BluRef: Unsupervised Image Deblurring with Dense-Matching References – Supplementary Material –

Abstract

In this supplementary material, we provide extended technical details and additional results supporting the BluRef framework. We first present implementation information and training-time analysis, clarifying how the full pipeline is executed in practice. We then describe the training procedure of the Dense Matching ($\mathcal{DM}$) module in greater depth, including the construction of synthetic training pairs, the degradation strategies used, and the role of augmentation in enabling robust correspondence learning. Additional ablation studies are included to further validate the design choices of BluRef, particularly regarding $\mathcal{DM}$ training configurations, reference aggregation strategies, and alternative deblurring backbones. We also report quantitative results for the Restormer backbone and provide extensive qualitative visualizations on GoPro, RB2V, and PhoneCraft, illustrating the improvements achieved by BluRef under diverse real-world blur conditions. Finally, we visualize the evolution of pseudo-sharp images and confidence masks across training iterations to highlight how the model progressively refines its supervision signals. All code and resources used in our experiments are included for reference.

1. Extra System Details

1.1. Extra Implementation Detail

During BluRef's pipeline training, each iteration processes an input image alongside a list of reference images, embedded into the deblurring model's backbone as data preprocessing. We utilize a batch size of 8, with code details provided in the supplementary materials.

1.2. Training time

Embedding an iterative refinement technique by Truong *et al.* [5], our training period lasts 4 days on 1 NVIDIA A100 GPU. This duration is approximately 1.2 times longer than that of supervised models, a difference we consider negligible given the ability to train on an unpaired dataset and subsequently achieve pseudo-sharp images.

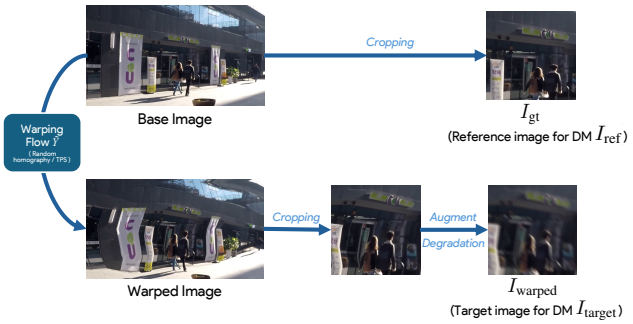

Figure 1. **Training pipeline of the dense matching module $\mathcal{DM}$.** A sharp image is resized and randomly warped to create a deformed view. Center crops from the original and deformed images form a synthetic pair $(I_{\text{warped}}, I_{\text{gt}})$, where I_{warped} is further degraded using BSRGAN Augmentation which is fixed to add blur augmentation. The $\mathcal{DM}$ learns to map $(I_{\text{warped}}, I_{\text{gt}})$ to a transformed image I_{trans} and a confidence mask M_{conf}.

1.3. Augmentation in DenseMatching Model

As described in Sec. 3.2 of the main paper and illustrated in our synthetic-pair generation pipeline (Fig. 1), we adopt the BSRGAN degradation pipeline [6] to augment the synthetic training pairs $(I_{\text{warped}}, I_{\text{gt}})$ for the Dense Matching ($\mathcal{DM}$) model. This augmentation is essential for enabling the DM module to learn reliable correspondences across blurred and sharp domains.

While BSRGAN was originally developed for blind image super-resolution, its diverse degradation space makes it well suited for our setting. We adapt the pipeline for blind image deblurring by expanding it with additional blur-oriented degradations. Concretely, our augmentation includes:

- **Gaussian blur:** kernel sizes sampled from $\{7, 9, 11\}$ with random variance
- **Anisotropic and isotropic blur** following the standard BSRGAN blocks
- **Synthetic motion blur** implemented using directional/diagonal averaging kernels and a simple frame-averaging strategy to mimic real motion smear
- **Gaussian noise** with random variance
- **JPEG compression noise** with quality factor $q \sim U(30, 60)$

Models	PSNR↑	SSIM↑
BluRef + PDC-Net+	27.72	0.820
BluRef + GLU-Net	26.63	0.798
Upperbound (NAFNet)	28.54	0.824

Table 1. Comparison of BluRef training between with PDC-Net+ and GLU-Net.

- **Random downsampling/upsampling** (bicubic or bilinear)
- **Degradation shuffle**, where all degradation operations are applied in a randomized order.

These augmentations generate a wide spectrum of blur and noise patterns, including realistic motion smear produced by averaging, enabling the $\mathcal{DM}$ model to learn dense correspondences under heterogeneous and challenging blur conditions. Importantly, keeping the degradation shuffle ensures the model is trained on a richly varied distribution of warped images, improving its robustness when matching real-world blurry inputs.

2. Restormer's Quantitative Result

As mentioned in the main paper, we provide the quantitative results of Restormer using the Weighted Average (Avg.) and Sequential Accumulation (Seq.) strategies. Additional details are available in Table 5.

3. Additional Ablation Studies

In the main paper, we conducted extensive ablation experiments to assess the effects of the number of reference frames, the pseudo-ground truth generation module, varying Δ values, different deblurring backbones, and diverse refinement strategies. This supplementary material presents additional ablation studies detailed below.

3.1. BluRef with Different DenseMatching Models

In the main paper, we employ PDC-Net+ as the DM to train BluRef, leveraging its state-of-the-art performance over other baselines. To assess the impact of different DM models, we substitute PDC-Net+ with a pre-trained GLU-Net [4], trained on the same paired $(I_{\text{warped}}, I_{\text{gt}})$ set as PDC-Net+, and follow the identical BluRef training pipeline. Using NAFNet on RB2V ($\Delta = 10$) with the Progressive (Prog.) strategy. Tab. 1 shows that PSNR↑/SSIM↑ reaches 26.43/0.792, which is comparable to PDC-Net+ and approaches the upperbound model. This result highlights the robustness of our BluRef pipeline.

3.2. BluRef with GAN-Based Deblurring Backbones

To validate BluRef's effectiveness with other GAN-based deblurring backbones, we integrate UID-GAN—an unsupervised deblurring backbone benchmarked in the main paper—into our BluRef framework, employing the Progressive (Prog.) strategy with the same DM model (PDC-Net+) and settings on RB2V ($\Delta = 10$). To adapt its unsupervised training to BluRef, we apply the same reconstruction loss as used with NAFNet. As can be seen in Tab. 2, this boosted PSNR↑/SSIM↑ to 24.56/0.698, compared to 22.29/0.581 for the original UID-GAN, confirming BluRef's robust performance across diverse deblurring backbones.

Models	PSNR↑	SSIM↑
UID-GAN	22.01	0.551
UID-GAN + BluRef	24.56	0.698

Table 2. Ablation studies with UID-GAN deblurring backbone.

$\mathcal{DM}$ Training Setting	PSNR↑	SSIM↑
No augmentation	27.52	0.802
Gaussian blur only	29.05	0.861
Gaussian + Noise	29.21	0.868
Motion blur only	29.44	0.874
Down/Up sampling only	28.93	0.855
BSRGAN (original)	30.58	0.892
BSRGAN + Motion Blur (Ours)	**31.87**	**0.955**

Table 3. **Ablation study of the Dense Matching (DM) training pipeline.** We compare different degradation configurations used to synthesize $(I_{\text{warped}}, I_{\text{gt}})$ training pairs. Results are reported on the GoPro validation set.

3.3. Effect of DenseMatching training setting on BluRef Performance

This ablation examines how different degradation settings used to train the Dense Matching (DM) model influence the final BluRef performance. For each degradation configuration, we train a separate DM model on synthetic $(I_{\text{warped}}, I_{\text{gt}})$ pairs generated by geometric warping , followed by one of the following degradation pipelines: (i) No blur or noise (identity), (ii) Gaussian blur only, (iii) Gaussian blur + noise, (iv) Motion blur only, (v) Downsampling/upsampling, (vi) BSRGAN degradations, and (vii) BSRGAN degradations with additional motion blur (Ours). We then keep the entire BluRef framework fixed (NAFNet backbone, Δ=10, N=6 , Progressive Reference Strategy) and plug in each DM model to generate pseudo-sharp targets and train the deblurring network. Thus, the PSNR/SSIM values in Tab. 3 directly reflect how each DM training variant affects the final BluRef performance on the GoPro dataset.

The results indicate that training DM without blur augmentation leads to a significant drop in BluRef performance, showing that the matching module fails to generalize to real blurry inputs. Simple degradations (Gaussian, noise, motion blur, or downsampling) offer only moderate improvements

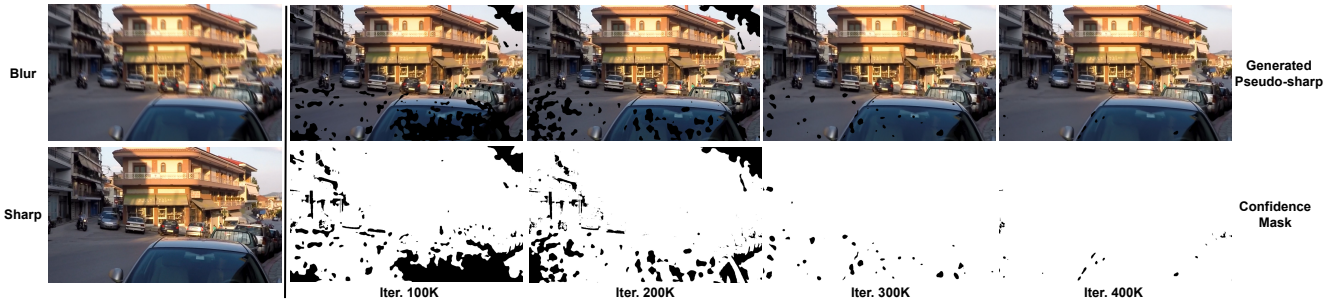

Figure 2. Our generated pseudo-sharp images and their confidence masks follow by iterations.

Figure 3. **Visualization of the Dense Matching output I_{trans} at the first iteration.** The top row shows the blurry input and three sharp reference frames. The middle row displays the ground-truth sharp image and the DM output $I_{\text{trans}} * \bar{M}_{\text{conf}}$, where $\bar{M}_{\text{conf}}$ is the binary confidence mask. The bottom row shows the pixel-wise error map between I_{trans} and the sharp image. Despite being generated at the first iteration and without any refinement, I_{trans} already captures high-frequency scene structures from the references, demonstrating that the Dense Matching module successfully transfers sharp details before the pseudo-sharp aggregation stage (Zoom in for best view).

and remain insufficient to bridge the synthetic–real gap. The original BSRGAN pipeline already yields a strong gain, thanks to its randomized degradation shuffle that exposes the DM model to diverse distortions. Finally, BSRGAN + Motion Blur (Ours) achieves the best PSNR/SSIM, as the additional motion-averaging blur improves robustness to camera shake and produces more stable correspondences. This confirms that a rich combination of geometric warps and different degradations is essential for training a DM model that can generalize to the diverse, unconstrained distortions present in real-world blurry images.

3.4. Impact of Pseduo ground truth generation

To study the necessity of pseudo-ground truth aggregation, we compare the performance of BluRef with and without aggregation strategies for generating pseudo-ground truth I_{pseudo}. Specifically, we train BluRef on RB2V and GoPro ($\Delta = 10$) using NAFNet as the backbone, but without aggregation strategies, forcing the model to learn from individual supervision signals I^n_{pseudo} instead of the aggregated supervision signal I_{pseudo} as proposed. As shown in Tab. 4, this leads to a significant performance drop due to inconsistent supervision signal during deblurring model training, underscoring the crucial role of our iterative refinement with reference images.

Dataset	GoPro	RB2V
BluRef (Prog.)	**31.87 / 0.955**	**27.72 / 0.820**
BluRef w/o aggregation strategy	18.89 / 0.323	18.73 / 0.325

Table 4. Comparison of BluRef training with and without the aggregation strategy. For each test, we report PSNR↑/SSIM↑ scores as evaluation metrics.

4. Additional Visualization

4.1. Qualitative Results in GoPro and RB2V dataset

In this section, we provide additional qualitative figures comparing the image deblurring results of our BluRef and other baselines. Figures 5 and 6 provide samples on GoPro[1] and RB2V[2], respectively.

4.2. Qualitative Results in PhoneCraft dataset

As discussed in Sec. 4.4 - Evaluating on 'BluRef dataset' of the main manuscript, we validate our generated pseudo-sharp images by pairing them with corresponding blurry images to form the paired version of PhoneCraft[3] dataset. We train a compact NAFNet on this dataset for performance benchmarking against generalized deblurring models BSRGAN and RSBlur, both utilizing NAFNet64. We report the qualitative results of these comparisons in Fig. 4. As shown in Fig. 4, training on our pseudo-sharp images facilitates the lightweight version of NAFNet to achieve sufficient and superior performance compared to BSRGAN and RSBlur, which use the default version of NAFNet. These examples demonstrate that our 'BluRef dataset' effectively enables the supervised network capture the blur kernel of real-world blur, reconstruct sharp details accurately, and avoid artifacts or excessive smoothing.

4.3. Visualization of Pseudo-sharp images

In this section, we present the evolution of pseudo-sharp images generated by our model through training iterations with a given blurry input. This experiment was conducted using the GoPro dataset, with $\Delta = 10$ and 6 reference frames utilized for training. As illustrated in Fig. 2, the quality of the pseudo-sharp images is enhanced progressively with increased iterations. Notably, at the final iteration of 400K, the pseudo-sharp image exhibits a high degree of similarity to the actual ground truth image, confirming the effectiveness of our model in synthesizing sharp images from blurred inputs.

Despite the high fidelity of the generated images, there is a residual impact from the confidence masks applied during the image restoration process. As can be seen in the Fig. 2, the black regions in confidence mask identify regions with lower reliability, have a limited influence on the learning of the blur kernel space. However, this influence is marginal and does not substantially degrade the deblurring performance, as evidenced by the minimal deviation from the ground truth in the final iteration. This underscores the robustness of our approach in handling artifacts by efficient iterative refinement process.

Visualization of $\mathcal{DM}$ results before aggregation. To analyze the behavior of the $\mathcal{DM}$ module, we visualize its transformed output I_{trans} at the very first iteration of BluRef training. As shown in Fig. 3, the $\mathcal{DM}$ output, when masked by its confidence map, already reconstructs meaningful edges, building contours, and scene geometry that closely resemble the ground-truth sharp image. While the reconstruction is still imperfect and contains notable mismatches, the corresponding error map shows that a substantial portion of the scene already exhibits relatively low error with respect to the sharp image.

This indicates that the DM module is able to transfer a meaningful amount of sharp structural information from the reference frames even before any refinement or aggregation takes place. However, the incomplete and spatially inconsistent regions also highlight the necessity of our iterative pseudo–ground-truth aggregation strategy, which progressively stabilizes and improves the supervision signals used to train the deblurring network (see the gradual improvements across iterations in Fig. 2).

References

[1] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, 2017. 4

[2] Bang-Dang Pham, Phong Tran, Anh Tran, Cuong Pham, Rang Nguyen, and Minh Hoai. Hypercut: Video sequence from a single blurry image using unsupervised ordering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9843–9852, 2023. 4

[3] Bang-Dang Pham, Phong Tran, Anh Tran, Cuong Pham, Rang Nguyen, and Minh Hoai. Blur2blur: Blur conversion for unsupervised image deblurring on unknown domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 4

[4] Prune Truong, Martin Danelljan, and Radu Timofte. Glu-net: Global-local universal network for dense flow and correspondences. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6258–6268, 2020. 2

[5] Prune Truong, Martin Danelljan, Radu Timofte, and Luc Van Gool. Pdc-net+: Enhanced probabilistic dense correspondence network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 1

[6] Kai Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4791–4800, 2021. 1

	GoPro			RB2V		
Delta (Δ)	*1 frame*	*10 frames*	*20 frames*	*1 frame*	*10 frames*	*20 frames*
BluRef (Ours)						
Restormer - BluRef (Avg.)	27.12 / 0.905	27.04 / 0.895	26.98 / 0.893	25.41 / 0.816	25.38 / 0.812	24.78 / 0.801
Restormer - BluRef (Seq.)	28.46 / 0.923	28.37 / 0.920	28.31 / 0.912	25.22 / 0.810	25.20 / 0.811	24.73 / 0.792
Supervised - Upperbound						
NAFNet		33.32 / 0.962			28.54 / 0.824	
Restormer		32.92 / 0.961			27.43 / 0.849	

Table 5. Comparison of our proposed BluRef (Restormer backbone) on GoPro and RB2V datasets. For each test, we report PSNR↑/SSIM↑ scores as evaluation metrics.

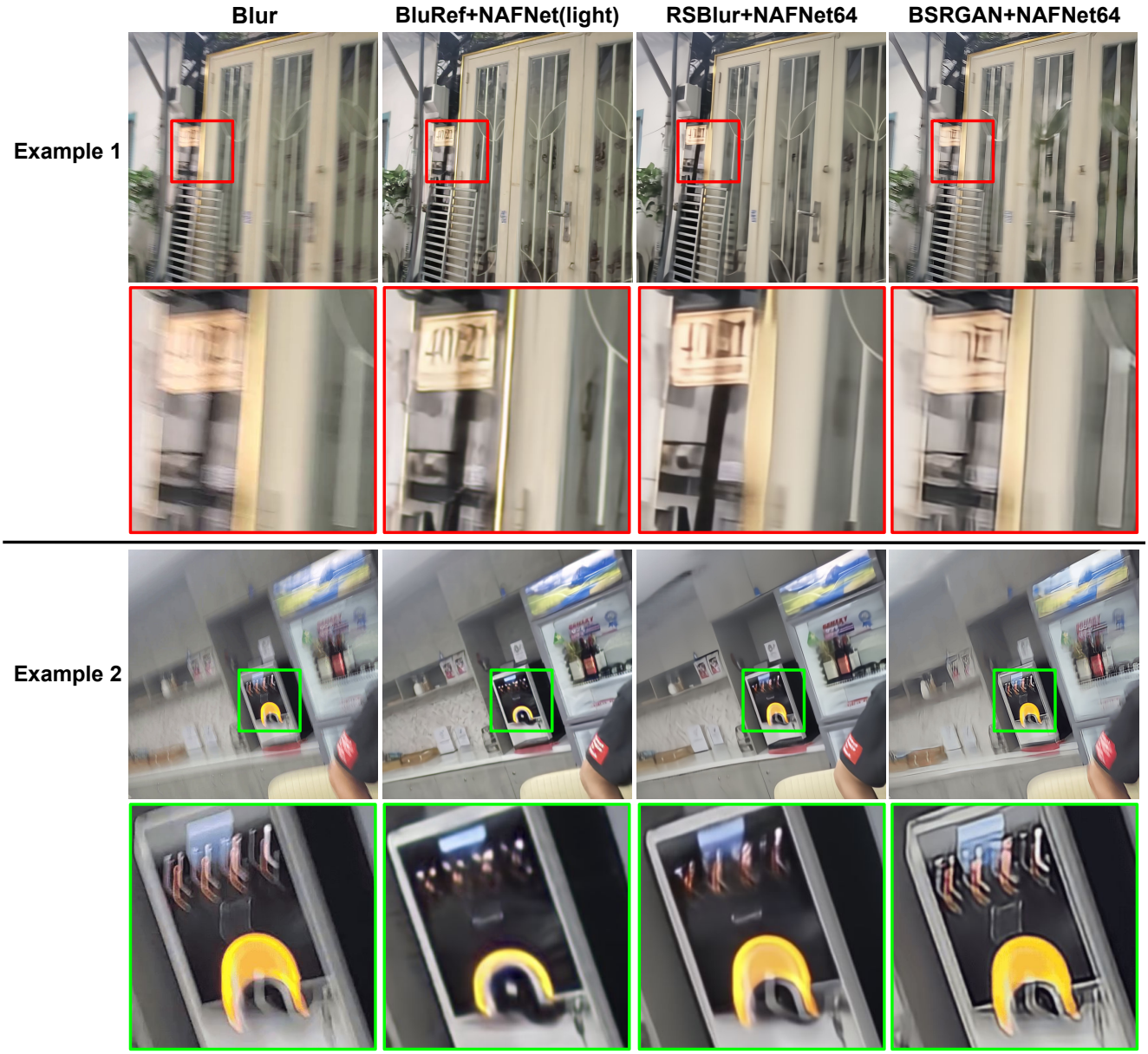

Figure 4. The Qualitative Results in PhoneCraft dataset.

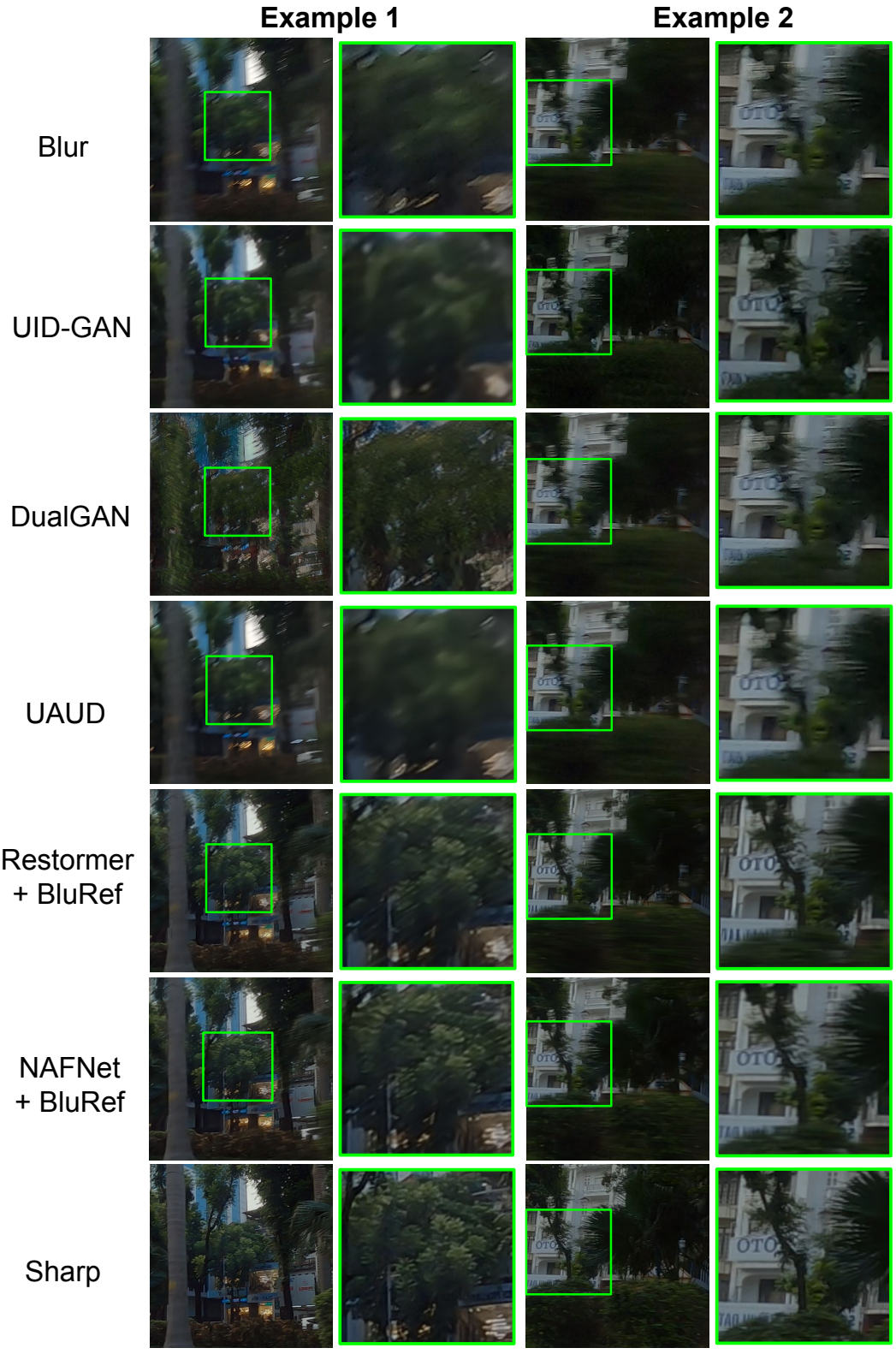

Figure 5. Addtional Qualitative Results in RB2V Dataset.

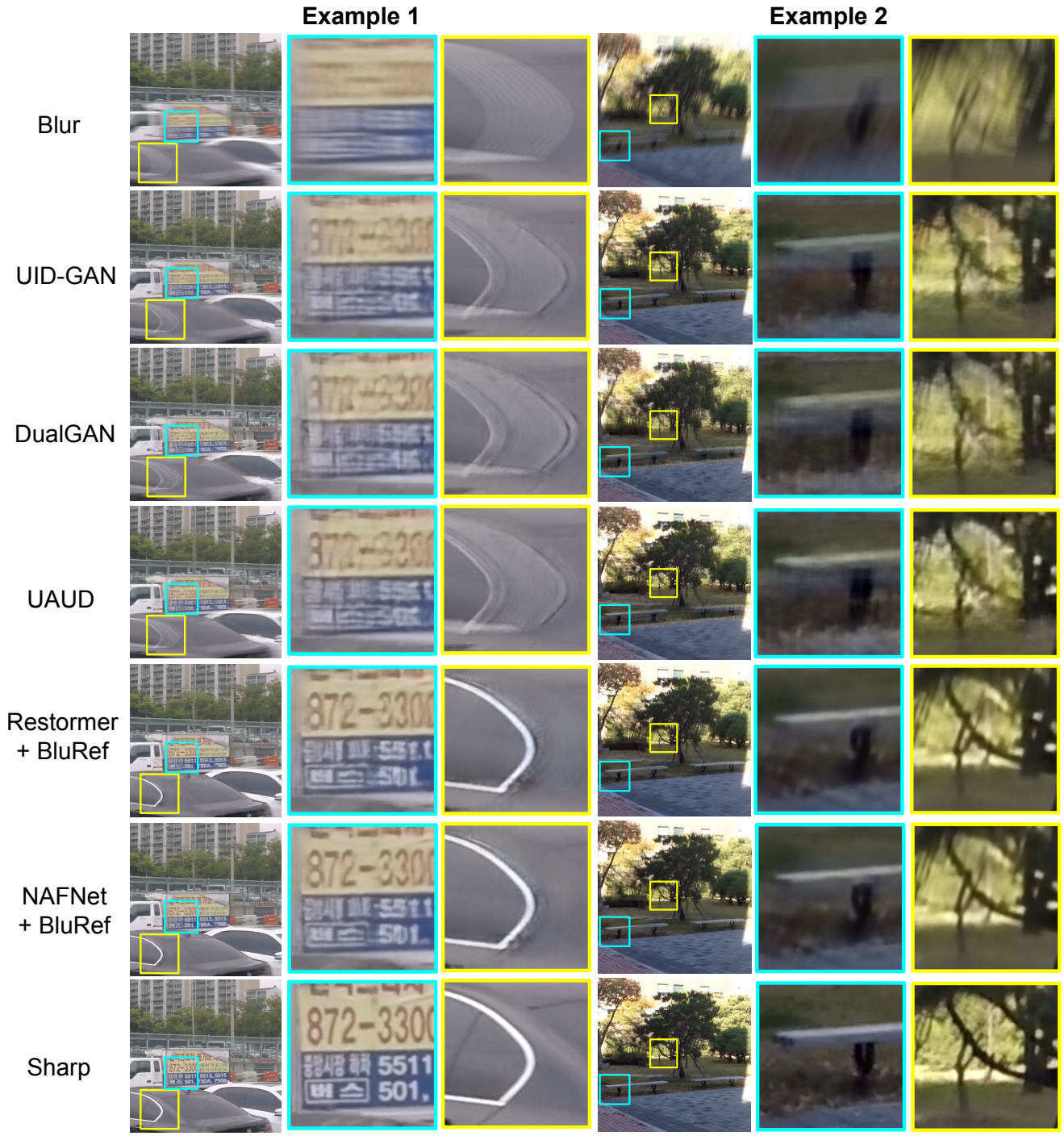

Figure 6. Addtional Qualitative Results in GoPro Dataset.