Supplementary Material:
Joint Location and Orientation Estimation by Cross-View Matching

1. Localization with Unknown Orientation and Limited FoV

1.1. Location Estimation

In the main paper, we report the top-1, top-5, top-10 and top-1% recall rates of our algorithm and the state-of-the-art on localizing ground images with unknown orientation and varying FoVs. In this section, we present the complete r@K performance in Figure 1. It can be seen that our method achieves consistently better performance than the state-of-the-art algorithms in all the localization scenarios.

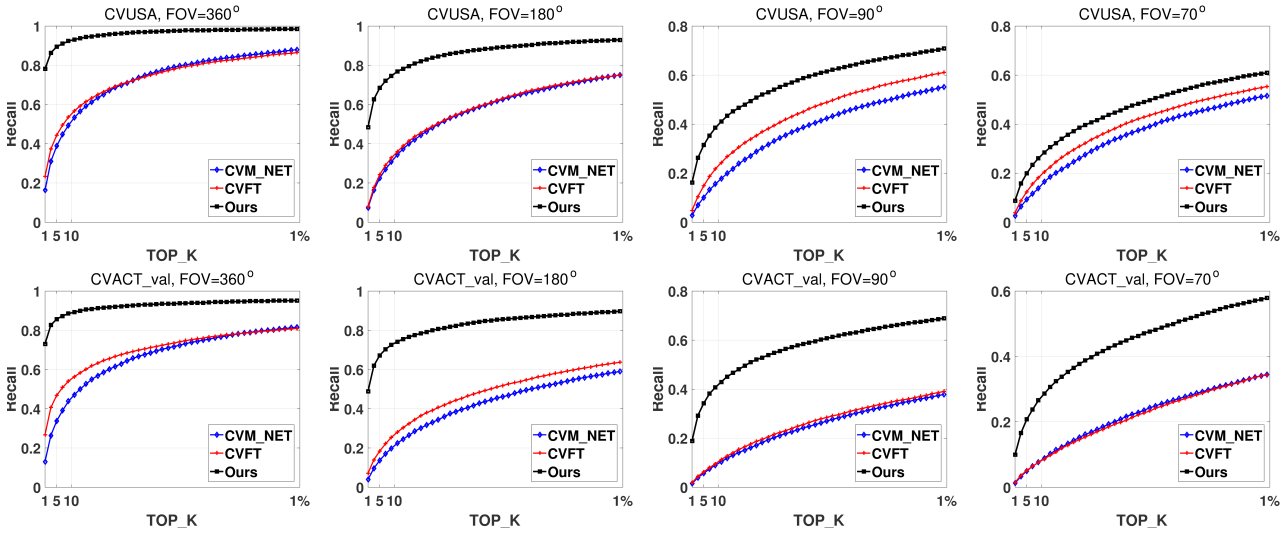


Figure 1. Location estimation performance (r@K) of different algorithms on unknown orientation and varying FoVs.

Training and Testing on Different FoVs: In real-world scenarios, a camera's FoV at inference time may not be the same as that used during training. Therefore, we also investigate the impact of using a model trained on a different FoV. We employ a model trained on ground images with a specific FoV and test its performance on ground images with varying FoVs. Figure 2 illustrates the recall curves at top-1, top-5, top-10 and top-1% with respect to different testing FoVs, and the numerical results are presented in Table 1.

It is apparent in Figure 2 that, as the test FoV increases, all models attain better performance. This implies that having a greater amount of scene contents reduces the matching ambiguity, and our method is able to exploit such information to achieve better localization. Furthermore, using a model trained with a FoV similar to the test image produces better results in general. Therefore, it is advisable to adopt a pretrained model with FoV similar to the test image.

1.2. Orientation Estimation

We provide additional visualization of estimating orientations for ground images with 360°, 180°, 90° and 70° FoV in Figure 4. As illustrated in Figure 4, our Dynamic Similarity Matching (DSM) module is able to estimate the orientation of ground images with varying FoVs.

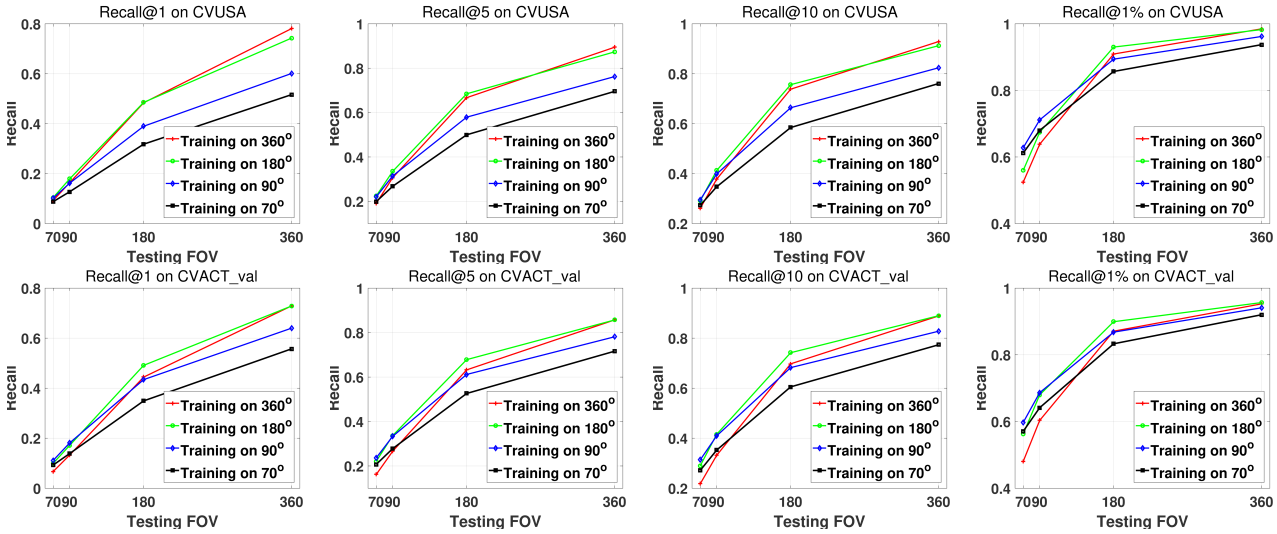


Figure 2. Recall performance at top-1, top-5, top-10 and top-1% of our method with different training and testing FoVs.

Table 1. Numerical recall results of our method with different training and testing FoVs.

Train FoV \ Test FoV		360°				180°				90°				70°			
		r@1	r@5	r@10	r@1%	r@1	r@5	r@10	r@1%	r@1	r@5	r@10	r@1%	r@1	r@5	r@10	r@1%
CVUSA	360°	78.11	89.46	92.90	98.50	48.38	66.73	73.82	90.93	16.36	30.67	37.78	63.82	9.26	19.82	26.08	52.45
	180°	74.30	87.37	91.25	98.27	48.53	68.47	75.63	93.02	17.91	33.51	41.24	67.40	10.42	22.43	29.06	55.98
	90°	60.13	76.23	82.40	96.21	38.98	57.94	66.38	89.39	16.19	31.44	39.85	71.13	10.15	22.02	29.50	62.78
	70°	51.65	69.56	76.09	93.75	31.79	49.99	58.45	85.69	12.74	26.78	34.73	68.00	8.78	19.90	27.30	61.20
CVACT_val	360°	72.91	85.70	88.88	95.28	44.43	63.23	69.73	87.09	13.26	26.62	33.14	60.33	6.70	16.18	21.77	48.05
	180°	72.87	85.68	88.97	95.67	49.12	67.83	74.18	89.93	17.13	33.68	41.55	67.99	10.00	22.10	28.97	56.26
	90°	64.00	78.11	82.77	94.10	43.42	61.17	68.22	86.80	18.11	33.34	40.94	68.05	11.14	23.65	31.34	59.73
	70°	55.73	71.63	77.35	92.02	34.92	52.62	60.52	83.31	13.86	27.81	35.24	64.14	9.29	20.72	27.13	57.08

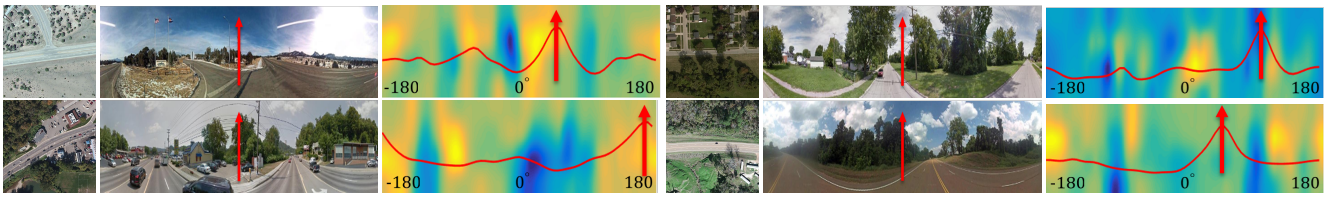
Figure 3. Examples of symmetric scenes (aerial images). At these locations, it is hard to determine the orientation (azimuth angle) of a ground image if it only contains a small sector of the aerial image.

When a camera has a small FoV, it suffers high ambiguity to determine the orientation by matching the ground image to its corresponding aerial one. We illustrate an example in Figure 4(c) where the error of orientation estimation for a road can be 180°. As seen in the first instance of Figure 4(c), the road occupies a large portion of the ground image. While in the aerial image, the road is symmetric in respect to the image center (*i.e.*, camera location). Thus there are two peaks in the similarity curve.

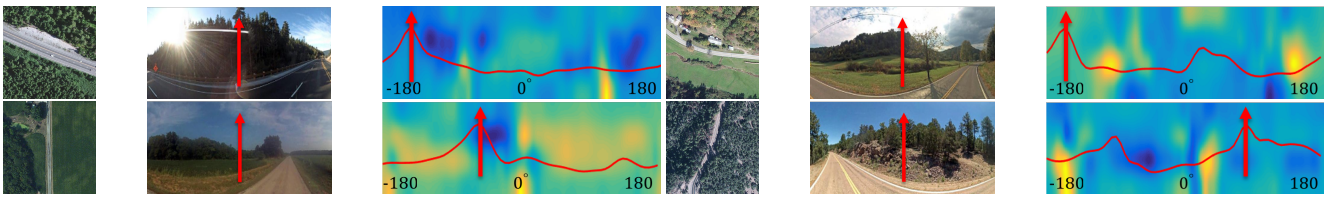
If the peak on the left is taller than the peak on the right, the estimated orientation will be wrong. Figure 3 provides another three examples where scene contents are similar in multiple directions. At these locations, it is hard to determine the orientation of a ground camera which has a small FoV while the estimated location is correct.

2. Time Efficiency

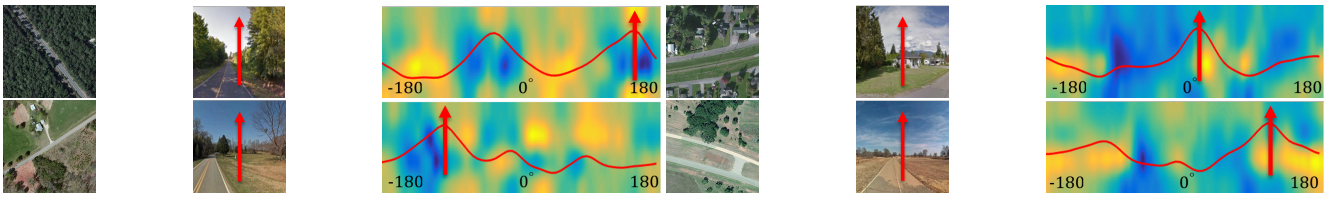
In order to improve the time efficiency, we compute the correlation in our DSM module by using Fast Fourier Transform during the inference process. To be specific, we store the Fourier coefficients of aerial features in the database, and calculate the Fourier coefficients of the ground feature in the forward pass. By doing so, the computation flops of the correlation



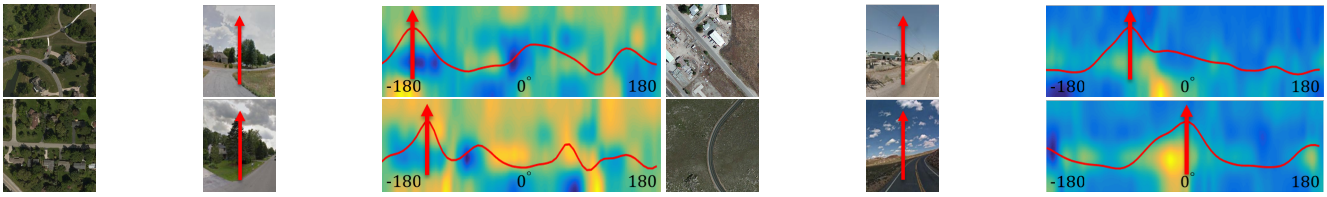
(a) FoV=360°



(b) FoV=180°



(c) FoV=90°



(d) FoV=70°

Figure 4. Visualization of estimated orientations for ground images with varying FoVs. For each ground and aerial pair, we visualize their similarity scores at different azimuth angles as a red curve on the polar-transformed aerial features. As indicated by the arrows on the similarity curves, the positions of similarity maxima in the curves correspond to the orientation of ground images.

are $13NHWC$ (including $4NHWC$ flops for coefficients multiplication in the spectral domain, and $1.5NHCW \log_2 W$ flops for the inverse Fast Fourier Transform), where H, W and C is the height, width and channel number of the global feature descriptor of an aerial image, N is the number of database aerial images, and $W = 64$ in our method. In contrast to conducting correlation in the spatial domain where the computation flops are $2NHW^2C$, the computation time is reduced by a factor of $\frac{1}{10}$ ($\frac{13NHWC}{2NHW^2C} \approx \frac{1}{10}$).

We conduct the retrieval process of a query image on a 3.70 GHz i7 CPU system, and the codes are implemented in Python3.6. For a ground panorama with unknown orientation, it takes an average time of $0.15s$ for retrieving its aerial counterpart from a database containing 8884 reference images. This demonstrates the efficiency of the proposed algorithm.

3. Trainable Parameters

Since the authors of CVM-NET [1], Liu & Li [2] and CVFT [3] provide the source code of their works, we compare the trainable parameters and model size of our network with their methods in Table 2. Our network not only outperforms the state-of-the-art but also is more compact, facilitating the deployment of our network.

Table 2. Comparison of trainable parameters and model size with recent methods.

Methods	# Parameters	Model size
CVM-NET [1]	160,311,424	1.8G
Liu & Li [2]	30,695,808	369.6M
CVFT [3]	26,814,657	336.8M
Ours	14,472,864	244.8M

References

[1] Sixing Hu, Mengdan Feng, Rang M. H. Nguyen, and Gim Hee Lee. Cvm-net: Cross-view matching network for image-based ground-to-aerial geo-localization. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3, 4

[2] Liu Liu and Hongdong Li. Lending orientation to neural networks for cross-view geo-localization. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3, 4

[3] Yujiao Shi, Xin Yu, Liu Liu, Tong Zhang, and Hongdong Li. Optimal feature transport for cross-view image geo-localization. *arXiv preprint arXiv:1907.05021*, 2019. 3, 4