DGC-GNN: Leveraging Geometry and Color Cues for Visual Descriptor-Free 2D-3D Matching

Supplementary Material

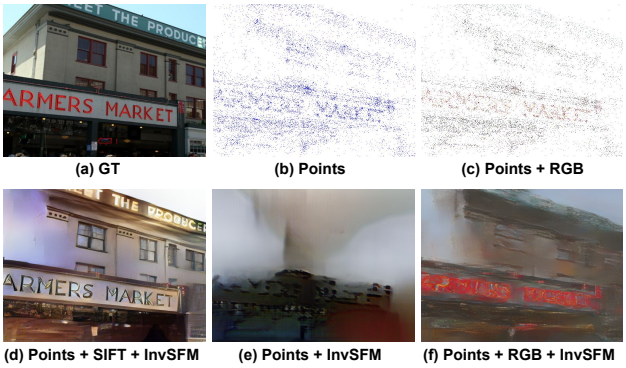


Figure 1. **Points Reprojection and Image Recovery Example**.

1. Training and Evaluation Details

Dataset Generation. The training data generation process for MegaDepth [7] follows the methodology outlined in GoMatch [13]. The undistorted SfM model reconstructions used in MegaDepth are provided by D2Net [4]. For training, we sample up to 500 images from each scene. For each sampled image, we select the top-k co-visible views that have at least 35% image overlap. This ensures that there are enough matches for training. The overlapping score is computed by dividing the number of co-visible 3D points by the total number of points in the training image.

In the case of ScanNet [3], a similar procedure is conducted. We also sample up to 500 images from each scene for the training set generation. The co-visible images are obtained using the co-visible scores provided by LoFTR [12]. We extract all the co-visible views of a training image with co-visible scores larger than 0.65. Then, we randomly sample the top-k views for training. Since ScanNet is an RGB-D dataset without an SfM reconstruction, we obtain the 3D points for each image by projecting the detected 2D keypoints with valid depth to 3D. By doing this for each image, we reconstruct a sparse 3D point cloud based on the detected 2D keypoints. Note that the correspondence between different co-visible frames is not required in this case.

In total, for MegaDepth, we generate a training set consisting of 25,624 images from 99 scenes and a test set comprising 12,399 images covering 53 scenes. For ScanNet, we create a training set with 52,008 images from 105 scenes. The test set for ScanNet consists of 14,892 query images from 30 scenes. The data generation of 7Scenes [10] and Cambridge dataset [6] follows the same procedure in [13].

Inference. We consider a query with at least 10 keypoints as valid input. The 3D points from the top-k retrieved database images are then applied to match against the queries with our proposed pipeline. We use the Sinkhorn algorithm [2, 11] to optimize the extended cost matrix $\bar{\mathcal{M}} \in \mathbb{R}^{N+1,M+1}$ in an iterative manner with up to 20 iterations to obtain the initial matches. The final matches are obtained by filtering the matches with matching confidence $\theta < 0.5$ in the outlier rejection module. For the visual localization task, the camera poses are estimated by the P3P solver with RANSAC [5] implemented in OpenCV [1] and then refined by Levenberg-Marquardt [8] algorithm on the inliers matches, minimizing the reprojection error.

Method	Reprojection		InvSFM [9]		
	Points	Points+RGB	Points	Points+RGB	Points+SIFT
SSIM (↓)	**0.240**	0.258	**0.352**	0.375	0.476

Table 1. **SSIM Results.** We evaluate the SSIM from Point Reprojection and Image Recovering, adding RGB to points leads only to a slight SSIM increase on both reprojection and image recovery.

2. Privacy Issue of RGB Points

We investigate the impact on privacy resulting from incorporating RGB information into pixels and points. To assess this, we compute the Structural Similarity Index Measure (SSIM) for 3D points reprojected onto the image plane against the ground truth (GT) images on MegaDepth over 500 images from multiple scenes. Additionally, we recover the images from points + RGB and points + descriptors with InvSFM [9] to calculate the SSIM against the GTs. The findings are detailed in Table 1 and Fig 1. The addition of RGB data to points results in only a marginal increase in SSIM for both direct reprojection and image reconstruction via InvSFM, significantly less than what is achieved by incorporating SIFT descriptors. It is worth noting that denser point clouds might provide sufficient context, potentially leading to privacy concerns. However, in our setting, we mitigate this risk by limiting the number of keypoints from each database image to a maximum of 1024.

3. Additional Results

Qualitative Results. More visualizations of inlier matches provided by DGC-GNN and GoMatch on MegaDepth are shown in Fig. 2. DGC-GNN consistently finds more correct matches on multiple scenes, highlighting the effectiveness of the proposed method.

Additional Ablation Results. In addition to the ablation re-

Methods	Global	C. Att.	Color	Ang.	Cluster	Reproj. AUC (%) @1 / 5 / 10px (↑)	Rotation (°) Quantile@25 / 50 / 75% (↓)	Translation
GoMatch [13] (w/o OR)						4.47 / 17.95 / 23.42	1.29 / 11.85 / 33.60	0.11 / 1.18 / 3.58
GoMatch [13]						5.67 / 22.43 / 28.01	0.60 / 10.08 / 34.63	0.06 / 1.06 / 3.73
Variants	G.Emb				K-means	7.68 / 28.41 / 34.36	0.28 / 6.78 / 34.52	0.03 / 0.73 / 3.77
	G.Label	✓			K-means	7.13 / 27.33 / 33.18	0.31 / 7.34 / 33.63	0.03 / 0.76 / 3.64
	G.Emb	✓			K-means	8.10 / 30.64 / 37.07	0.24 / 4.48 / 34.30	0.03 / 0.63 / 3.51
	G.Emb	✓	✓		K-means	9.82 / 35.29 / 41.16	0.17 / 2.88 / 31.74	0.02 / 0.27 / 3.24
	G.Emb	✓	✓	✓	Mean-shift	10.07 / 36.01 / 43.03	0.16 / 2.15 / 28.99	**0.01** / 0.20 / 3.26
DGC-GNN (w/o OR)	G.Emb	✓	✓	✓	K-means	8.56 / 30.79 / 37.03	0.22 / 4.85 / 30.07	0.02 / 0.47 / 3.10
DGC-GNN	G.Emb	✓	✓	✓	K-means	**10.20 / 37.64 / 44.04**	**0.15 / 1.53 / 27.93**	**0.01 / 0.15 / 3.00**

Table 2. **Additional Ablation Results.** AUC scores thresholded at 1, 5, and 10 pixels on $k = 1$; rotation and translation error quantiles at 25, 50, 75% with the proposed components added one by one to the GoMatch pipeline.

sults presented in the main paper, we also provide ablation results for single-view matching with $k = 1$ on MegaDepth [7]. Furthermore, we conduct two additional ablations to investigate the impact of different component selections. Firstly, we compare the effectiveness of the geometric global embedding (G. Emb.) used in the main paper with the global clustering label embedding (G. Label). Instead of encoding geometric cues, we encode the label of each global cluster and concatenate it to the local point feature. Then, we explore the selection of different clustering algorithms. We compare the performance of K-Means and Mean-Shift clustering algorithms in our pipeline. Last, we study the effectiveness of the outlier rejection (OR) network.

The results are presented in Table 2. We observe similar conclusions for each component as in the main paper. The results obtained using the global label embedding (G. Label) with cluster attention (C.Att) show even worse performance compared to geometric embedding (G. Emb.) only, indicating the superiority of our clustering-based geometric embedding over the label embedding and highlighting the importance of incorporating geometric cues in the embedding process for effective point matching. Regarding the impact of different clustering algorithms, we only observe a minor difference in K-Means and Mean-Shift results, suggesting that our approach is robust to the choice of the clustering algorithm. The results also demonstrate that outlier rejection is an essential post-processing module to achieve good performance. In addition to the numerical results, we visualize the inlier matches (see Fig. 3) to provide deep insights into the behavior and performance of different architectures.

Hyperparameters analysis. Besides the component ablations, we also give an in-depth analysis of the hyperparameters used in our main pipeline. Here, we add additional ablations on the number of input keypoints, the number of cluster groups at the coarse level, the number of nearest neighbors in the local graph build, and the outlier rejection threshold by retraining our DGC-GNN. The results are presented in Table 3. We observe that DGC-GNN with G. Clusters = 10 and Local NN = 10 achieves overall the best performance. Setting the outlier rejection threshold to 0.7 leads to the best performance. However, the results are stable across different configurations, indicating robustness to the parameter setting.

Matching Results in pixel threshold. As mentioned in the main paper, we selected the ground truth matches in normalized image coordinates. The described GT difference only affects the reprojection AUC scores. Here, we present the matching results in Table 1 by selecting the ground truth matches in pixel coordinate with 1 pixel threshold as done in [13]. Our conclusions still hold.

4. Model Parameters and Timing

We discuss the model parameters and running time of DGC-GNN in this section. DGC-GNN incorporates global geometric embedding and local clustering attention, which has around 5.7 million trainable parameters and an estimated model size of 22.6 MB. The average inference time for each image pair over the Megadepth evaluation queries is 77.8ms. It roughly breaks down into point encoding (24 ms), global geometric embedding (14 ms), cluster-based attention (22 ms), optimal transport (7 ms), and outlier rejection (8 ms). The measurements are conducted on a 32GB NVIDIA Telsa V100 GPU with a maximum of 1024 keypoints.

Methods	G. Cluters	Local NN	OR Threshold	Reproj. AUC (%) @1 / 5 / 10px (↑)	Rotation (°) Quantile@25 / 50 / 75% (↓)	Translation
DGC-GNN	10	10	0.5	15.30 / 51.70 / 60.01	**0.07** / 0.26 /	5.41 **0.01** / **0.02** / 0.57
HyperParam.	5	10	0.5	14.73 / 50.12 / 58.26	0.08 / 0.28 /	8.76 **0.01** / 0.03 / 0.99
	15	10	0.5	15.14 / 50.56 / 58.62	**0.07** / 0.28 /	7.66 **0.01** / 0.03 / 0.89
	10	20	0.5	14.77 / 49.84 / 57.97	**0.07** / 0.29 /	8.26 **0.01** / 0.03 / 0.90
	10	30	0.5	14.75 / 50.95 / 59.45	0.08 / 0.28 /	5.48 **0.01** / 0.03 / 0.58
	10	10	0.3	13.28 / 46.44 / 55.05	0.08 / 0.43 /	8.63 **0.01** / 0.04 / 0.98
	10	10	0.7	**16.63 / 56.26 / 64.46**	**0.07 / 0.19 /**	**2.58** **0.01 / 0.02 / 0.27**

Table 3. **Ablation Study on Hyperparameters.** We report the results of ablations with retrieved image $k = 10$ on the number of global clusters, the number of nearest neighbour points for local graph build, and different thresholds for outlier rejection. The best results are bold.

References

[1] Gary Bradski. The opencv library. *Software Tools for the Professional Programmer*, 25(11):120–123, 2000. 1

[2] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013. 1

[3] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5828–5839, 2017. 1

[4] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint description and detection of local features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8092–8101, 2019. 1

[5] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 1

[6] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE international conference on computer vision*, pages 2938–2946, 2015. 1

[7] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2041–2050, 2018. 1, 2

[8] Jorge J Moré. The levenberg-marquardt algorithm: implementation and theory. In *Numerical Analysis: Proceedings of the Biennial Conference*, pages 105–116. Springer, 2006. 1

[9] Francesco Pittaluga, Sanjeev J Koppal, Sing Bing Kang, and Sudipta N Sinha. Revealing scenes by inverting structure from motion reconstructions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 145–154, 2019. 1

[10] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2930–2937, 2013. 1

[11] Richard Sinkhorn and Paul Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21(2):343–348, 1967. 1

[12] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. LoFTR: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 1

[13] Qunjie Zhou, Sérgio Agostinho, Aljoša Ošep, and Laura Leal-Taixé. Is geometry enough for matching in visual localization? In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 407–425. Springer, 2022. 1, 2, 4

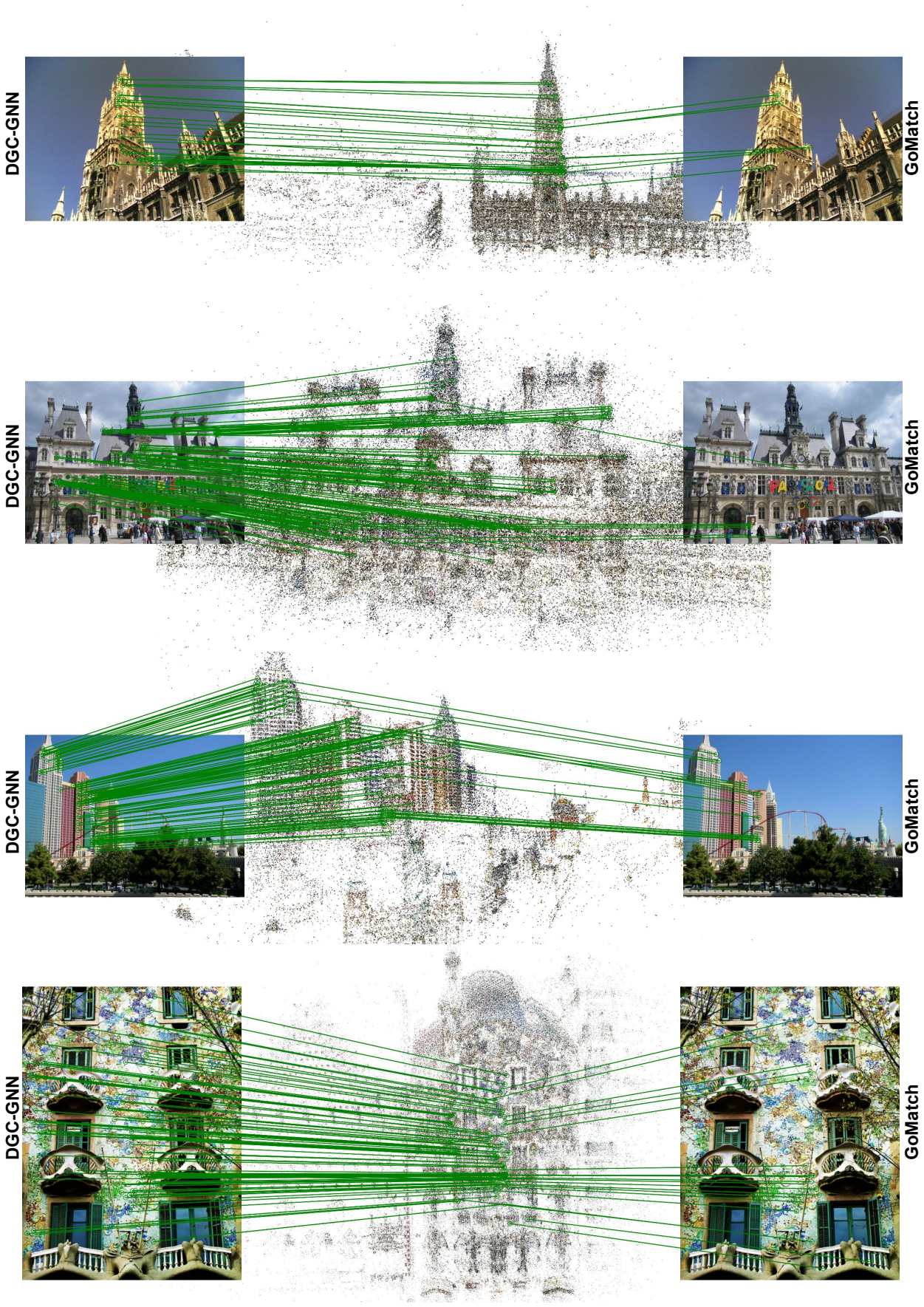
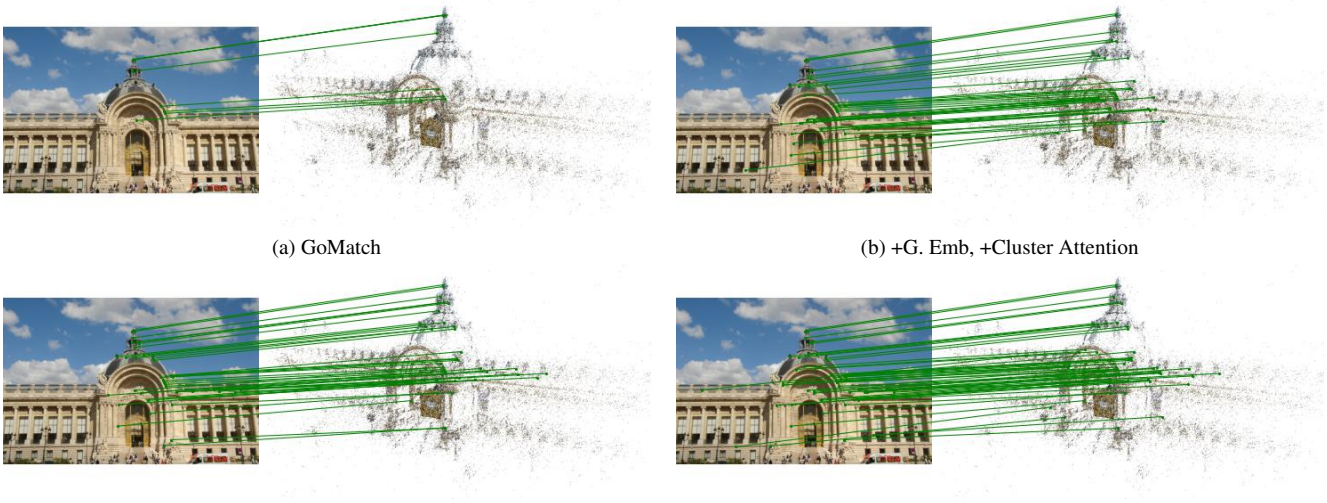


Figure 2. **2D-3D Matching** (shown by green lines) with the proposed DGC-GNN and GoMatch [13].



(a) GoMatch

(b) +G. Emb, +Cluster Attention

(c) +G. Emb, +Cluster Attention, +Color

(d) DGC-GNN

Figure 3. **Qualitative Matching Results of Different Architectures.** We visualize the number of inlier matches after the PnP-RANSAC with different architectures (shown by green lines).