is the Fourier transformation of f and G is a Gaussian function. Minimizing the term $\psi(f)$ forces f to be smooth by penalizing high-frequency components of f.

In the view of Graph Fourier Transformation [7], we show that the proposed LMF has a similar form as ψ of Equation 8, which also penalizes the high frequency components of motions. The solution to LMF is given by $\boldsymbol{U}diag([1/(1+\eta\lambda_i)])\boldsymbol{U}^\intercal\boldsymbol{m}$. First, $\boldsymbol{U}^\intercal\boldsymbol{m}$ is the Graph Fourier Transformation of $\boldsymbol{m}$, where different rows stand for signals in different frequencies. Then, high frequency components of $\boldsymbol{U}^\intercal\boldsymbol{m}$ is penalized by $1+\eta\lambda_i$. Finally, it is transformed back by left-multiplying $\boldsymbol{U}$, which is a Inverse Graph Fourier Transformation.

3. Network Details

Fig. 1 shows some details of each component. "IN" is an instance normalization layer. "BN" is a batch normalization layer. "FC" is a fully connected layer. "KNNDiff(8)" computes the difference between a feature with its neighboring features on the graph and "SmoothDiff" computes $\boldsymbol{f} - \boldsymbol{R}(\eta)\boldsymbol{f}$ according to the Proposition 1.

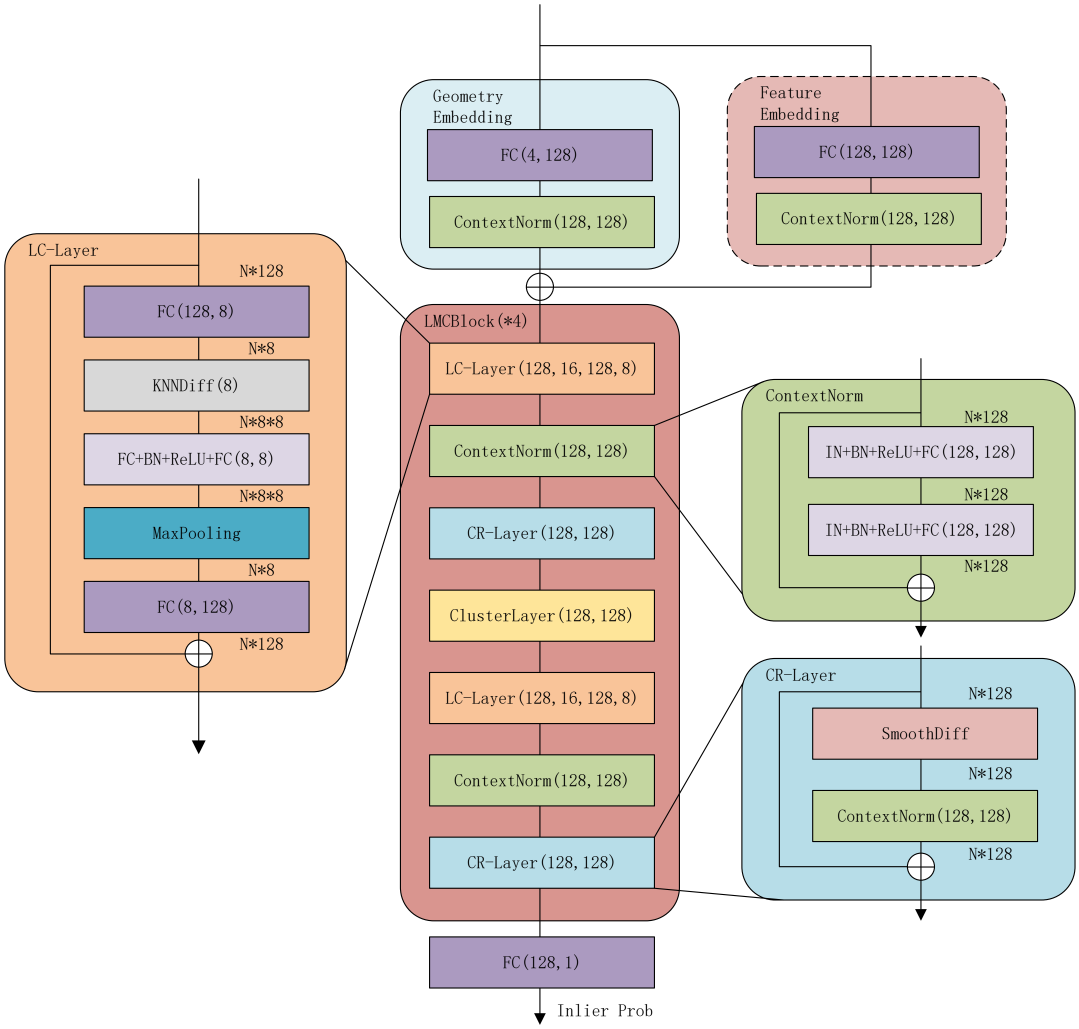

Figure 1. Detailed network architecture of LMCNet.

4. Training Details

We use Adam optimizer with 1e-3 as learning rate and the learning rate is frozen for 200k steps. After 200k steps, we halve the learning rate every 20k steps. The batch size is 8 in all experiments.

5. Qualitative Results on SUN3D [10]

We show more qualitative results of LMCNet, PointCN [5] and OANet [12] on the indoor SUN3D dataset in Fig. 2.

6. Qualitative Results on DAVIS [8]

We evaluate the propose model on the DAVIS [8] dataset which mainly consists of images containing a dynamic foreground object with non-rigid deformation. Some qualitative results are shown in Fig. 4. LMCNet is able to find coherent correspondences with non-rigid motions among noisy putative correspondences.

7. Compatibility with Learning-based Descriptor and Matcher

We have conducted experiments on the SUN3D dataset and the ScanNet dataset to show the compatibility of LMCNet with SuperPoint [2] descriptor and SuperGlue [9] matcher. We use the official pretrained models provided by SuperPoint and SuperGlue. Some qualitative results are shown in Fig. 3. On every image, we extract 1024 SuperPoint features and match them by SuperGlue between every image pair. However, we do not adopt the filtering strategy of SuperGlue but retain all correspondences as input to the LMCNet, which is shown by the Column 2 of Fig. 3. The results show that LMCNet is able to find much more dense coherent correspondences from all putative cor-

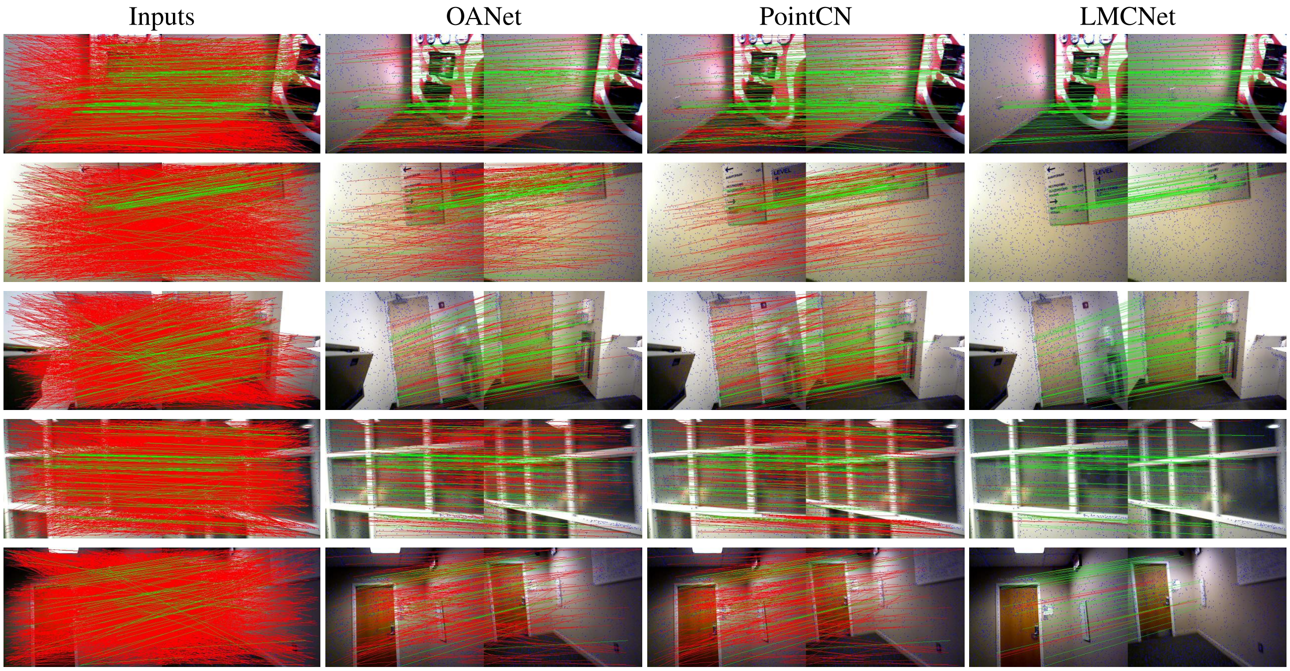

Figure 2. Qualitative results on the SUN3D [10] dataset.

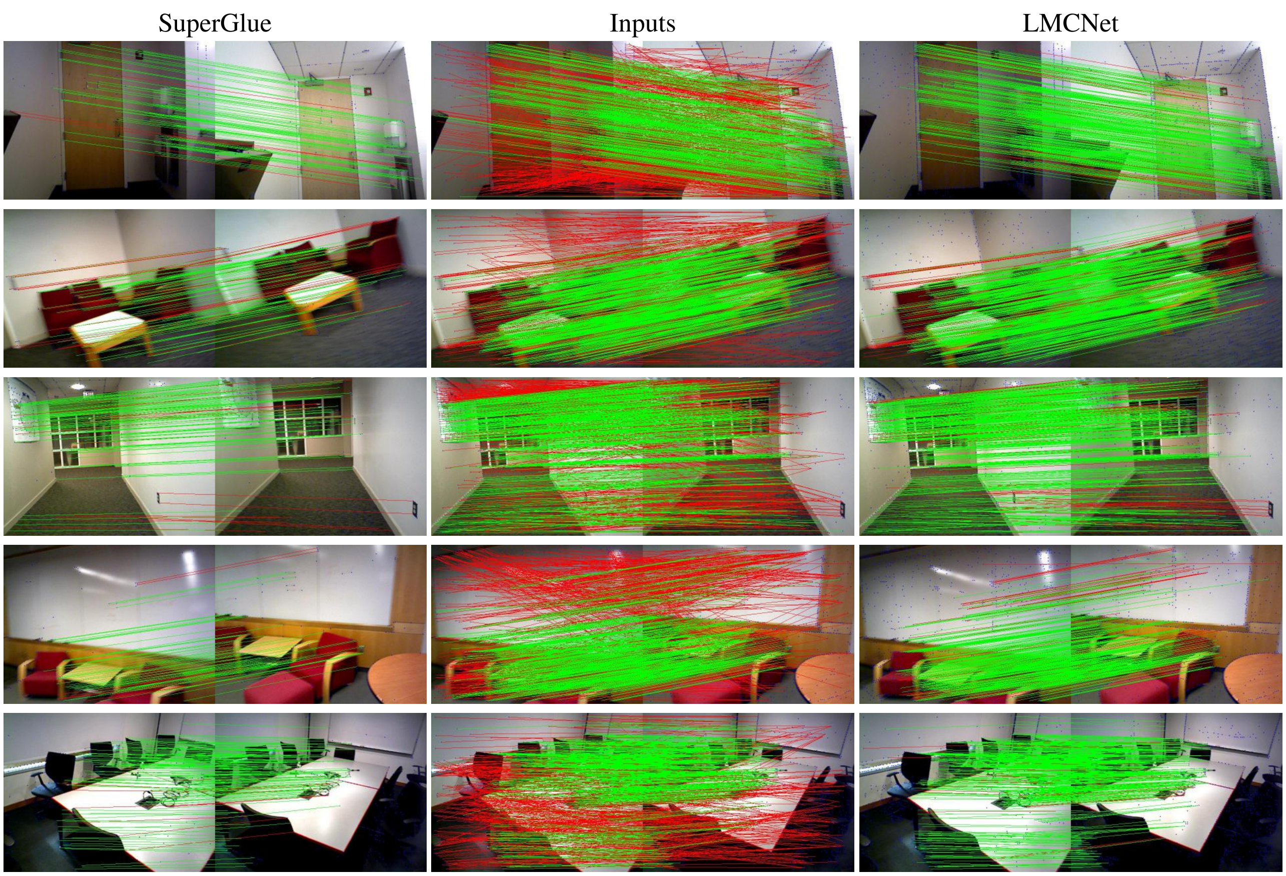

Figure 3. Results of LMCNet with putative correspondences produced by the SuperPoint [2] descriptor and SuperGlue [9] matcher.

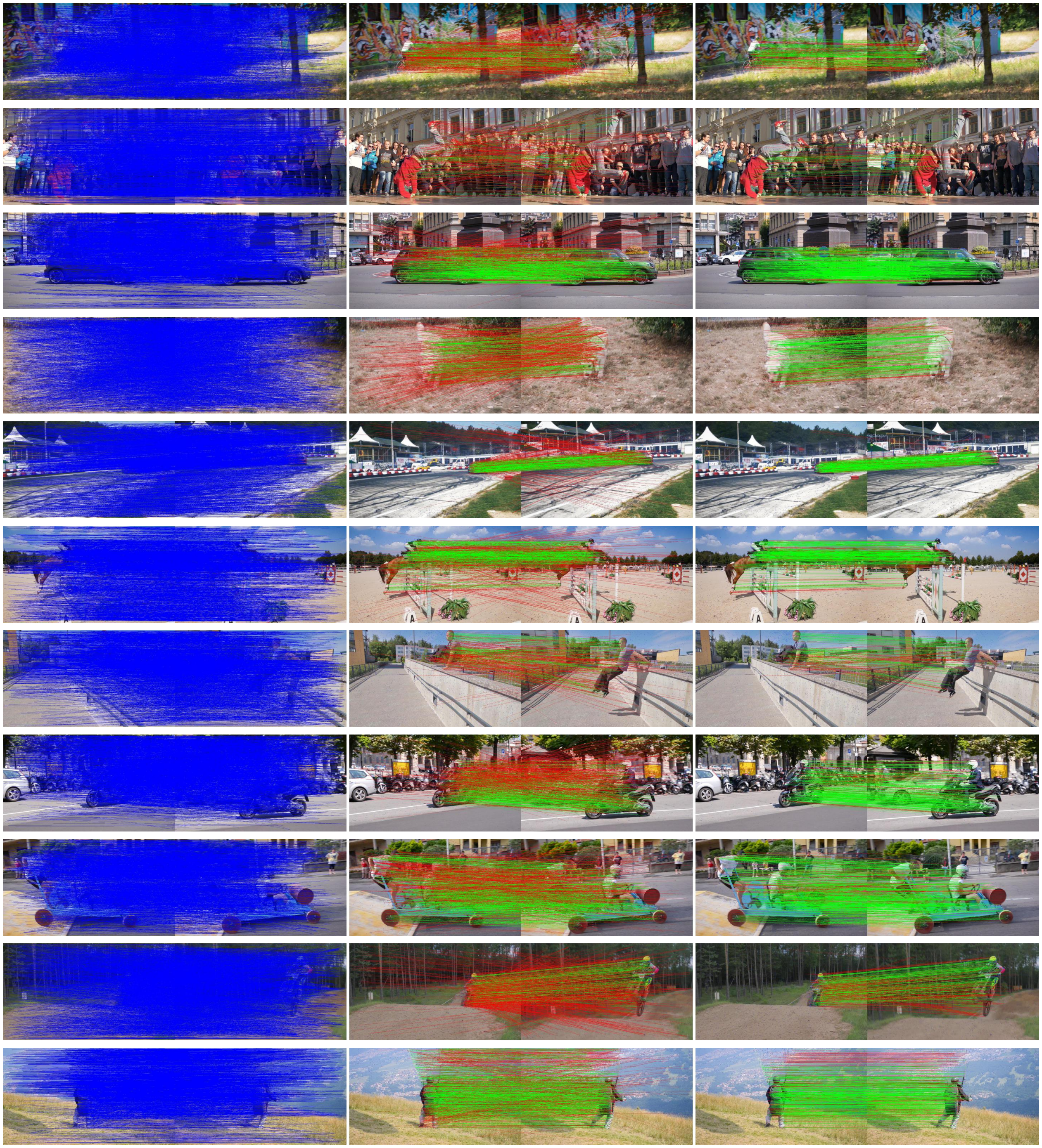

Figure 4. Qualitative results on the DAVIS [8] dataset. Column 1 shows the input correspondences which contain both foreground and background correspondences. Column 2 shows the pseudo ground-truth label for foreground correspondences. Green color represents true correspondences which connect the same instances between two frames while red color represents false correspondences. Column 3 shows the output foreground correspondences of LMCNet, where background correspondences are not drawn for clear visualization.

respondences than the default filtering strategy of SuperGlue. Improved performances may be achieved with other descriptors or matchers, which we leave in future works.

8. Motion Coherence

It is hard to exactly measure how much motion coherence property is learnt by a network. We conduct an experiment by manually adding random or systematic pertur-

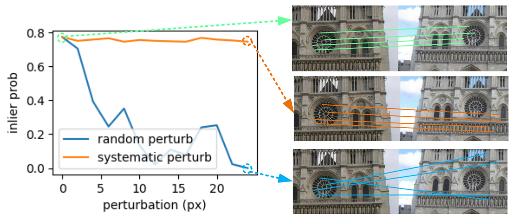

Figure 5. Predicted mean inlier probability under perturbations. Note all true correspondences are perturbed.

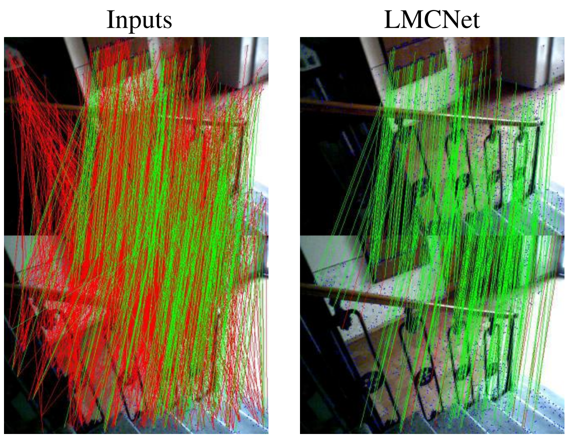

Figure 6. A typical example of piece-wise smooth motions caused by changes of depths.

bations to true correspondences, where random perturbations violate the motion coherence property while systematic ones do not. As shown in Fig. 5, the inlier probabilities predicted by LMCNet remain unchanged when coherence property keeps while they drop fast otherwise.

9. Piece-wise smooth motions

Since the graph is built on the "bilateral" correspondence space [3], LMCNet has the ability to detect piece-wise smooth coherent correspondences. Fig. 6 shows a typical example of piece-wise motions caused by varying depths in the SUN3D dataset.

References

[1] Zhe Chen and Simon Haykin. On different facets of regularization theory. *Neural Computation*, 14(12):2791–2846, 2002.

[2] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 224–236, 2018.

[3] Wen-Yan Lin, Fan Wang, Ming-Ming Cheng, Sai-Kit Yeung, Philip HS Torr, Minh N Do, and Jiangbo Lu. Code: Coherence based decision boundaries for feature correspondence. *Transactions on pattern analysis and machine intelligence*, 40(1):34–47, 2017.

[4] Wen-Yan Daniel Lin, Ming-Ming Cheng, Jiangbo Lu, Hongsheng Yang, Minh N Do, and Philip Torr. Bilateral functions for global motion modeling. In *ECCV*, 2014.

[5] Kwang Moo Yi, Eduard Trulls, Yuki Ono, Vincent Lepetit, Mathieu Salzmann, and Pascal Fua. Learning to find good correspondences. In *CVPR*, 2018.

[6] Andriy Myronenko, Xubo Song, and Miguel A Carreira-Perpinán. Non-rigid point set registration: Coherent point drift. In *NeurIPS*, 2007.

[7] Antonio Ortega, Pascal Frossard, Jelena Kovačević, José MF Moura, and Pierre Vandergheynst. Graph signal processing: Overview, challenges, and applications. *Proceedings of the IEEE*, 106(5):808–828, 2018.

[8] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *ArXiv*, 2017.

[9] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning feature matching with graph neural networks. In *CVPR*, 2020.

[10] Jianxiong Xiao, Andrew Owens, and Antonio Torralba. Sun3d: A database of big spaces reconstructed using sfm and object labels. In *ICCV*, 2013.

[11] Alan L Yuille and Norberto M Grzywacz. A mathematical analysis of the motion coherence theory. *International Journal of Computer Vision*, 3(2):155–175, 1989.

[12] Jiahui Zhang, Dawei Sun, Zixin Luo, Anbang Yao, Lei Zhou, Tianwei Shen, Yurong Chen, Long Quan, and Hongen Liao. Learning two-view correspondences and geometry using order-aware network. In *CVPR*, 2019.