Toward Real-world Panoramic Image Enhancement

Supplementary material

Yupeng Zhang[1] Hengzhi Zhang[1] Daojing Li[1] Liyan Liu[1] Hong Yi[1] Wei Wang[1]
Hiroshi Suitoh[2] Makoto Odamaki[2]
[1] Ricoh Software Research Center (Beijing) Co., Ltd. Haidian District, Beijing, China
[2] Ricoh Company, Ltd. Tokyo, Japan
{Yupeng.Zhang, Hengzhi.Zhang, Daojing.Li, Liyan.Liu, Hong.Yi, wei.wang}@srcb.ricoh.com
{hiroshi.suitoh, makoto.odamaki}@jp.ricoh.com

Abstract

This supplementary material consists of four sections. The first section presents examples about how existing perspective image datasets do not work well for panoramic image enhancement. The second section describes the mis-alignment issue between panoramic and high-quality images caused by different projection methods, and the detailed workflow of our two-step matching algorithm. This is followed by the third section showing stain artifacts caused by color balance difference between panoramic and high-quality images, and our solution. The fourth section compares the enhanced images and ground truths to demonstrate that we achieve high quality results by using Pano-Hi dataset.

1. Panoramic image enhancement using dataset of perspective images

This section visually shows how datasets of perspective images do not work well for panoramic image enhancement due to low data similarity. The reason and analysis can be found in Section 1 page 2 of the main paper.

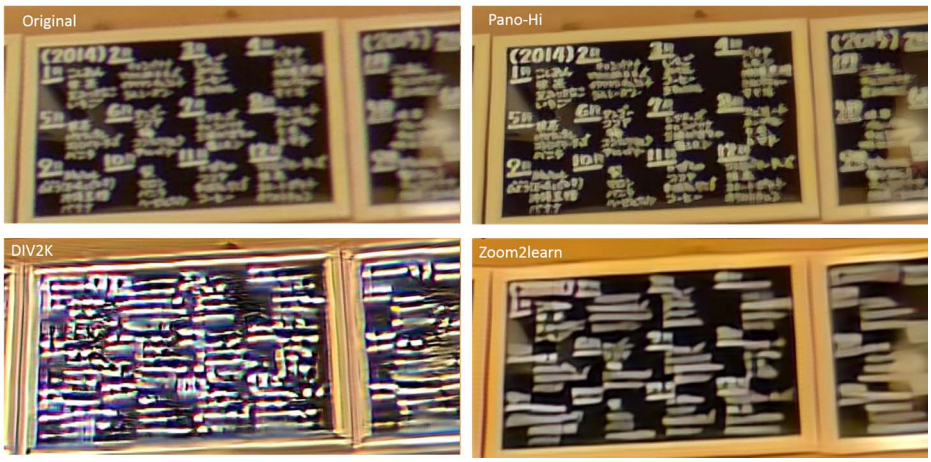
In order to evaluate the enhanced panoramic image quality using state-of-the-art super-resolution (SR) and enhancement models trained on perspective image datasets, we use the models provided by the authors' github sites. These models are trained by using original network architectures proposed by the authors and we use them without modification. For SR methods, since we need to keep the image size unchanged between the input and output, we downsample the original 5K (5376×2688) image and use the models of each state-of-the-art to super-resolve it, resulting in a 5K output image. Results are shown in Figure 1.

We can see that SRGAN [1], RCAN [2] and ESRGAN [3], which are trained using perspective image datasets, do not generate perceptually good results. They are blurrier than the high-end camera image. ESRGAN trained by Pano-Hi dataset (our real world panoramic dataset for enhancement purpose, see Section 1 of the main paper for details), however, has similar texture details and sharpness to the high-end camera image.

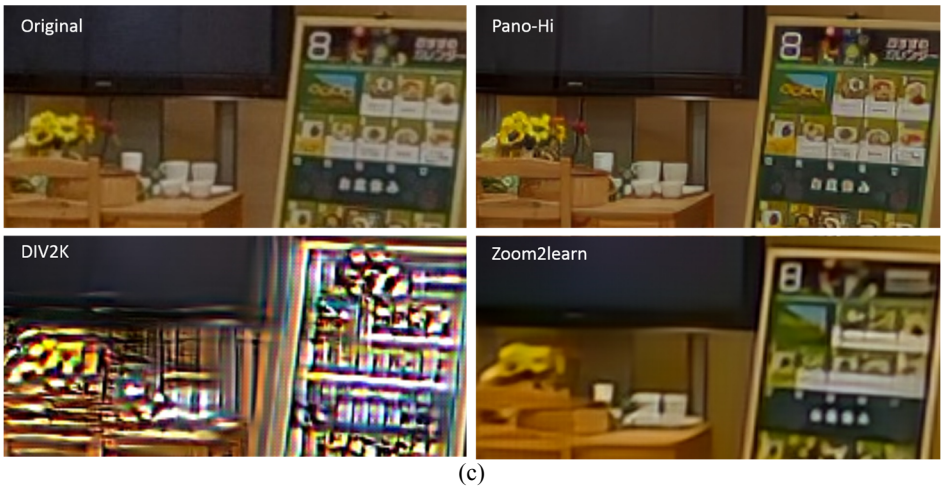
Figure 1: Apply state-of-the-art models trained by datasets of perspective images to panoramic image enhancement. ESRGAN using Pano-Hi dataset is shown for comparison purpose. The images except the second one are cropped from 5K (5376×2688) panoramic images. The first is an original panoramic image. The third to fifth images are enhanced by different models using perspective image datasets. The last image is enhanced using Pano-Hi dataset. The second image is cropped from a ground truth perspective image captured by a high-end APS-C camera. To obtain the same field of view (FoV) for low-high-quality (LQ-HQ in short) pairs for matching, we downsample the high-end camera image.

In addition, we also trained our compact network (see Section 3.3 of the main paper for details) with different datasets of perspective images. We use DIV2K [4] combined with Flickr2K [5] (DIV2K_Flickr2K in short), and Zoom2learn [6] as our training sets, and create low quality (LQ) training images synthetically. Specifically, LQ images are created by applying Gaussian noise and blur to the high quality (HQ) ones. We then extract small LQ-HQ patches from LQ-HQ images without patch matching. Our

(a)



(b)



(c)

Figure 2: Enhancement results by using datasets of perspective images and Pano-Hi dataset.

testing set consists of 39 panoramic images of 5K resolution. Visual results of three sample images are shown in Figure 2. We also list the average perceptual index (PI) [7] scores of the 39 panoramic images for three different datasets in Table 1. Here ours_Pano-Hi, ours_DIV2K and ours_Zoom represent our compact network trained with Pano-Hi dataset, DIV2K_Flickr2K dataset and Zoom2learn dataset, respectively.

The three sample images in Figure 2 shows that visually Pano-Hi dataset outperforms models trained with DIV2K_Flickr2K and Zoom2learn datasets, which consist of perspective images. Results by Pano-Hi dataset obtains naturally enhanced images and richer texture details compared to the original ones. DIV2K_Flickr2K and Zoom2learn datasets generate unpleasant images without clear texture details.

The PI score in Table 1 also indicates that the Pano-Hi dataset obtains the best result for all 39 testing images.

Models	PI
ours_Pano-Hi	**3.74**
ours_DIV2K	3.80
ours_Zoom	6.37

Table 1: The average perceptual index (PI) scores computed from 39 panoramic images of 5K (5376×2688) resolution for Pano-Hi dataset and two perspective image datasets.

2. Patch matching algorithm in details

In this section, we first describe the necessity of converting from equirectangular projection to perspective projection before matching LQ and HQ images. Then, we introduce our two-step matching algorithm in details.

2.1. Equirectangular projection of panoramic image

In this work, we focus on 360 degree panoramic images, also known as omnidirectional images. There are many projection methods to describe omnidirectional image. Equirectangular projection is the well-known one as a representation of omnidirectional view because it is a rectangular shape with orthogonal latitude and longitude of equal spacing. The mathematical definition of the equirectangular projection from the spherical coordinate and its inverse projection can be found in [8].

In our case, we need to align LQ and HQ patches precisely before training. This requires converting equirectangular projection to perspective one first because the matching phase requires two perspective images: LQ converted from equirectangular projection and HQ the ground truth image. However, equirectangular projection cannot be converted to perspective projection using linear transformation such as homography because this conversion involves non-linear equations. Fortunately, the conversion from equirectangular to perspective projections has already been studied and implemented [9]. Figure 3 demonstrates how the matching between equirectangular and perspective projections fails if we apply homography transformation to equirectangular projection without converting to perspective projection first. We can see from Figure 3 that there are curved lines in the equirectangular projection. By converting equirectangular to perspective

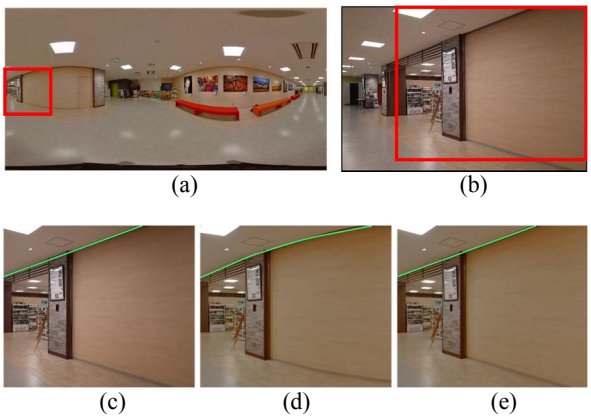


Figure 3: Image matching between perspective and equirectangular projections. (a)Equirectangular projection of the panoramic image (b)Perspective projection of the high-end camera image. The red boxes in (a) and (b) indicate regions to be matched. (c) Red box cropped from high-end image. A green line shows that it is a straight line. (d)Matched region cropped from equirectangular projection. The green line is curved due to mis-matching between equirectangular and perspective projections. (e) Matched region cropped from perspective converted from equirectangular projection. The green line becomes straight because matching is done between two perspective projections (Zoom-in for better visual comparison).

projection first, we can use traditional homography transformation to obtain well-aligned LQ-HQ image pairs.

If we select the areas near the equator of the equirectangular projection, we can roughly align equirectangular with perspective projection without converting to perspective projection first. However, for pole areas, we should convert before matching.

2.2. Detailed patch matching algorithm

In order to align LQ and HQ patches efficiently, we adopt a two-step matching algorithm as shown in Figure 4. In the first step, we obtain matched LQ-HQ images of a large size. In this step, LQ is an original low resolution panoramic image in the equirectangular projection. HQ is an original high resolution perspective APS-C camera image whose content is partially overlapped with the panoramic image. Firstly, a feature matching method is used to find the view point and get the matched positions between LQ and HQ. We use Oriented FAST and Rotated BRIEF (ORB) [10] as the feature descriptor to detect key points. Secondly, divide the matched LQ-HQ images into n pieces (Here n is determined empirically). HQ pieces have no overlap while LQ pieces have large overlapped areas. Then, we conduct matching for each small pieces of LQ and HQ. It involves calculating several matrices for n different pieces, and use them to convert the equirectangular LQ to a perspective LQ image as discussed in section 2.1. Then we align each piece

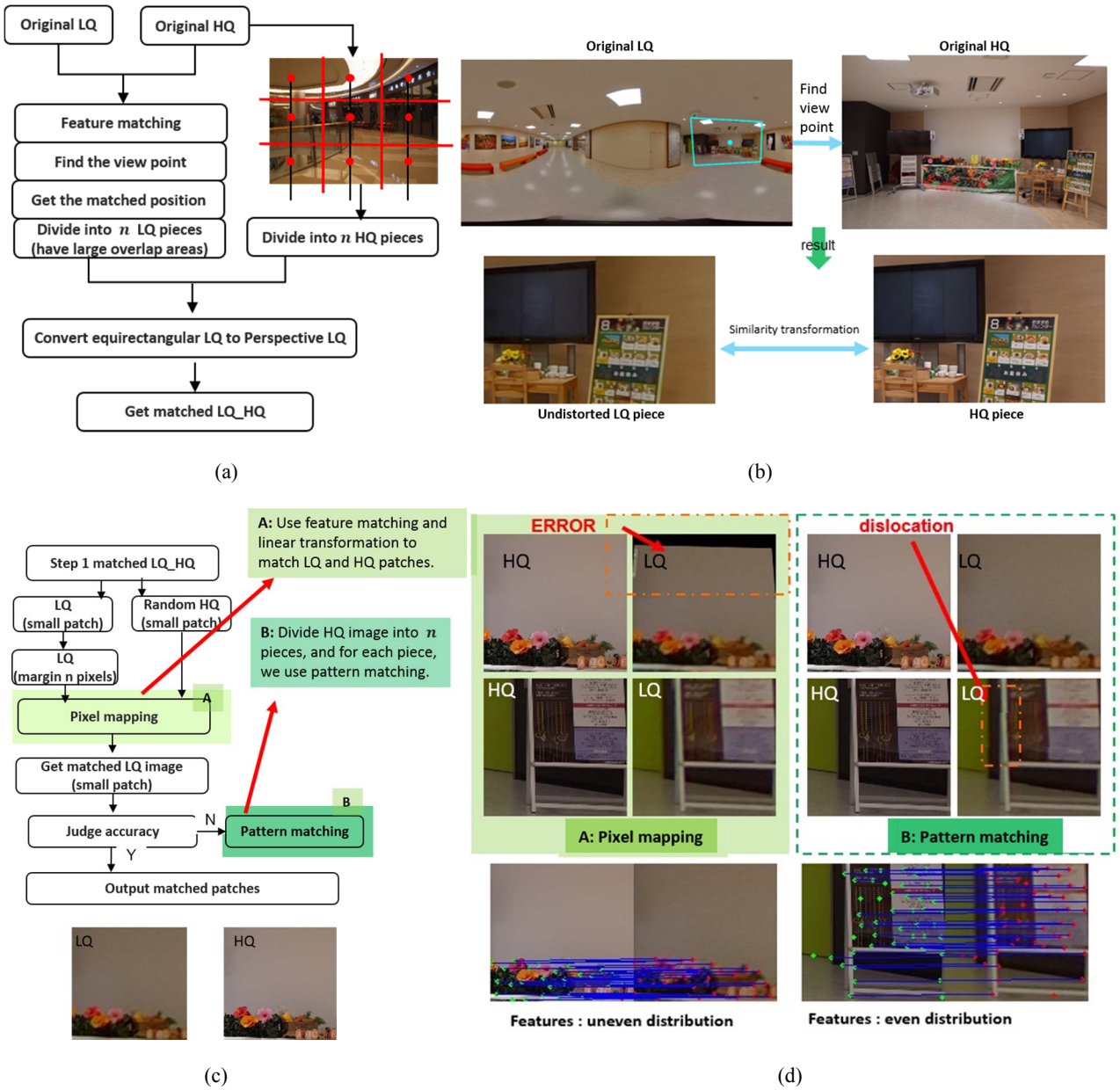


Figure 4: Two-step matching algorithm for LQ-HQ patch generation. The matched LQ and HQ patches serve as the network input and ground truth, respectively. (a) illustrates workflow of the first step. (b) shows the matching results of the first step. (c) illustrates workflow of the second step and the matched LQ and HQ patches. (d) shows pixel mapping and pattern matching results as well as their respective problems. Pixel mapping produces errors when there are uneven feature distributions on the patch. Pattern matching has dislocation issue as shown in the right column of (d).

of the perspective LQ with HQ using similarity transformation by ORB descriptor again. Figure 4 (a) describes the workflow of the first step of our matching algorithm, and Figure 4(b) shows matching results of the first step.

In the second step, the matched LQ-HQ images in the first step serve as the input of this step. Firstly, a random small image patch is obtained from the HQ image, another patch with several pixels expanded (denoted by margin n pixels in Figure 4(c)) is extracted at the same location in the LQ image. The LQ patch is matched with the HQ by using pixel mapping. Secondly, we judge the accuracy of the matched patches. If it meets the requirements of the accuracy, which can be defined, for example, by computing

PSNR [11] before and after the matching, we output it as the final matched patches. Otherwise, we use pattern matching to re-match. The reason for using pixel mapping first is that it is more effective than pattern matching. The latter has the dislocation problem in some cases as shown in Figure 4(d) right column. However, if the feature distribution of the patch is uneven, error will appear on the matched LQ patch by pixel mapping (see Figure 4 (d) left column). Pattern matching, however, does not have such problem. Therefore, if the pixel mapping is not appropriate (as judged by the accuracy mentioned above), the pattern matching is selected.

3. Artifacts caused by color balance difference between low-high quality patches

This section shows how Pano-Hi dataset alone leads to unpleasant artifacts such as unnatural stain on no-feature regions in an image, and increasing data diversity by adding synthetic data from perspective image datasets solves this problem.

The amount of patch pairs for Pano-Hi and perspective image dataset is 1134k and 919k with patch size 32×32, respectively. We can see from Figure 5(a) that the color balance between LQ and HQ in the Pano-Hi dataset is different, which leads to the stain artifacts shown in the left column of (b) (c) and (d). In the perspective image dataset shown in Figure 5(a), there is no color balance difference between LQ and HQ patches. The right columns of Figure 5 (b) (c) and (d) show results using the combination of Pano-Hi and perspective image datasets.

Besides data augmentation, we can impose other constraints to solve the color balance issue. We used SFTGAN [12] by conditioning our network. We found that the artifacts mostly appear at the non-texture regions (e.g. car ceiling and sky in Fig.5 (c) and (d), respectively) and textures with regular patterns (e.g.Fig.5(b)). We then made a categorical segmentation to the input images and used SFTGAN to train a conditioned network to guide the network to generate correct textures (sky, ceiling without artifact).

4. Comparison between enhanced results and the high-end camera quality (downsampled)

This section presents more results to compare the enhanced image to the high-end (downsampled ground truth). As seen from Figure 6, the enhanced patches improve the original image quality effectively in the aspects of texture details, clarity, noise and chromatic aberration reductions. Note that there is obvious color balance problem between the panoramic and high-end patches (see Figure 6 (b) bottom row and (c) middle row). This is why

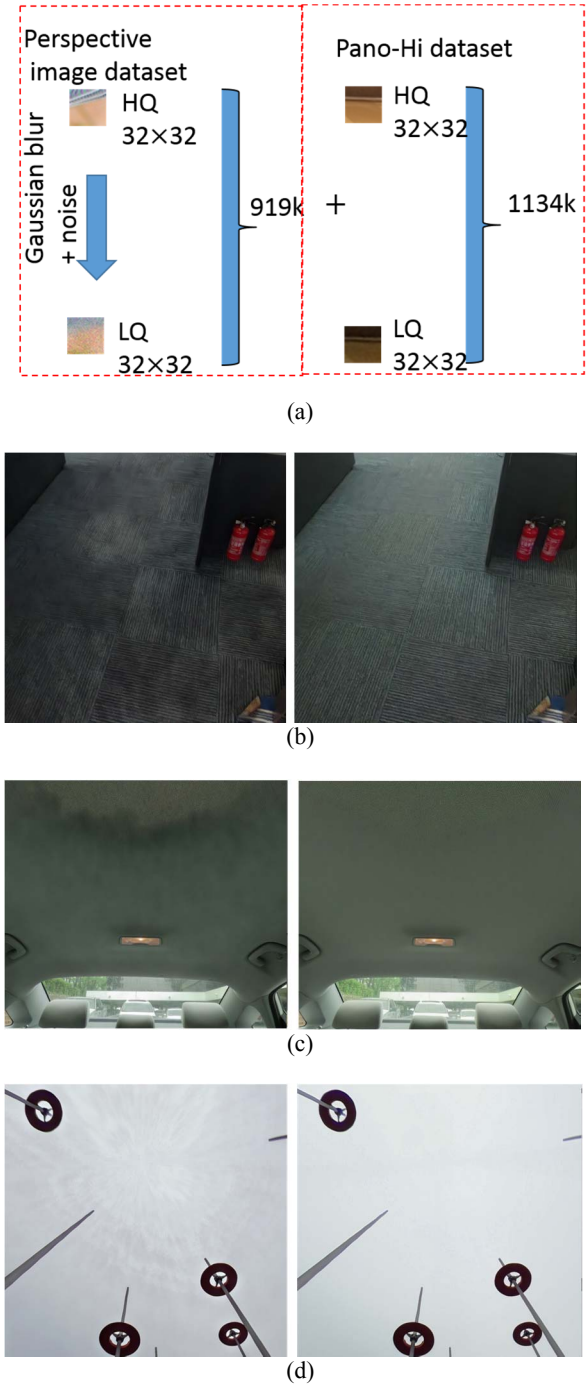


Figure 5: Stain artifacts caused by color balance difference between low-high-quality pairs of patches in the Pano-Hi dataset. (a) Increase data diversity by synthetically adding perspective image datasets to the Pano-Hi dataset. The patches show different color balance between HQ and LQ in Pano-Hi dataset but same in perspective image dataset. (b)-(d) compare results between Pano-Hi dataset alone (left) and the combination of Pano-Hi and perspective image datasets(right).

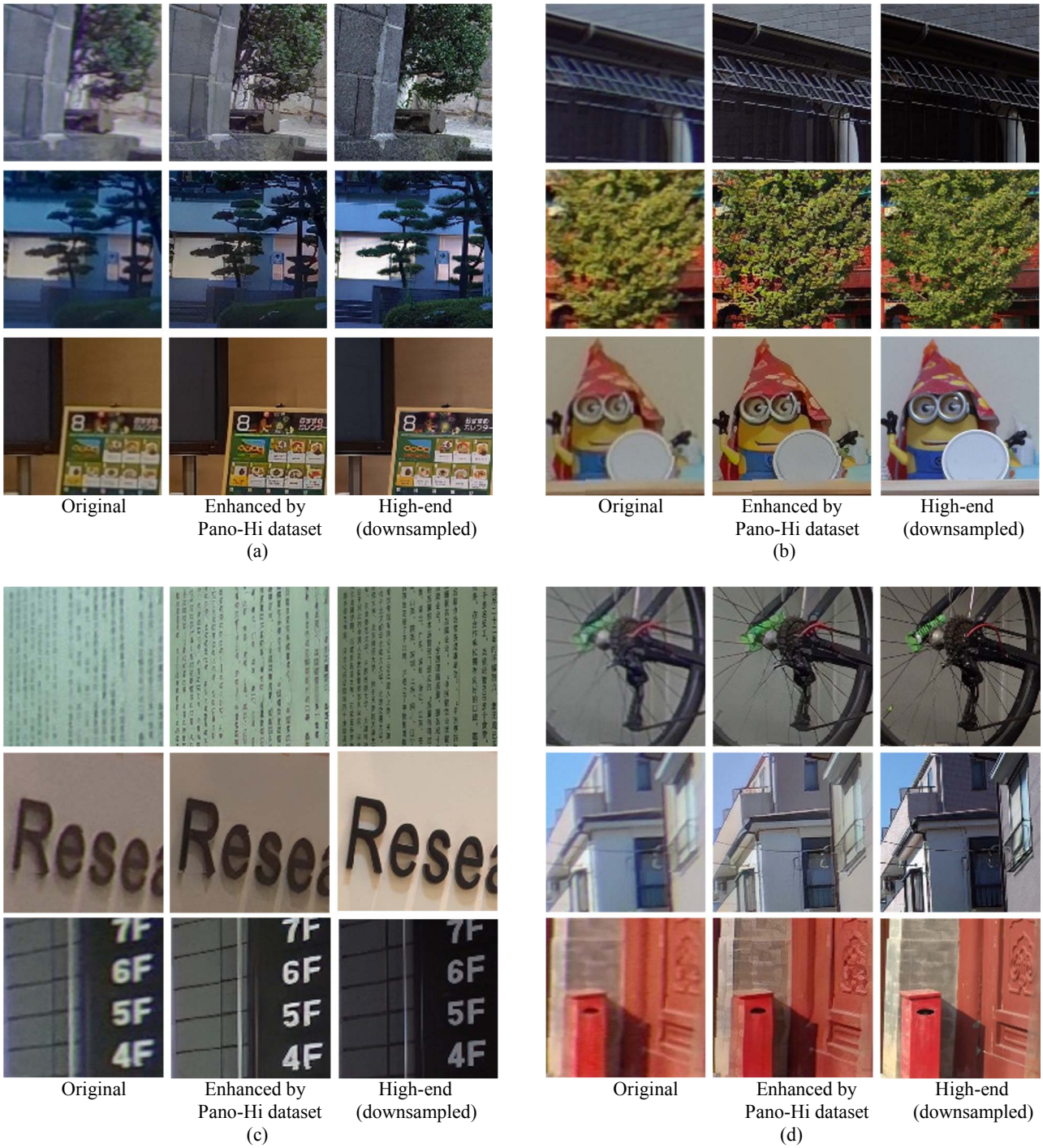


Figure 6: Comparison of original, enhanced and high-end images. To obtain the same field of view (FoV) for LQ-HQ pairs for matching, we downsampled the high-end (HQ).

stain artifacts appear in the resulting image as introduced in Section 3. By using data augmentation, the color balance of the enhanced patches becomes very close to that of the original ones.

References

[1] C.Ledig, L.Theis, F.Huszár, J.Caballero, A.Cunningham, A.Acosta, A.Aitken, A.Tejani, J.Totz, Z.Wang, and W.Shi. Photo-realistic single image super-resolution using a generative adversarial network. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4681-4690, 2017.

[2] Y.Zhang, K.Li, K.Li, L.Wang, B.Zhong, and Y.Fu. Image super-resolution using very deep residual channel attention networks. In Proceedings of the European Conference on Computer Vision (ECCV), pages 286-301, 2018.

[3] X.Wang , K.Yu , S.Wu, J.Gu, Y.Liu , C.Dong , C.C.Loy, Y.Qiao and X. Tang. Esrgan: Enhanced super-resolution generative adversarial networks. In Proceedings of the European Conference on Computer Vision (ECCV), 2018.

[4] E.Agustsson and R.Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pages 126-135, 2017.

[5] R.Timofte et al. Ntire 2017 challenge on single image super-resolution: Methods and results. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pages 114-125, 2017.

[6] X.Zhang, Q.Chen, R.Ng, and V.Koltun. Zoom to Learn, Learn to Zoom. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3762-3770, 2019.

[7] Y.Blau, R.Mechrez , R.Timofte, T.Michaeli, and L.Zelnik-Manor, The 2018 PIRM Challenge on Perceptual Image Super-Resolution. In European Conference on Computer Vision, pages 334-355, 2018

[8] https://en.wikipedia.org/wiki/Equirectangular_projection, 2019.

[9] P.Bourke. Converting an equirectangular image to a perspective projection. http://paulbourke.net/miscellaneous/sphere2persp/, 2016.

[10] E.Rublee, V.Rabaud, K.Konolige, and G.Bradski. ORB: An efficient alternative to SIFT or SURF. In IEEE International conference on computer vision, pages 2564-2571, 2011.

[11] Z.,Wang, A.C.Bovik, H.R. Sheikh, and E.P.Simoncelli. Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing, 13(4): 600-612, 2004.

[12] X.Wang, K.Yu, C.Dong, and C.C. Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 606-615, 2018.