# Supplementary Material for “Hallucinated-IQA: No-Reference Image Quality Assessment via Adversarial Learning”

Kwan-Yee Lin[1] and Guanxiang Wang[2]
[1]Department of Information Science, School of Mathematical Sciences, Peking University
[2]Department of Mathematics, School of Mathematical Sciences, Peking University

[1]linjunyi@pku.edu.cn [2]gxwang@math.pku.edu.cn

## 1. Introduction

This supplementary file presents: (1) the pseudo codes of the whole training process of our approach; (2) additional experimental analysis and quantitative results on TID2013, TID2008, and CSIQ datasets; (3) more qualitative evaluations of the effectiveness of our key components.

## 2. Training Strategy

Algorithm 1 demonstrates the whole training process of our approach as the pseudo codes. We list again the core formulas that introduced in the main paper for convenience.

The loss function of the iqa-discriminator $D$ is formulated as

$$\max_{\omega} \mathbb{E}[\log D_{\omega}(\mathbf{I}_r)] + \mathbb{E}[\log(1 - |D_{\omega}(G_{\theta}(\mathbf{I}_d)) - \mathbf{d}_{fake}|)], \tag{1}$$

where

$$\mathbf{d}^{i}_{fake} = \begin{cases} 1 & \text{if } \|R(I^i_d, I^i_{sh}) - s^i\|_F < \epsilon \\ 0 & \text{if } \|R(I^i_d, I^i_{sh}) - s^i\|_F \geq \epsilon \end{cases}. \tag{2}$$

The adversarial loss of the quality-aware generative network $G$ is formulated as

$$\mathcal{L}_{adv} = \mathbb{E}[\log(1 - D_{\omega}(G_{\theta}(I_d)))], \tag{3}$$

and the overall loss function of $G$ is given by

$$\mathcal{L}_G = \mu_1 \mathcal{L}_p + \mu_2 \mathcal{L}_s + \mu_3 \mathcal{L}_{adv}. \tag{4}$$

The loss function of the hallucination-guided quality regression network $R$ is formulated as

$$\mathcal{L}_R = \frac{1}{T}\sum_{t=1}^{T} \|\mathcal{R}_2(f(\mathcal{H}_{m,n}(I_d)^t) \otimes (\mathcal{R}_1(I^t_d, I^t_{map}))) - s^t\|_{\ell 1}. \tag{5}$$

**Algorithm 1** The training process of our method.

```
Require: Training images: $\mathbf{I_d}$, the corresponding ground-truth references $\mathbf{I_r}$, and quality scores $\mathbf{s}$, initial parameters of $G$ netwrok $\hat{\theta}_G$, initial parameters of $R$ network $\hat{\gamma}_R$, initial parameters of $D$ network $\hat{\omega}_D$
1:  Forward $G$ by $\mathbf{I_{sh}} = G(\mathbf{I_d}, \hat{\theta}_G)$ , and calculate $\mathbf{I_{map}} = |I_d - I_{sh}|$
2:  Forward $R$ by $\hat{\mathbf{s}} = R(\mathbf{I_d}, \mathbf{I_{map}}, \mathbf{s})$
3:  while $\hat{\theta_G}$ has not converged do
4:      Calculate $\boldsymbol{d}_{fake}$ by Eq.2
5:      Forward $D$ by $\{\boldsymbol{d}_{real}\} = D(\mathbf{I_r})$, and optimise $D$ net by Eq.1;
6:      Forward $D$ by $\{\boldsymbol{d}_{fake}\} = D(\mathbf{I_{sh}})$, and optimise $D$ net by Eq.1;
7:      Optimise $G$ net by Eq.3;
8:      Forward $R$ by $\hat{\mathbf{s}} = R(\mathbf{I_d}, \mathbf{I_{map}}, \mathbf{s})$, and optimise $R$ net by Eq.5;
9:      Forward $G$ by $\mathbf{I_{sh}} = G(\mathbf{I_d}, \hat{\theta}_G)$, and optimise $G$ net by Eq.4;
10: end while
11: while $\hat{\gamma}_R$ has not converged do
12:     Forward $R$ by $\hat{\mathbf{s}} = R(\mathbf{I_d}, \mathbf{I_{map}}, \mathbf{s})$, and optimise $R$ net by Eq.5;
13: end while
14: return $\hat{\gamma}_R$
```

It should be noted that, during the training process, $R$ in step 4 and 8 is the same one that optimises in a mutually reinforce manner with $G$. While the regression network $R_{pre}$ used in the term $\mathcal{L}_s$ of step 9 is an independent one with fixed weights with no feature fusion from $G$.

## 3. Additional Experimental Results

**Evaluation Metrics.** The detailed definitions of the two performance metrics (*i.e.*, SROCC, LCC) we use in this pa-

| Methods | # 1 | # 2 | # 3 | # 4 | # 5 | # 6 | # 7 | # 8 | # 9 | # 10 | # 11 | # 12 | # 13 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| BIECON [3] | - | - | - | - | - | - | - | - | - | - | - | - | - |
| RankIQA [5] | 0.652 | 0.588 | 0.796 | 0.326 | 0.780 | 0.703 | 0.776 | **0.811** | 0.819 | 0.894 | **0.894** | **0.755** | **0.798** |
| Baseline | 0.384 | 0.216 | 0.701 | 0.086 | 0.591 | 0.358 | 0.587 | 0.740 | 0.769 | 0.565 | 0.455 | 0.048 | 0.305 |
| **Ours** | **0.940** | **0.920** | **0.942** | **0.440** | **0.960** | **0.820** | **0.837** | 0.781 | **0.949** | **0.933** | 0.602 | 0.318 | 0.704 |
| **Ours+Oracle** | 0.959 | 0.939 | 0.972 | 0.764 | 0.957 | 0.743 | 0.879 | 0.848 | 0.931 | 0.971 | 0.696 | 0.582 | 0.750 |
| Methods | # 14 | # 15 | # 16 | # 17 | # 18 | # 19 | # 20 | # 21 | # 22 | # 23 | # 24 | **ALL** | |
| BIECON [3] | - | - | - | - | - | - | - | - | - | - | - | 0.765 | |
| RankIQA [5] | 0.472 | **0.626** | 0.260 | **0.628** | **0.629** | 0.593 | **0.661** | **0.798** | 0.782 | **0.834** | **0.874** | 0.799 | |
| Baseline | 0.035 | -0.033 | -0.225 | -0.121 | 0.396 | 0.253 | 0.211 | 0.418 | 0.724 | 0.328 | 0.430 | 0.573 | |
| **Ours** | **0.649** | 0.165 | **0.353** | 0.538 | 0.054 | **0.890** | 0.404 | 0.768 | **0.892** | 0.474 | 0.600 | **0.880** | |
| **Ours+Oracle** | 0.878 | 0.536 | 0.824 | 0.881 | 0.527 | 0.911 | 0.555 | 0.857 | 0.881 | 0.553 | 0.716 | 0.939 | |

Table 1: Performance evaluation (LCC) on the entire TID2013 database.

per are as follows:

$$SROCC = 1 - \frac{6\sum_{t=1}^{T} d_t}{T(T^2-1)} \tag{6}$$

where $T$ is the number of distorted images, and $d_t$ is the rank difference between the ground-truth quality score and predicted score of image $t$.

$$LCC = \frac{\sum_{t=1}^{T}(s_t - \bar{s_t})(\hat{s_t} - \bar{\hat{s_t}})}{\sqrt{\sum_{t=1}^{N}(s_t - \bar{s_t})^2}\sqrt{\sum_{t=1}^{N}(\hat{s_t} - \bar{\hat{s_t}})^2}} \tag{7}$$

where $\bar{s_t}$ and $\bar{\hat{s_t}}$ denote the means of the ground truth and predicted score, respectively.

**Evaluation on TID2013.** Besides the SROCC results analysed in the main paper, we also evaluate the LCC results on the entire TID2013 dataset. The results in Table 1 lead to the following conclusions. Firstly, our approach achieves $10\%$ and $15\%$ relative improvements over the most state-of-the-arts RankIQA, and BIECON on ALL score, respectively. We also achieve superior results than RankIQA on most of individual distortions. It is interesting to observe that, although RankIQA synthesizes masses of ranked images according to the distortion types in a specific dataset, our model outperforms RankIQA on distortion types like $\#1 \sim \#8$ by significant margins. It may be because, comparing with providing plenty of similar samples, compensating perceptual discrepancy information is a more effective way for deep network to learn desirable feature representation. This observation demonstrates the effectiveness of our model. Secondly, our method performs significantly better than the baseline on almost all individual distortions. Simply utilizing deep network will easily lead to over-fitting problem due to the lack of samples, like the baseline (Res-18 Network) performs. This observation demonstrates the generalization ability of our model.

**Evaluation on TID2008.** We compare SROCC performance of our approach with the state-of-the-art methods, *i.e.*, BRISQUE [6], LBIQ [8], Tang *et al.* [9], and BIECON [3]. The results of previous three works are from the paper [9], and the results of BIECON are re-implemented using the codes provided in their homepage. As shown in Table 2, our method outperforms BRISQUE, LBIQ on most of individual distortions. Comparing with Tang *et al.* [9], our model obtain over $5\%$ improvments on more than half types, and achieves comparable results for the rest types, except distortion types like changing contrast, where the normalization operation in network leads to a certain degree

| Statistics | Methods | # 1 | # 2 | # 3 | # 4 | # 5 | # 6 |
|---|---|---|---|---|---|---|---|
| SROCC | BRISQUE [6] | 0.660 | 0.317 | 0.799 | -0.220 | 0.841 | 0.830 |
| | LBIQ [8] | 0.820 | 0.911 | 0.881 | 0.735 | 0.920 | 0.835 |
| | Tang [9] | 0.840 | **0.936** | 0.893 | 0.638 | 0.851 | 0.850 |
| | BIECON [3] | - | - | - | - | - | - |
| | **Ours** | **0.927** | 0.898 | **0.940** | **0.747** | **0.967** | **0.940** |
| | **Ours+Oracle** | 0.941 | 0.916 | 0.941 | 0.730 | 0.981 | 0.905 |
| | Methods | # 7 | # 8 | # 9 | # 10 | # 11 | # 12 |
| | BRISQUE [6] | 0.690 | 0.810 | 0.445 | 0.821 | 0.745 | 0.279 |
| | LBIQ[8] | 0.812 | 0.919 | 0.780 | 0.911 | 0.931 | 0.675 |
| | Tang [9] | **0.835** | **0.890** | 0.910 | 0.925 | **0.950** | **0.831** |
| | BIECON [3] | - | - | - | - | - | - |
| | **Ours** | 0.714 | 0.618 | **0.917** | **0.937** | 0.610 | 0.628 |
| | **Ours+Oracle** | 0.777 | 0.627 | 0.950 | 0.899 | 0.635 | 0.557 |
| | Methods | # 13 | # 14 | # 15 | # 16 | # 17 | **ALL** |
| | BRISQUE [6] | 0.740 | 0.130 | 0.316 | 0.305 | 0.091 | 0.610 |
| | LBIQ [8] | **0.775** | 0.150 | 0.449 | 0.270 | 0.581 | 0.740 |
| | Tang [9] | 0.750 | 0.600 | **0.720** | **0.425** | **0.765** | 0.841 |
| | BIECON [3] | - | - | - | - | - | 0.826 |
| | **Ours** | 0.381 | **0.733** | 0.331 | 0.275 | 0.357 | **0.941** |
| | **Ours+Oracle** | 0.460 | 0.833 | 0.759 | 0.686 | 0.696 | 0.952 |

Table 2: Performance evaluation (SROCC) on the entire TID2008 database.

| Statistics | Methods | # 1 | # 2 | # 3 | # 4 | # 5 | # 6 |
|---|---|---|---|---|---|---|---|
| LCC | BIECON [3] | - | - | - | - | - | - |
| | **Ours** | 0.920 | 0.873 | 0.925 | 0.836 | 0.970 | 0.947 |
| | **Ours+Oracle** | 0.918 | 0.884 | 0.928 | 0.822 | 0.964 | 0.881 |
| | Methods | # 7 | # 8 | # 9 | # 10 | # 11 | # 12 |
| | BIECON [3] | - | - | - | - | - | - |
| | **Ours** | 0.727 | 0.585 | 0.920 | 0.974 | 0.660 | 0.519 |
| | **Ours+Oracle** | 0.750 | 0.635 | 0.959 | 0.970 | 0.640 | 0.537 |
| | Methods | # 13 | # 14 | # 15 | # 16 | # 17 | **ALL** |
| | BIECON [3] | - | - | - | - | - | 0.835 |
| | **Ours** | 0.407 | 0.802 | 0.230 | 0.267 | 0.378 | **0.949** |
| | **Ours+Oracle** | 0.353 | 0.830 | 0.777 | 0.679 | 0.802 | 0.955 |

Table 3: Performance evaluation (LCC) on the entire TID2008 database.

| Type | Methods | LIVE |  | TID2008 |  | TID2013 |  | CSIQ |  |
|---|---|---|---|---|---|---|---|---|---|
|  |  | SROCC | LCC | SROCC | LCC | SROCC | LCC | SROCC | LCC |
| FR | PSNR | 0.876 | 0.872 | 0.553 | 0.573 | 0.636 | 0.706 | 0.806 | 0.800 |
|  | SSIM [10] | 0.948 | 0.945 | 0.775 | 0.773 | 0.637 | 0.691 | 0.913 | 0.899 |
|  | VIF [7] | 0.963 | 0.960 | 0.749 | 0.808 | 0.677 | 0.772 | 0.920 | 0.928 |
|  | FSIMc [13] | 0.960 | 0.961 | 0.884 | 0.876 | 0.851 | 0.877 | 0.931 | 0.919 |
|  | FR-DCNN [4] | 0.975 | 0.977 | - | - | - | - | - | - |
|  | DeepQA [2] | 0.981 | 0.982 | 0.947 | 0.951 | 0.939 | 0.947 | 0.961 | 0.965 |
| NR | BIECON [3] | 0.960 | 0.962 | 0.826 | 0.835 | 0.721 | 0.765 | 0.825 | 0.838 |
|  | RankIQA [5] | 0.981 | 0.982 | - | - | 0.780 | 0.799 | - | - |
|  | **Ours** | 0.982 | 0.982 | 0.941 | 0.949 | 0.879 | 0.880 | 0.885 | 0.910 |

Table 4: SROCC and LCC comparison with FR-IQA(full-reference) methods.

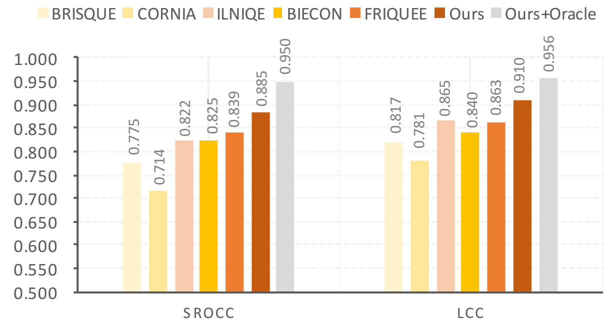


Figure 1: Performance evaluations (both SROCC and LCC) on the entire CSIQ database.

invariance to them. For ALL score, our approach achieves the highest results. Specifically, we obtain 14% improvement than BIECON and 12% improvement than Tang *et al.* [9]. We also compare LCC performance of our approach with BIECON. Our method obtains 14% improvement than BIECON, as demonstrated in Table 3.

**Evaluation on CSIQ.** We compare both SROCC and LCC performances of our approach on CSIQ with the state-of-the-art methods, *i.e.*, FRIQUEE [1], BRISQUE [6], CORNIA [11], BIECON [3], and ILNIQE [12]. As shown in Figure 1, we obtain about 5% relative improvements on SROCC and LCC. Similar conclusion as analyzing in other three datasets, our method could compensate effective perceptual discrepancy information unconstrained to specific definition or image content of a particular dataset, and therefore has the merit of robustness to various distortions.

**Comparison with FR-IQA methods.** In addition, as an extension of our work, we also compare our NR-IQA approach with the state-of-the-art FR-IQA methods, as shown in Table 4. Our approach achieves better results than most of state-of-the-art FR-IQA models. On LIVE dataset, our approach performs better than the most state-of-the-art FR-IQA method DeepQA [2]. As for rest datasets, we obtain comparable results with DeepQA and FSIMc, which are superior to other NR-IQA methods at this point.

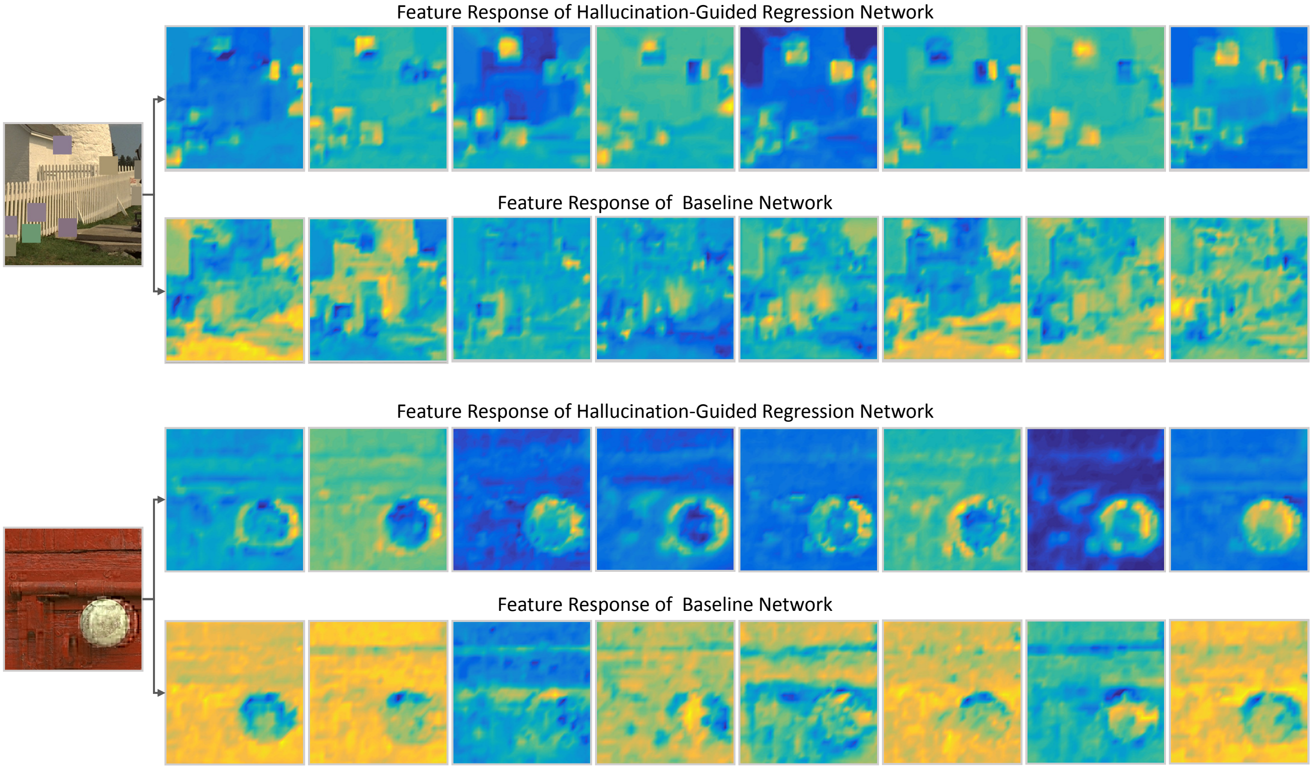
## 4. Qualitative Evaluations

In this section, we first show the feature responses of baseline model and our approach to different distorted images[1] to verify the effectiveness of perceptual discrepancy compensation qualitatively. Then, we present representative examples of hallucinated reference generation to demonstrate the effectiveness of our key components.

Figure 2 shows two examples of different feature response results of typical NR-IQA regression network and our approach. To illustrate the effectiveness of perceptual discrepancy compensation mechanism, we use baseline network (Res-18 Network with only distorted images as input) to represent typical NR-IQA network. We randomly sample eight feature maps from second residual block on the two networks respectively, and visualize the feature responses, where brighter values indicate higher responses. First example is an image under local block-wise distortions, and the second one is distorted by jpeg transmission errors. As shown in the Figure 2, the feature responses of baseline network tend to be rambling in both two examples. In contrast, the feature maps of the hallucination-guided regression network response strongly on distorted parts. This observation qualitatively demonstrates the perceptual discrepancy compensation mechanism could provide more specific and effective information to a deep network to learn desirable image quality feature representations comparing with typical NR-IQA regression network.

Figure 3 shows the qualitative comparison of the baseline generator, generator with quality-aware loss, and our final generator $G$ (*i.e.*,“IQA-GAN ”,which contains quality-aware loss and iqa-discriminator) on *common* distortion types, such as additive gaussian noise, spatially correlated noise, masked noise, gaussian blur *etc*. Figure 4 shows the qualitative comparison on *typical* distortion types, such as jpeg transmission errors, local block-wise distortions, jpeg2k compression *etc*. As can be seen from the two figures, with the quality-aware loss and IQA-GAN scheme

[1] The distorted images in the qualitative evaluations are chosen from TID2008 dataset.


Feature Response of Hallucination-Guided Regression Network

Feature Response of  Baseline Network

Feature Response of Hallucination-Guided Regression Network

Feature Response of  Baseline Network

Figure 2: Examples of feature response comparison of typical NR-IQA regression network with our Hallucination-Guided Regression Network for verifying the effectiveness of our hallucinated reference information compensation mechanism. (Best viewed in color)

adding, the hallucinated images are improved to be more and more clear and plausible, and $G$ performs well robustness to various distortions. This leads to highly effective discrepancy maps that can further benefit the final quality prediction with precise distortion information (*e.g.*, location, form) captured.

## References

[1] S. Bianco, L. Celona, P. Napoletano, and R. Schettini. On the use of deep learning for blind image quality assessment. *CoRR*, 2016.

[2] J. Kim and S. Lee. Deep learning of human visual sensitivity in image quality assessment framework. In *CVPR*, 2017.

[3] J. Kim and S. Lee. Fully deep blind image quality predictor. *J. Sel. Topics Signal Processing*, 11(1):206–220, 2017.

[4] Y. Liang, J. Wang, X. Wan, Y. Gong, and N. Zheng. Image quality assessment using similar scene as reference. In *ECCV*, 2016.

[5] X. Liu, J. van de Weijer, and A. D. Bagdanov. Rankiqa: Learning from rankings for no-reference image quality assessment. In *ICCV*, 2017.

[6] A. Mittal, A. K. Moorthy, and A. C. Bovik. No-reference image quality assessment in the spatial domain. *TIP*, pages 4695–4708, 2012.

[7] H. R. Sheikh and A. C. Bovik. Image information and visual quality. pages 430–444, 2006.

[8] H. Tang, N. Joshi, and A. Kapoor. Learning a blind measure of perceptual image quality. In *CVPR*, 2011.

[9] H. Tang, N. Joshi, and A. Kapoor. Blind image quality assessment using semi-supervised rectifier networks. In *CVPR*, 2014.

[10] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *TIP*, 13(4):600–612, 2004.

[11] P. Ye, J. Kumar, L. Kang, and D. Doermann. Unsupervised feature learning framework for no-reference image quality assessment. In *CVPR*, 2012.

[12] L. Zhang, L. Zhang, and A. C. Bovik. A feature-enriched completely blind image quality evaluator. *TIP*, 24(8):2579–2591, 2015.

[13] L. Zhang, L. Zhang, X. Mou, and D. Zhang. FSIM: A feature similarity index for image quality assessment. *TIP*, 20(8):2378–2386, 2011.

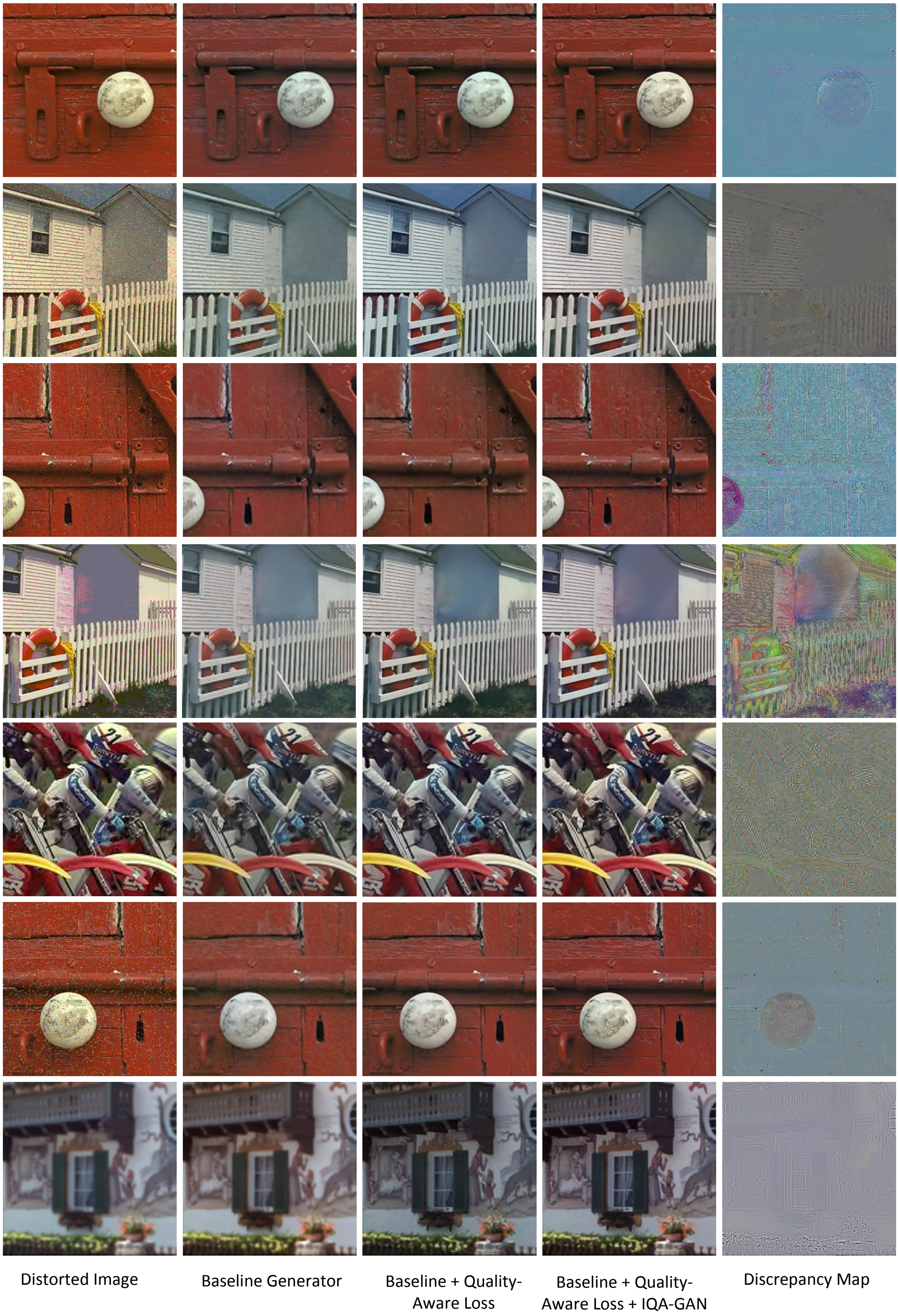

Distorted Image
Baseline Generator
Baseline + Quality-Aware Loss
Baseline + Quality-Aware Loss + IQA-GAN
Discrepancy Map

Figure 3: Additional qualitative comparison of hallucinated reference generation for verifying the effectiveness of key components of $G$ on *common* distortion types. (Best viewed in color)

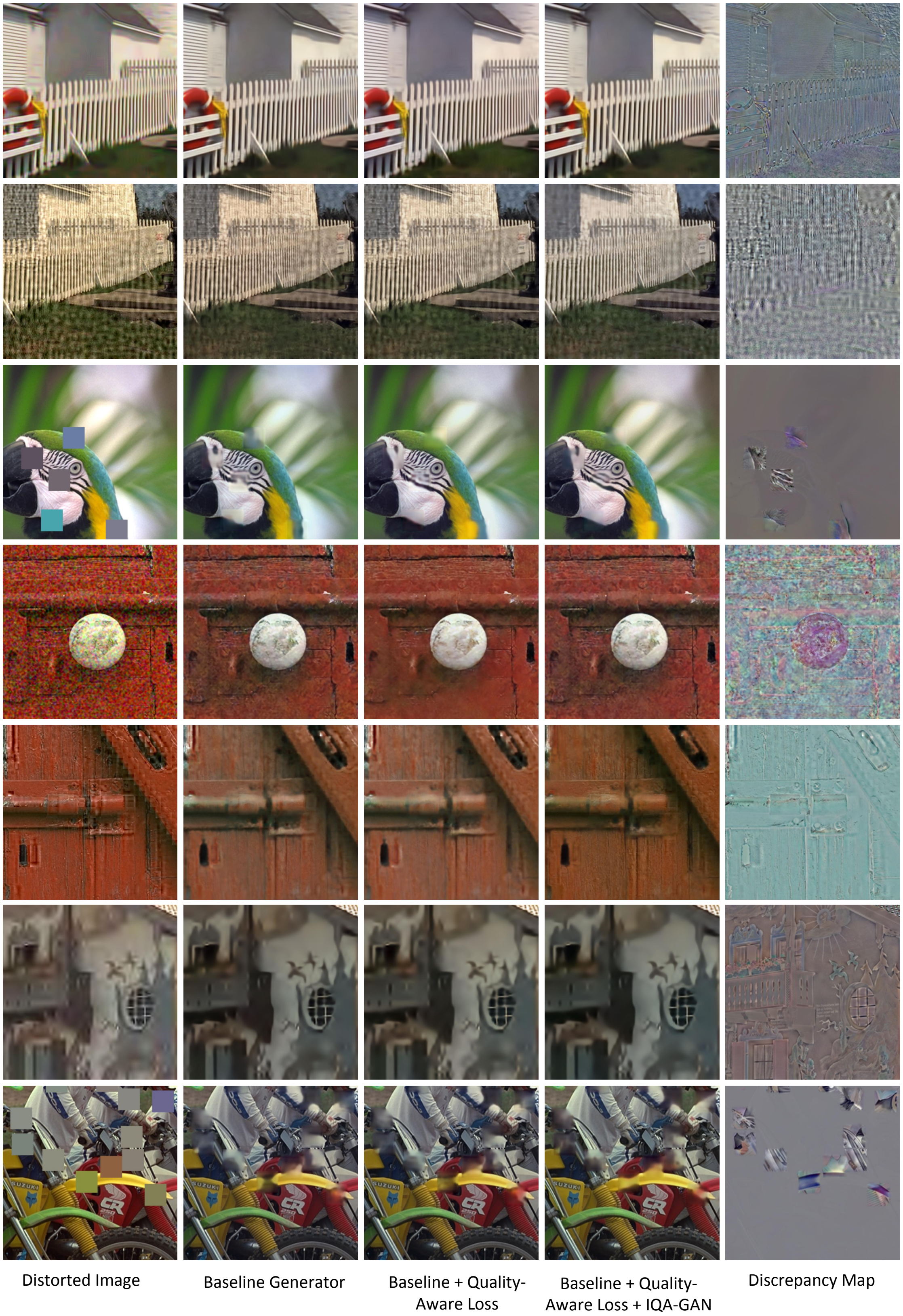

Distorted Image
Baseline Generator
Baseline + Quality-Aware Loss
Baseline + Quality-Aware Loss + IQA-GAN
Discrepancy Map

Figure 4: Additional qualitative comparison of hallucinated reference generation for verifying the effectiveness of key components of $G$ on *typical* distortion types. (Best viewed in color)