Joint Bilateral Learning for Real-time Universal Photorealistic Style Transfer - Supplementary Material

Xide Xia[1*], Meng Zhang[2*], Tianfan Xue[3], Zheng Sun[3], Hui Fang[3*], Brian Kulis[1], and Jiawen Chen[3]

[1] Boston University
{xidexia,bkulis}@bu.edu
[2] PixelShift.AI
meng@pixelshift.ai
[3] Google Research
{tianfan,zhengs,hfang,jiawen}@google.com

1 More Comparisons

1.1 Comparisons with Additional Baselines

We show additional comparisons with baseline photorealistic style transfer techniques including the optimization-based Deep Photo Style Transfer (DPST) by Li et al. [5], Neural Color Transfer (NCT) by He et al. [2], and HDRnet by Gharbi et al. [1], shown in Figures 1–3 respectively. For completeness, we also include result from PhotoWCT [4], LST [3], and WCT^2 [6].

Note that NCT relies on dense correspondence between the input style and content images. Occasionally, the matching algorithm can fail and NCT will generate a bad output, as shown in the second example in Figure 2.

1.2 High-resolution Image Style Transfer

To demonstrate the scalability and generalizability of our method to different resolutions, in Figures 4–12, we show additional high-resolution results. Notice how fine details are preserved. All results were generated with an affine bilateral grid prediction network at a fixed 256×256 input resolution, while rendering scales linearly with the resolution of the full-resolution input.

1.3 Comparisons on an Additional Test Set

From Figure 13–20, we show more qualitative comparisons on the test set we used for the user study described in the main paper.

* Work done while working at Google Research.

2 Video Photorealistic Style Transfer

We also evaluated our stylization network on video input, by processing each frame independently. Although our network is trained exclusively on images, it generalizes well to video input. Our method produces an output with a temporally consistent style despite no explicit temporal filtering or data augmentation. Please refer to `*.mp4` files in the `video_mp4` folder for those results.

3 Mobile Runtime

To achieve real-time performance at 4K on a mobile device, we implement a custom inference library in OpenCL and benchmark it on a Google Pixel 4 smartphone's Qualcomm Adreno 640 GPU. At 256×256 coefficient prediction resolution, 4032×3032 rendering resolution, and using 16-bit floating point, end-to-end runtime is a disappointing 4.4 seconds, which is dominated by computing VGG features. By reducing coefficient prediction resolution to 128×128 and using only 2 splatting blocks (removing `conv4_1` from VGG-19), runtime improves to 290 ms. Computational cost is still dominated by VGG features, taking 140 ms each, while coefficient prediction and rendering take 1.5 ms and 8.5 ms, respectively. Since we typically pick a single style and repeatedly use it over multiple content images (e.g., the camera viewfinder), we can compute style VGG features only once, further reducing runtime to 150 ms. Finally, by quantizing the content VGG network to use 8-bit integers and using the Pixel Neural Core, we are able to run that subgraph in only 1.4 ms. With significant engineering effort, we are able to achieve an end-to-end runtime of 12 ms per frame.

4 User Study

The included `eccv_main.m` is a MATLAB script that processes all the rating scores given by our 20 raters. In the "ratings" matrix, there are 240 rows and 20 columns. Each column represents one content/style combination. The rows are lexically ordered by `[rater's ID, category, algorithm]`, where there are 20 raters, 3 categories (Photorealism, Stylization, Overall quality), and 4 algorithms (PhotoWCT, WCT^2, LST, and Ours).

For example, the first 12 rows represent scores for:

- Rater0_Photorealism_PhotoWCT,
- Rater0_Photorealism_WCT^2,
- Rater0_Photorealism_LST,
- Rater0_Photorealism_Ours,
- Rater0_Stylization_PhotoWCT,
- Rater0_Stylization_WCT^2,
- Rater0_Stylization_LST,
- Rater0_Stylization_Ours,

- Rater0_Overall_PhotoWCT,
- Rater0_Overall_WCT^2,
- Rater0_Overall_LST,
- Rater0_Overall_Ours.

The script generates mean rating scores for each algorithm under each category, the same as we show in the user study table (Figure 8(c) in the main paper).

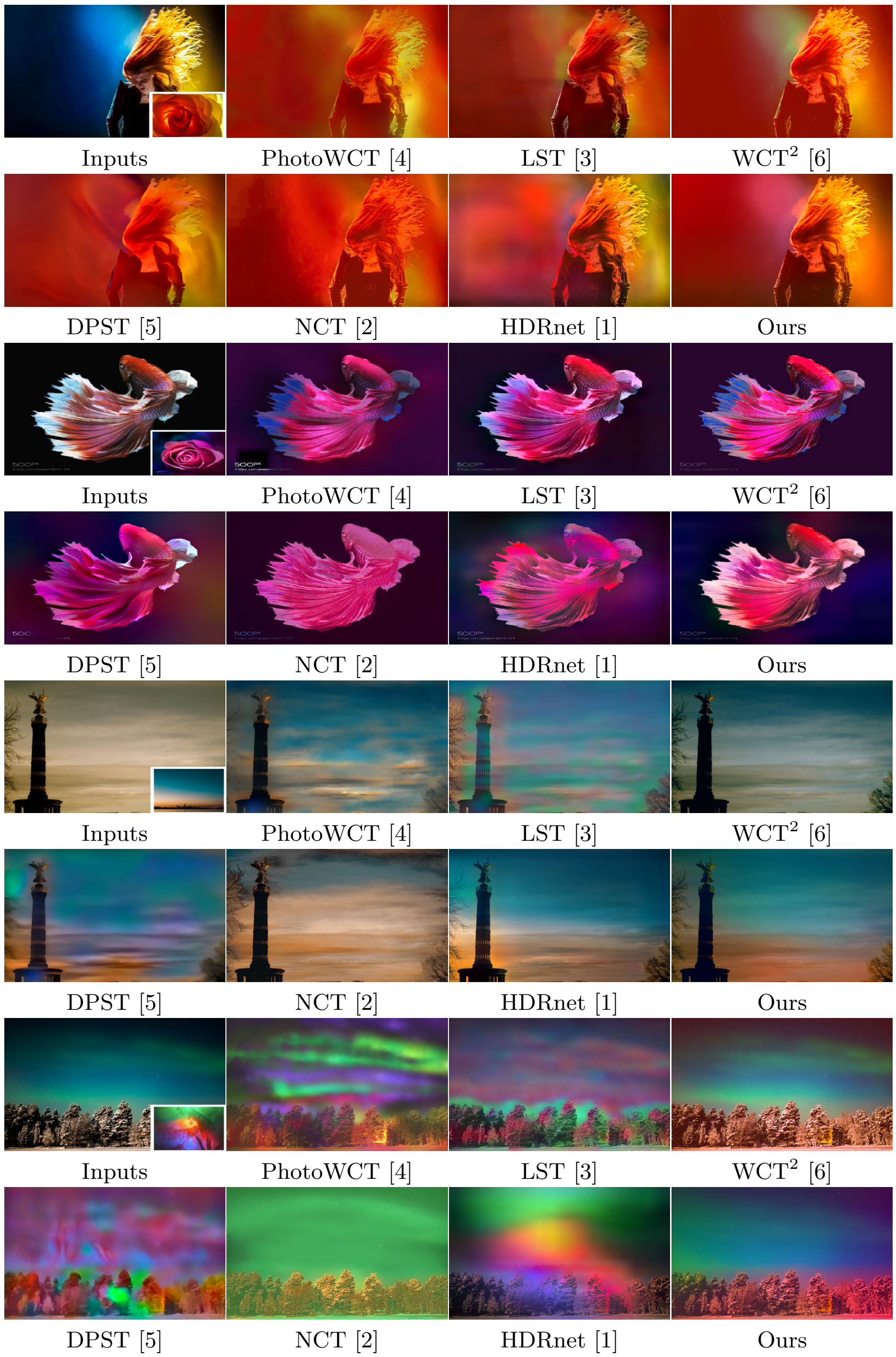


Fig. 1. Qualitative comparison of our method against six baselines on some challenging examples.

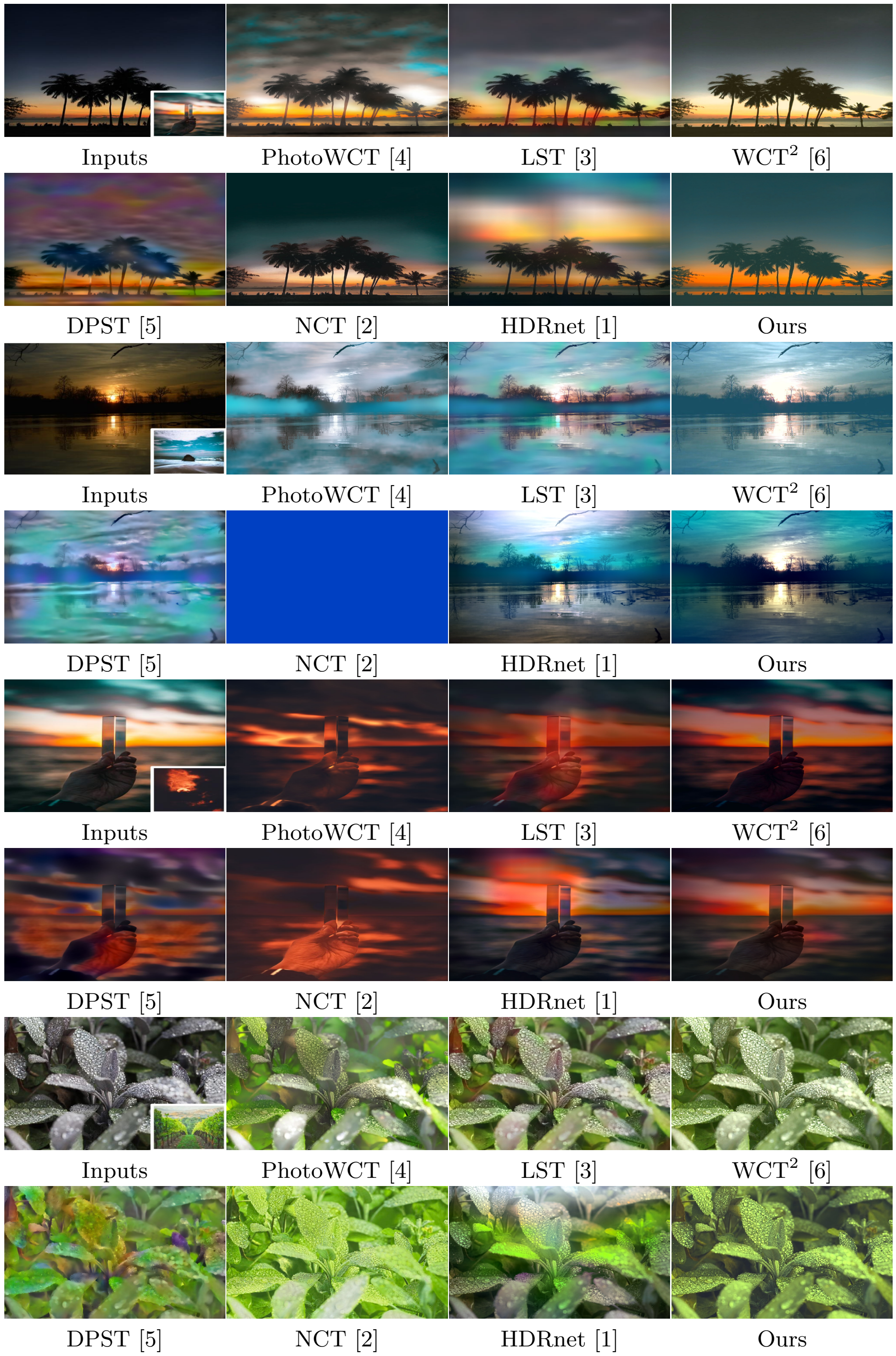


Fig. 2. Qualitative comparison of our method against six baselines on some challenging examples.

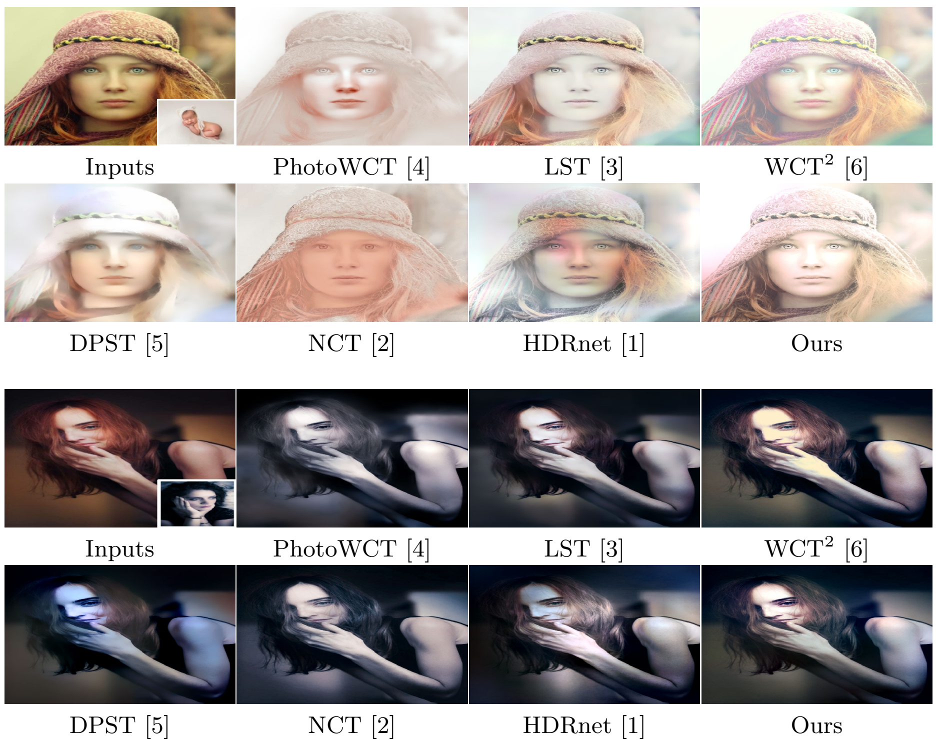


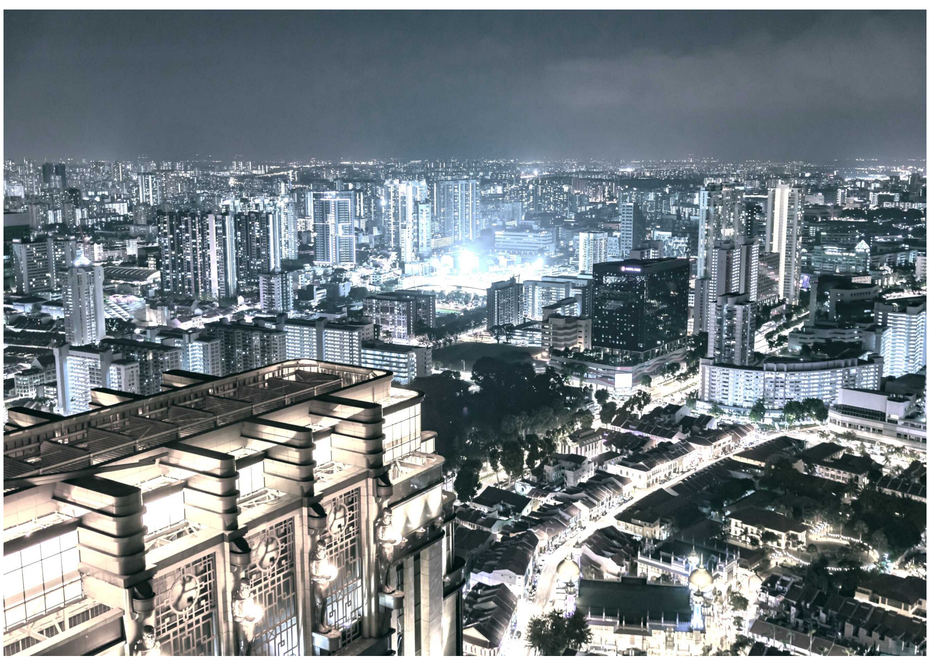
Fig. 3. Qualitative comparison of our method against six baselines on some challenging examples.

Inputs

Output

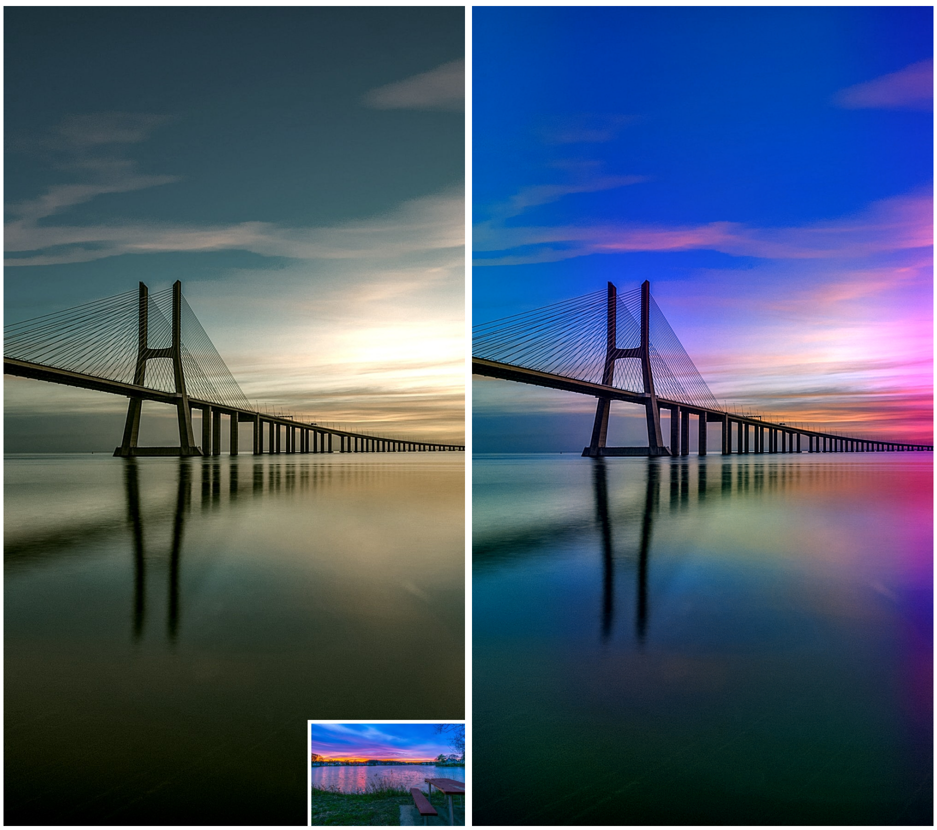
Fig. 4. High-resolution example.

Inputs



Output

Fig. 5. High-resolution example.



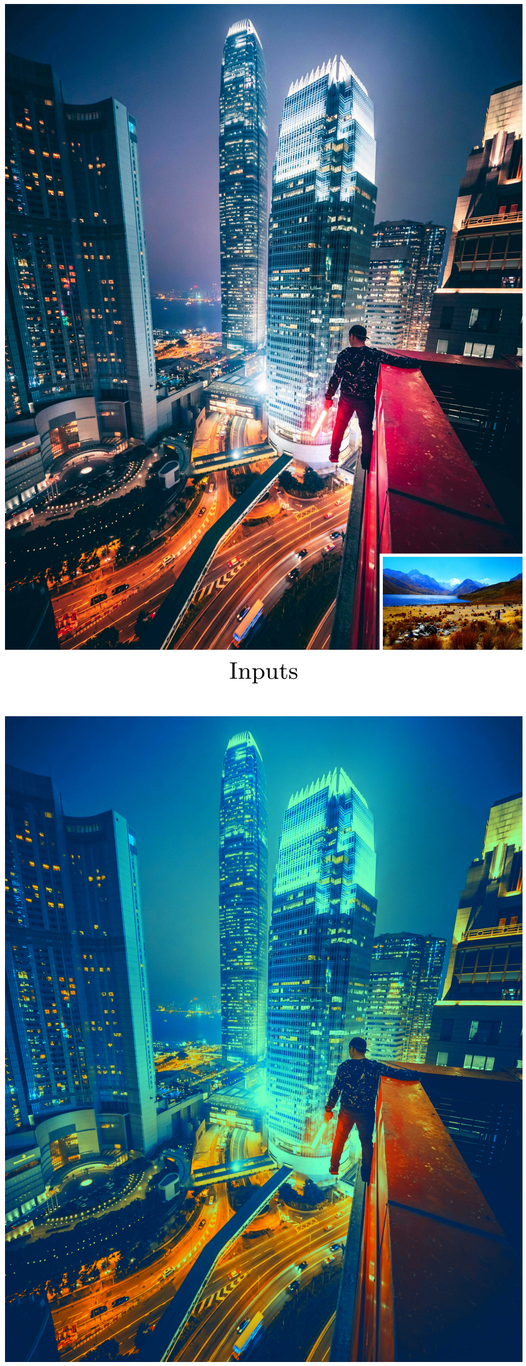
Inputs Output

Fig. 6. High-resolution example.

Inputs

Output

Fig. 7. High-resolution example.



Inputs

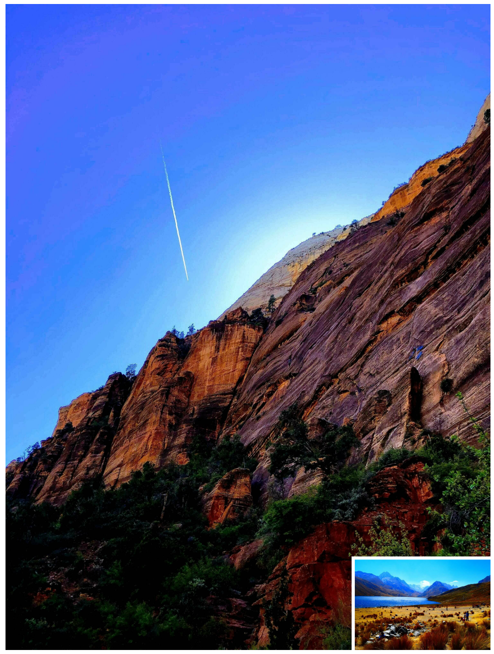
Output

Fig. 8. High-resolution example.

Inputs

Output

Fig. 9. High-resolution example.



Inputs

Output

Fig. 10. High-resolution example.

Inputs Output

Fig. 11. High-resolution example.

Inputs

Output

Fig. 12. High-resolution example.

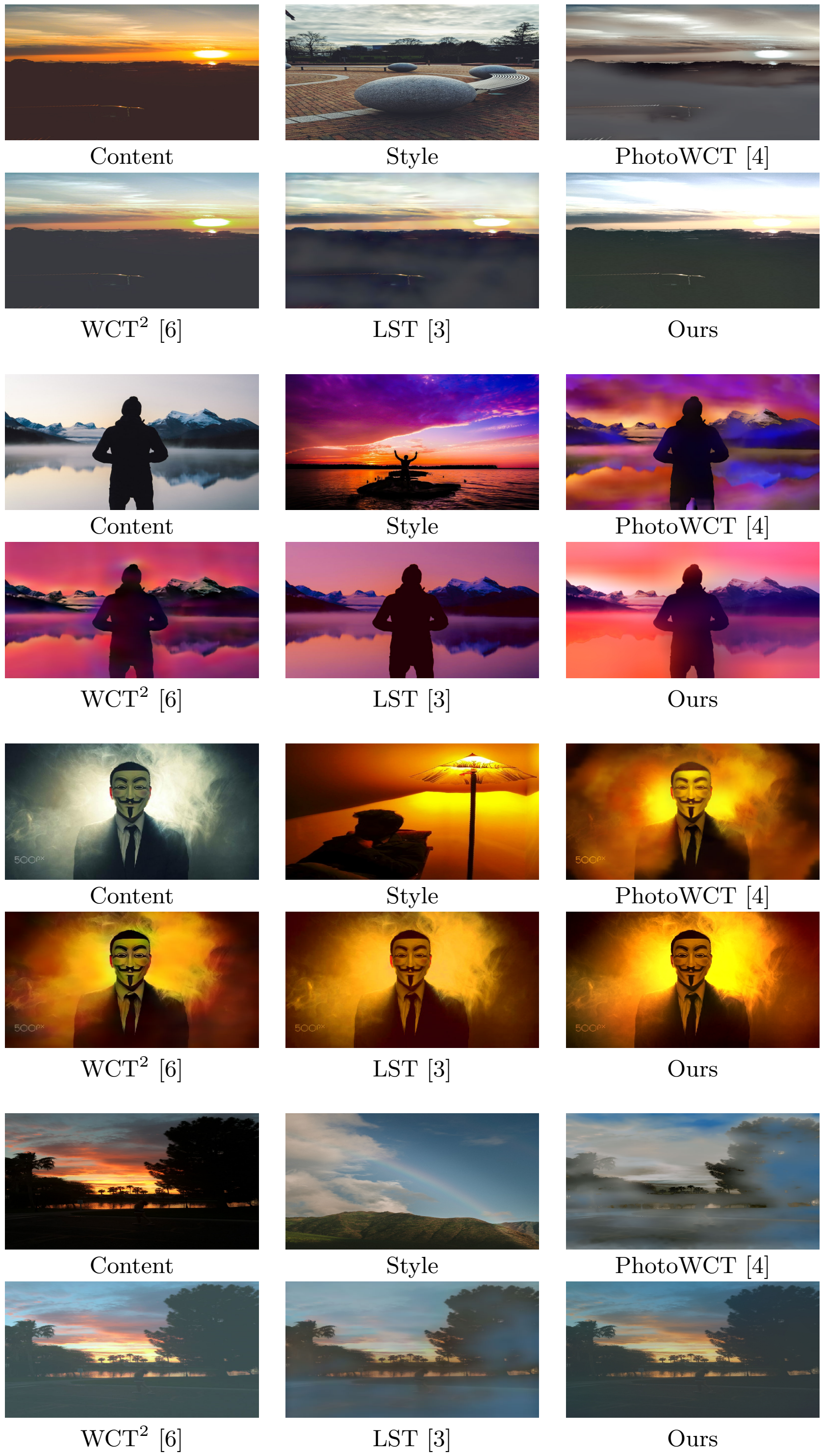


Fig. 13. Additional Qualitative Comparisons on Test Set.

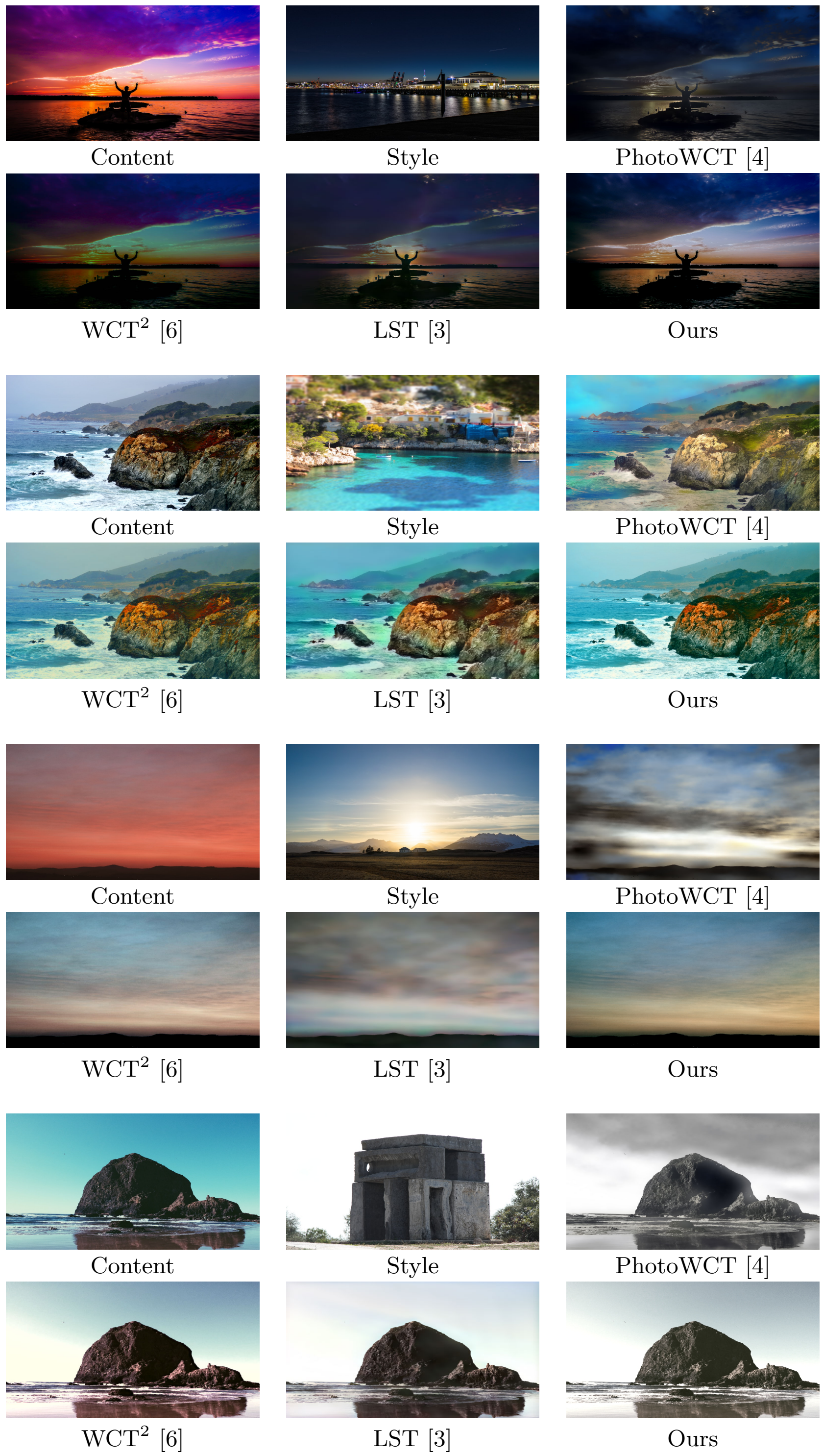


Fig. 14. Additional Qualitative Comparisons on Test Set.

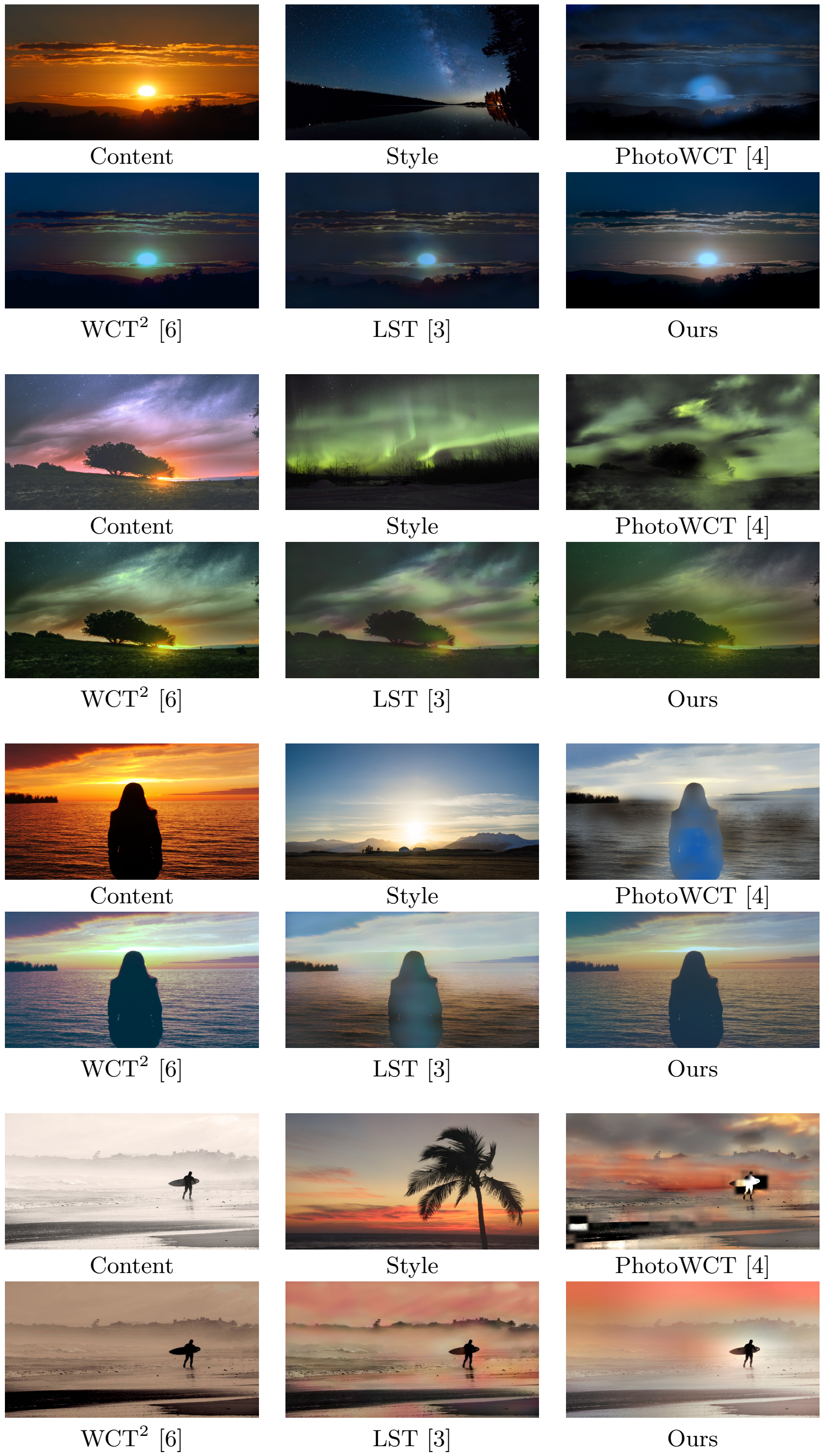


Fig. 15. Additional Qualitative Comparisons on Test Set.

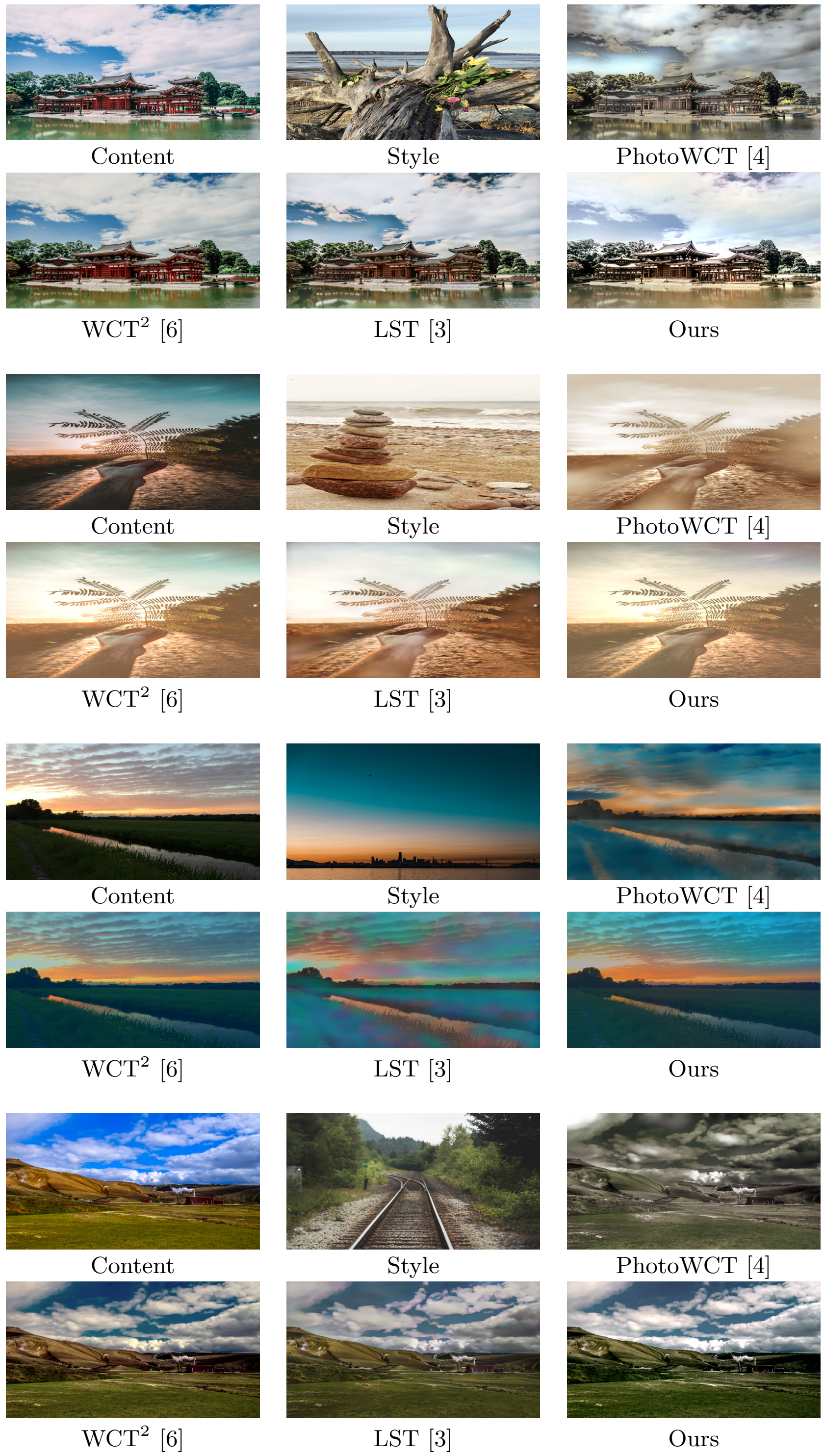


Fig. 16. Additional Qualitative Comparisons on Test Set.

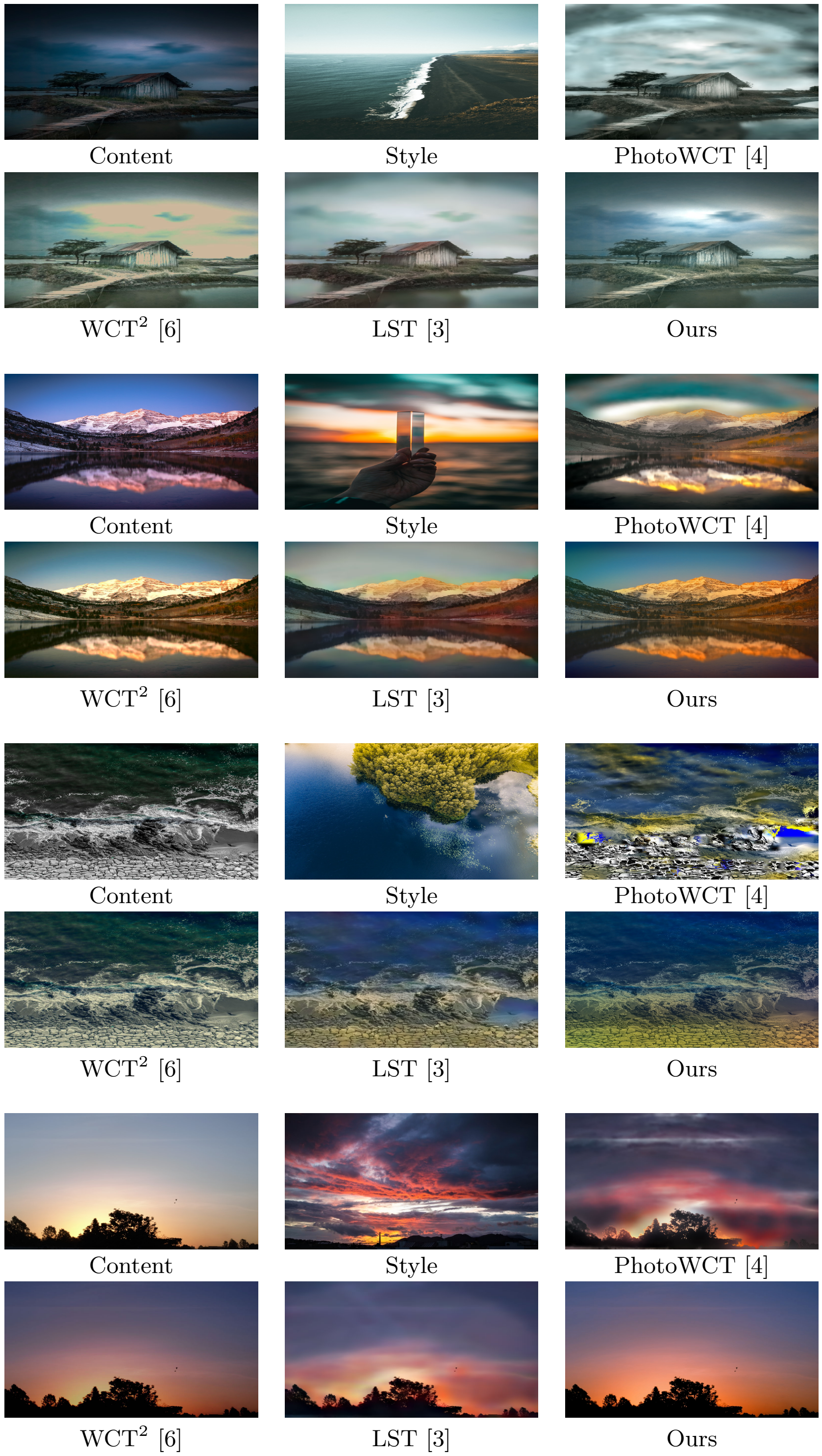


Fig. 17. Additional Qualitative Comparisons on Test Set.

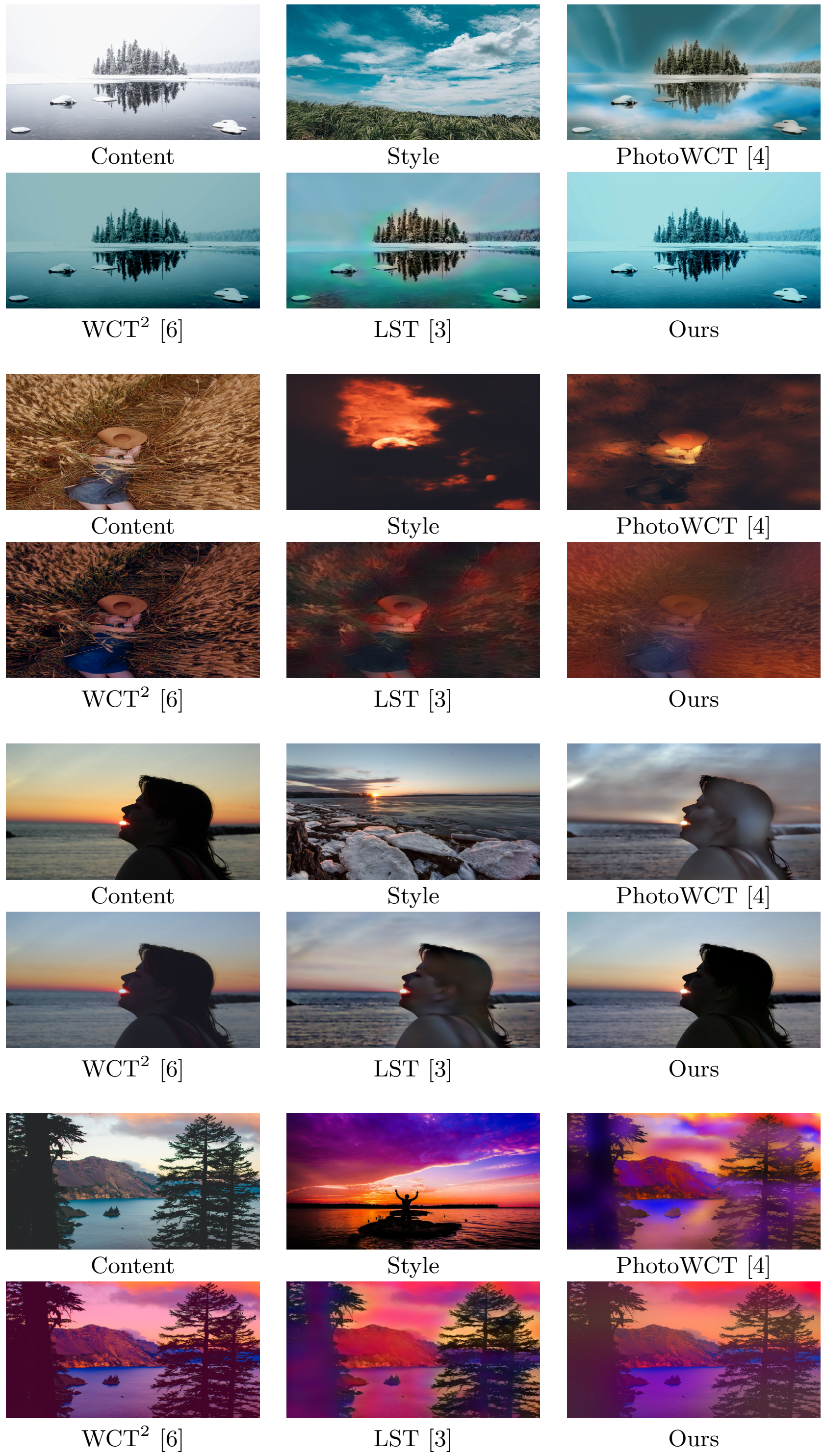


Fig. 18. Additional Qualitative Comparisons on Test Set.

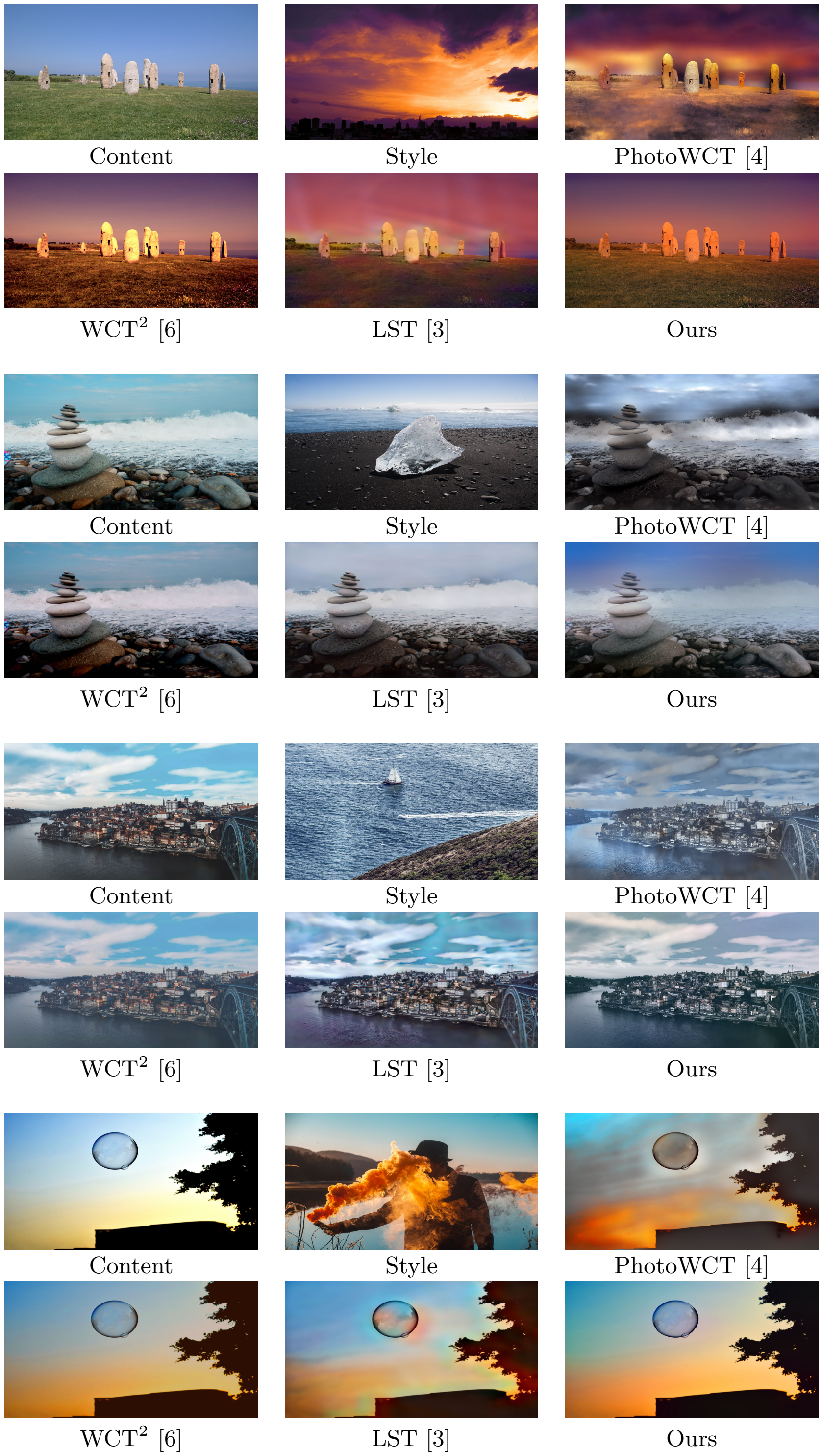


Fig. 19. Additional Qualitative Comparisons on Test Set.

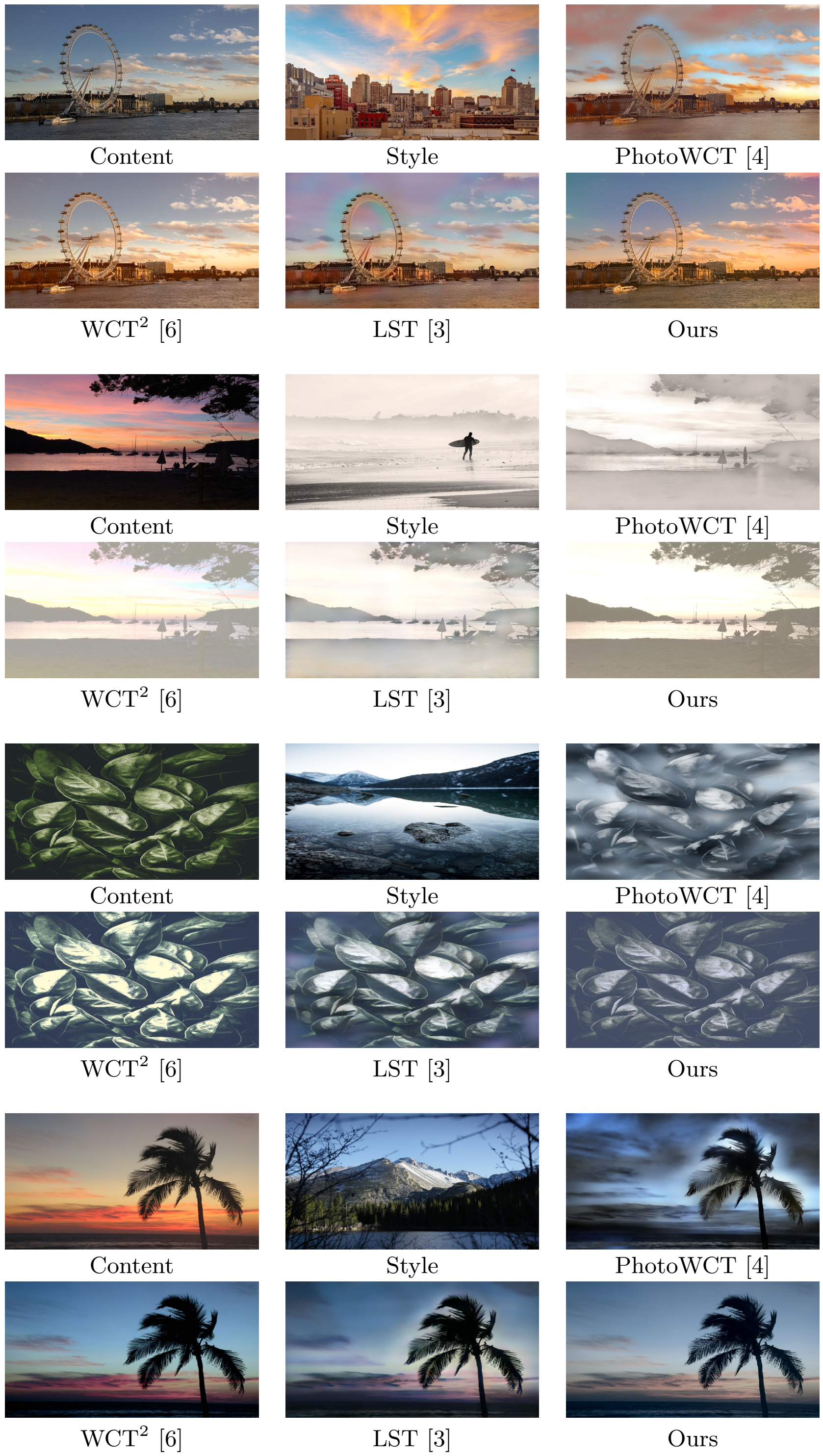


Fig. 20. Additional Qualitative Comparisons on Test Set.

References

1. Gharbi, M., Chen, J., Barron, J.T., Hasinoff, S.W., Durand, F.: Deep bilateral learning for real-time image enhancement. ACM TOG (2017)
2. He, M., Liao, J., Chen, D., Yuan, L., Sander, P.V.: Progressive color transfer with dense semantic correspondences. ACM TOG (2019)
3. Li, X., Liu, S., Kautz, J., Yang, M.H.: Learning linear transformations for fast image and video style transfer. In: CVPR (2019)
4. Li, Y., Liu, M.Y., Li, X., Yang, M.H., Kautz, J.: A closed-form solution to photorealistic image stylization. In: ECCV (2018)
5. Luan, F., Paris, S., Shechtman, E., Bala, K.: Deep photo style transfer. In: CVPR (2017)
6. Yoo, J., Uh, Y., Chun, S., Kang, B., Ha, J.W.: Photorealistic style transfer via wavelet transforms. In: ICCV (2019)