Supplementary Material for LocalTrans: A Multiscale Local Transformer Network for Cross-Resolution Homography Estimation

Ruizhi Shao[1,*], Gaochang Wu[2,*], Yuemei Zhou[1], Ying Fu[3], Lu Fang[1], and Yebin Liu[1]

[1]Tsinghua University [2]Northeastern University [3]Beijing Institute of Technology

In this supplementary material, we provide the details of network architecture (Sec. 1.1), training details ((Sec. 1.3)), theoretical analysis (Sec. 2) additional ablation studies (Sec. 3) and more visual evaluations (Sec. 4) cross-resolution setting. Please also refer to the supplementary video for cross-resolution results on multiscale gigapixel photography. We will release our training and testing codes.

1. Implementation Details

In this section, we introduce the implementation details of the proposed LocalTrans, including network architecture (deep siamese network and homography estimation module), data preprocessing and training details.

1.1. Network Architecture

Table 1 shows the detail configuration of the deep siamese network in scale-level $k = 1$. For the rest scale-levels ($k = 2$ and $k = 3$), the output layers are Maxpool3 and Maxpool2, respectively. Table 2 shows the detail configuration of the homography estimation module in scale-level k. Note that the input of the average-pooling layer is ReLU3_2 in scale-level 1, and ReLU4_3 in scale-level 2, etc.

1.2. Data Preprocessing

We train our network using the MS-COCO 2014 dataset [5] and follow the same way with official setting to split dataset into Train/Test/Val. For the data augmentation, we adopt the same processes of adding Gaussian noise and randomly adjusting brightness, saturation and contrast, as those in [1]. Gaussian noise is added with standard deviation 0.02 to both of the two input images. And we randomly pick one image from the two input images to enhance brightness, saturation and contrast between 0.5 to 1.5 in a random order. These image operations are implemented by Torchvision [7]. For the synthesized cross-resolution setting, the low-resolution image is first downsampled and then upsampled to the original size using bicubic interpolation.

Layer	kernel	stride	channel	Input
Conv1_1	3	1	3/32	I
BN1_1	-	-	32	Conv1_1
ReLU1_1	-	-	-	BN1_1
Conv1_2	3	1	32/32	ReLU1_1
BN1_2	-	-	32	Conv1_2
ReLU1_2	-	-	-	BN1_2
Maxpool1	2	2	-	ReLU1_2
Conv2_1	3	1	32/64	Maxpool1
BN2_1	-	-	64	Conv2_1
ReLU2_1	-	-	-	BN2_1
Conv2_2	3	1	64/64	ReLU2_1
BN2_2	-	-	64	Conv2_2
ReLU2_2	-	-	-	BN2_2
Maxpool2	2	2	-	ReLU2_2
Conv3_1	3	1	64/64	Maxpool2
BN3_1	-	-	64	Conv3_1
ReLU3_1	-	-	-	BN3_1
Conv3_2	3	1	64/64	ReLU3_1
BN3_2	-	-	64	Conv3_2
ReLU3_2	-	-	-	BN3_2
Maxpool3	2	2	-	ReLU3_2
Conv4_1	3	1	64/128	Maxpool3
BN4_1	-	-	128	Conv4_1
ReLU4_1	-	-	-	BN4_1
Conv4_2	3	1	128/128	ReLU4_1
BN4_2	-	-	128	Conv4_2
ReLU4_2	-	-	-	BN4_2
Maxpool4	2	2	-	ReLU4_2

Table 1. Detail configuration of the deep siamese network for feature extraction in scale-level $k = 1$, where Conv denotes the 2D convolution layer, BN the batch normalization layer and Maxpool the max-pooling layer.

1.3. Training Details

We initialize the weights of both convolution and deconvolution layers by drawing randomly from a Gaussian

Layer	kernel	stride	channel	Input
Conv1_1	3	1	81/128	I
BN1_1	-	-	128	Conv1_1
ReLU1_1	-	-	-	BN1_1
Conv1_2	3	1	128/128	ReLU1_1
BN1_2	-	-	128	Conv1_2
ReLU1_2	-	-	-	BN1_2
Maxpool1	2	2	-	ReLU1_2
Conv2_1	3	1	128/256	Maxpool1
BN2_1	-	-	256	Conv2_1
ReLU2_1	-	-	-	BN2_1
Conv2_2	3	1	256/256	ReLU2_1
BN2_2	-	-	256	Conv2_2
ReLU2_2	-	-	-	BN2_2
Maxpool2	2	2	-	ReLU2_2
Conv3_1	3	1	256/256	Maxpool2
BN3_1	-	-	256	Conv3_1
ReLU3_1	-	-	-	BN3_1
Conv3_2	3	1	256/256	ReLU3_1
BN3_2	-	-	256	Conv3_2
ReLU3_2	-	-	-	BN3_2
Maxpool3	2	2	-	ReLU3_2
Conv4_1	3	1	256/256	Maxpool3
BN4_1	-	-	256	Conv4_1
ReLU4_1	-	-	-	BN4_1
Conv4_2	3	1	256/256	ReLU4_1
BN4_2	-	-	256	Conv4_2
ReLU4_2	-	-	-	BN4_2
Maxpool4	2	2	-	ReLU4_2
Conv5_1	3	1	256/256	Maxpool4
BN5_1	-	-	256	Conv5_1
ReLU5_1	-	-	-	BN5_1
Conv5_2	3	1	256/256	ReLU5_1
BN5_2	-	-	256	Conv5_2
ReLU5_2	-	-	-	BN5_2
Avgpool	-	-	-	ReLU$[k+2]$_2
Conv1D	1	1	256/8	Avgpool

Table 2. Detail configuration of the homography estimation module in scale-level k, where Avgpool denotes the average-pooling, Conv1D the 1D convolution layer.

distribution with a zero mean and a standard deviation of 1×10^{-3}, and the biases by zero. The network is optimized by using ADAM solver [3] with learning rate of 1×10^{-4} ($\beta_1 = 0.9$, $\beta_2 = 0.999$) and mini-batch size of 50. The network converges after 8×10^5 steps of backpropagation, taking about 35 hours on a NVIDIA Quadro GV100. The settings are the same for DHN[2], UDHN [9] and MHN [4], except the mini-batch size for UDHN [9] is 15.

2. Theoretical analyses

In this section, we provide additional theoretical analyses of LAK and the explicit formulation. First, the self-attention and the cross-attention achieved by the proposed LAK are able to highlight the feature, especially around the object boundary, as shown in Fig. 1. This characteristic ensures the better performance of the proposed LAK. Second, the explicit formulation is more interpretable. We can directly derive a sparse optical flow from the local attention map, as shown in Fig. 3, while the implicit formulations in [2, 4, 9] behave like a blackbox.

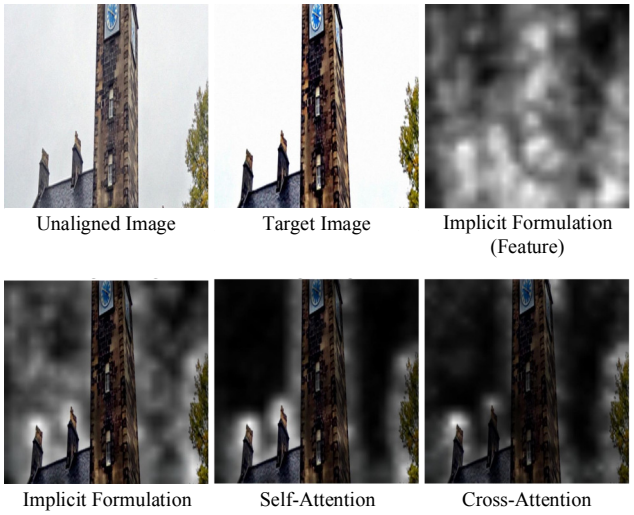

Figure 1. Feature map visualization. We visualize feature map by calculating the difference of one feature with its adjacent 4 features.

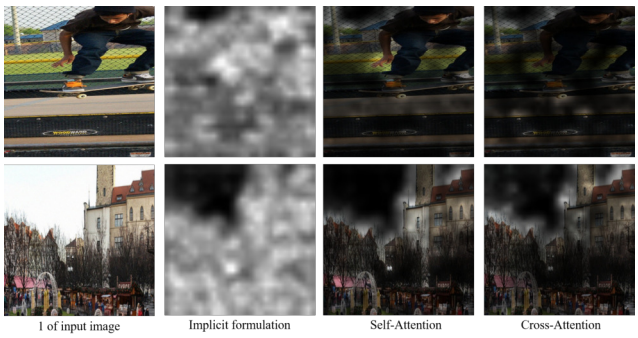

Figure 2. Additional feature map visualization.

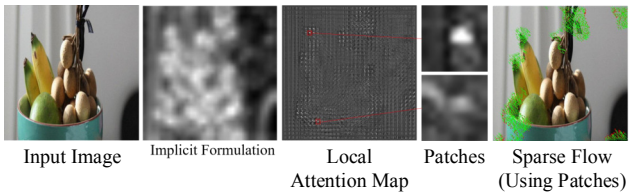

Figure 3. Comparison of the implicit and explicit formulation. For optical flow, red arrows are GT, green arrows are the derived sparse flow.

3. Additional Ablation Studies

In this section, we verify the effectiveness of the proposed LocalTrans by performing the following ablation

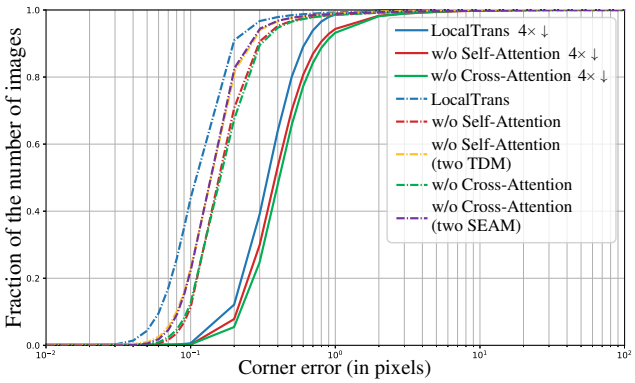

Figure 4. Comparison with the network without the self-attention (w/o Self-Attention) and without the cross-attention (w/o Cross-Attention).

studies.

Without self-attention. We implement this network by simply feeding the features from the deep siamese network into the Transformer Decoder Module (TDM). Since the Self-Attention Encoder Module (SAEM) hardly influences the depth or the receptive field of the network (it only contains two convolutional layers with kernel size of 1×1), we simply ignore the module. The result "w/o Self-Attention" in Fig. 4 shows that the performance of the network degrades obviously without the self-attention.

Without cross-attention. We implement this network by simply concatenating the features from the SEAM along the channel dimension and feed them into the homography estimation module. Similarly, this implementation will hardly influence the depth or the receptive field of the network. The result "w/o Cross-Attention" in Fig. 4 shows that the network performance significantly degrades without the cross-attention, which even worse than that without the self-attention.

Two identical attention modules. We evaluate our method using two SAEM only and two TDM only. The results are shown in Fig. 4. Two TDM perform slightly better than two SAEM but they still can not surpass our method, i.e., one SAEM and one TDM.

The above ablation studies validate that the self-attention and cross-attention in the proposed LocalTrans are superior than the concatenation of features from the input images.

4. Additional Results

4.1. Visual Evaluations on Synthesized Cross-Resolution Data

We show additional results on the MS-COCO Dataset [5] under cross-resolution settings ($4\times$ and $8\times$) in Fig. 5 and Fig. 6. We compare our LocalTrans with 4 baseline methods, a coventional feature-based method, SIFT+RANSAC [6], three deep learning-based methods, DHN [2], UDHN [9] and MHN [4]. The conventional feature-based method, SIFT+RANSAC [6], fails to estimate the homography matrices under high-resolution gaps. The proposed method has lower error and is more robust to large resolution gap compared with the deep learning-based methods, DHN [2], UDHN [9] and MHN [4].

4.2. Visual Evaluations on Optical Zoom-in Cross-Resolution Data

To ensure the color consistency among the high-resolution target images in each gigapixel scene, we adopt affine color mapping model introduced in [8] for color correction in the post-processing. Fig 7 shows additional results on the multiscale gigapixel dataset [8] under about $6\times$ resolution gap (top three cases) and the cross-resolution stereo dataset [10] under about $10\times$ resolution gap (bottom one case). The results demonstrate the superiority of the proposed LocalTrans. Please also refer to the supplementary video for cross-resolution results on multiscale gigapixel photography.

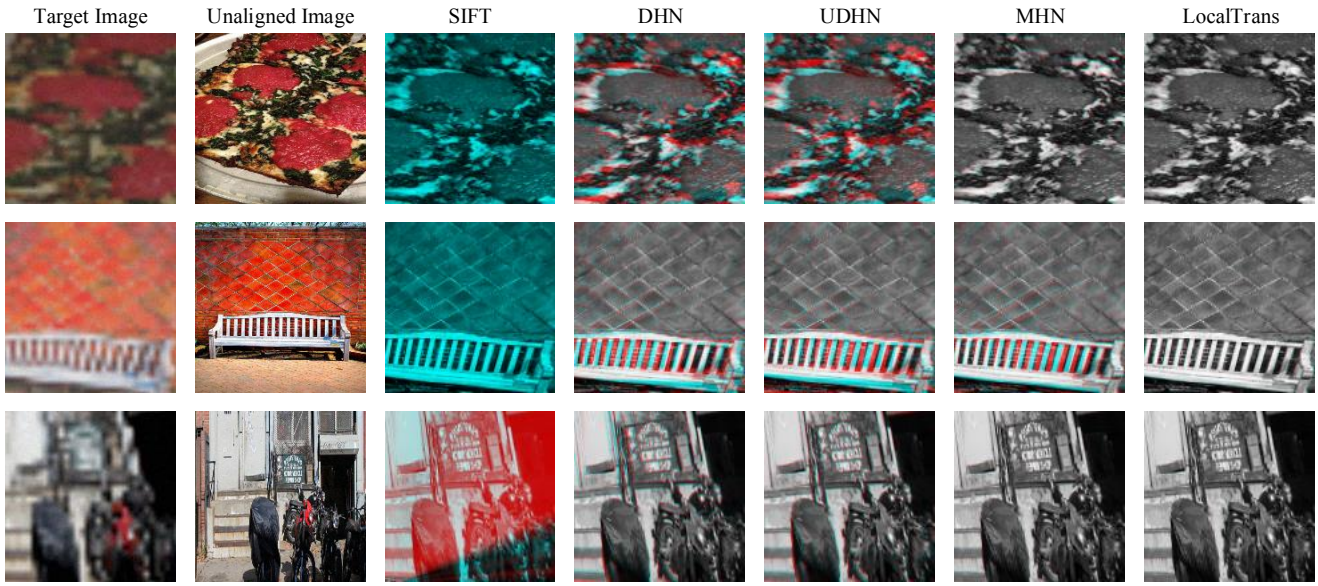

Figure 5. Visual evaluation on synthesized $4\times$ cross-resolution data. We convert the images to gray-scale, and mix the G channels of the aligned image and the R channel of the high-resolution target image. Note that the input target image is low-resolution. The misaligned pixels appear as red or green ghosts.

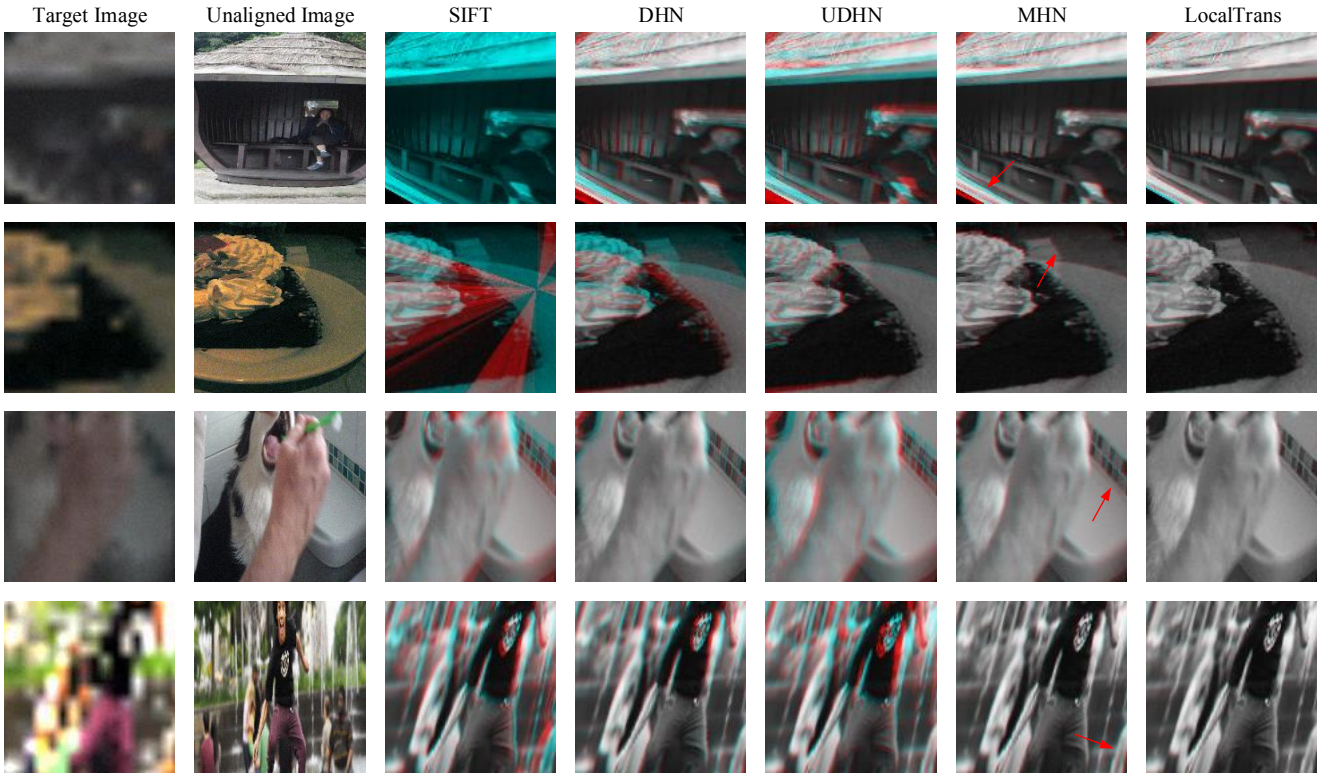

Figure 6. Visual evaluation on synthesized $8\times$ cross-resolution data.

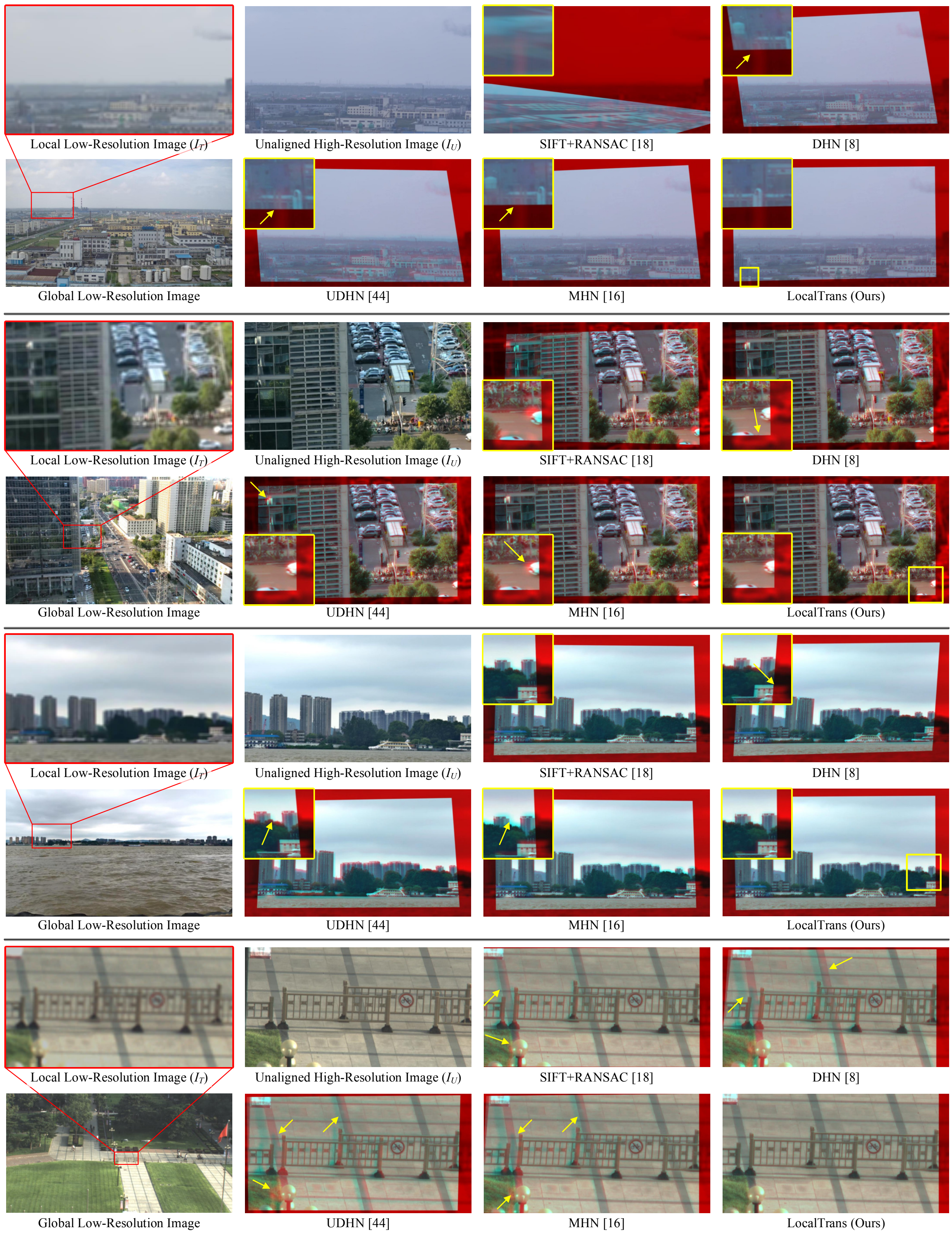

Figure 7. Visual evaluation on the multiscale gigapixel dataset [8] (top three, 6×) and the cross-resolution stereo dataset [10] (bottom, 10×).

References

[1] Che-Han Chang, Chun-Nan Chou, and Edward Y Chang. Clkn: Cascaded lucas-kanade networks for image alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2213–2221, 2017. 1

[2] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Deep image homography estimation. *arXiv preprint arXiv:1606.03798*, 2016. 2, 3

[3] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *Computer Science*, 2014. 2

[4] Hoang Le, Feng Liu, Shu Zhang, and Aseem Agarwala. Deep homography estimation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7652–7661, 2020. 2, 3

[5] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 1, 3

[6] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 3

[7] Sébastien Marcel and Yann Rodriguez. Torchvision the machine-vision package of torch. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1485–1488, 2010. 1

[8] Xiaoyun Yuan, Lu Fang, Qionghai Dai, David J Brady, and Yebin Liu. Multiscale gigapixel video: A cross resolution image matching and warping approach. In *2017 IEEE International Conference on Computational Photography (ICCP)*, pages 1–9. IEEE, 2017. 3, 5

[9] Jirong Zhang, Chuan Wang, Shuaicheng Liu, Lanpeng Jia, Nianjin Ye, Jue Wang, Ji Zhou, and Jian Sun. Content-aware unsupervised deep homography estimation. In *European Conference on Computer Vision*, 2020. 2, 3

[10] Yuemei Zhou, Gaochang Wu, Ying Fu, Kun Li, and Yebin Liu. Cross-mpi: Cross-scale stereo for image super-resolution using multiplane images. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3, 5