A Supplementary Material

We provide additional materials to supplement our main paper. Sec. A.1 provides additional ablation studies on the image patch size (Sec. A.1.1), the batch size used training (Sec. A.1.2). Sec. A.2 evaluates on six additional benchmarks where we also achieve state-of-the-art results. In Sec. A.3 we provide additional qualitative examples for the application of our trained model as a training objective for the super-resolution task. Sec. A.4 reports the licenses of the datasets used in our experimental work. Finally, Sec. A.5 discusses potential societal impacts and limitations of our paper.

A.1 Ablations

A.1.1 Training image patch size

We assess the effect of the patch size used during training. For example, LPIPS [62] relies on a patch size of 64×64 while DISTS [11] considers 256×256. Tab. S1 shows that the larger the patch size, the better the performance. This concurs with our assumption that image content matters. A larger patch size captures more image content whereas a smaller one only provides a local view. Increasing the patch size beyond 256×256 creates memory issues, and might not be optimal given that it reduces the variety of samples seen during training. Following Ding *et al*. [11], we then adopt a patch size of 256×256 in our experiments.

Patch	LIVE[45]			CSIQ[21]			TID2013[40]		
size	PLCC	SRCC	KRCC	PLCC	SRCC	KRCC	PLCC	SRCC	KRCC
64	0.936	0.945	0.792	0.943	0.948	0.795	0.823	0.796	0.605
128	0.949	0.959	0.820	0.949	0.955	0.810	0.879	0.870	0.681
256	**0.964**	**0.969**	**0.843**	**0.957**	**0.960**	**0.824**	**0.915**	**0.907**	**0.731**

Table S1: **Image patch size** used for training. A larger patch size captures more image content and results in improved performance.

A.1.2 Training batch size

We assess the effect of the batch size during training. For the experiment, we vary the batch size, and keep the number of epochs similar. Tab. S2 shows that the bigger the batch size, the better the performance. A bigger batch size increases the number of comparisons that are made at every training iteration. Thus, the benefit of all pairwise and listwise comparisons becomes much more effective. Increasing the batch size above 64 creates memory issues (with current hardware). As such, we adopt a batch size of 64 in our experiments.

A.2 Quantitative evaluation

Setup. First, We assess the ability to rank image pairs in a two-alternative forced choice (2AFC) setting, using the BAPPS [62] dataset of 26,904 image pairs. Second, we compare on four datasets with task-specific artifacts arising from image restoration and enhancement algorithms: Liu13 [29] for deblurring, Ma17 [31] for super-resolution, Min19 [34] for dehazing, and Tian18 [52] for rendering. They contain 1200, 1620, 600 and 140 distorted images, respectively. Third, we evaluate on the PIPAL dataset [13], which was used in the recent NTIRE'21 and '22 challenges [14].

Batch size	LIVE[45]			CSIQ[21]			TID2013[40]		
	PLCC	SRCC	KRCC	PLCC	SRCC	KRCC	PLCC	SRCC	KRCC
8	0.949	0.955	0.811	0.944	0.952	0.804	0.887	0.880	0.694
16	0.953	0.959	0.821	0.944	0.952	0.803	0.899	0.893	0.711
32	0.957	0.964	0.831	0.952	0.956	0.813	0.909	0.902	0.722
64	**0.964**	**0.969**	**0.843**	**0.957**	**0.960**	**0.824**	**0.915**	**0.907**	**0.731**

Table S2: **Batch size** used for training over 10 epochs. A bigger batch size increases the the number of comparisons, which results in improved performance.

Method	Colori-zation	Video deblur.	Frame interp.	Super-resolution	All
Human	0.688	0.671	0.686	0.734	0.695
PSNR	0.624	0.590	0.543	0.642	0.614
SSIM [56]	0.522	0.583	0.548	0.613	0.617
MS-SSIM [55]	0.522	0.589	0.572	0.638	0.596
VSI [61]	0.597	0.591	0.568	0.668	0.622
MAD [21]	0.490	0.593	0.581	0.655	0.599
VIF [44]	0.515	0.594	0.597	0.651	0.603
FSIMc [60]	0.573	0.590	0.581	0.660	0.615
NLPD [20]	0.528	0.584	0.552	0.655	0.600
GMSD [58]	0.517	0.594	0.575	0.676	0.613
PieAPP [41]	0.594	0.582	0.598	0.685	0.626
LPIPS [62]	0.625	**0.605**	**0.630**	0.705	**0.641**
DISTS [11]	**0.627**	0.600	0.625	**0.710**	**0.641**
Ours	**0.632**	**0.605**	**0.631**	**0.712**	**0.645**

Table S3: **Comparison on image pair ranking** in a 2AFC setting on the BAPPS[62] dataset, with top-2 results highlighted in bold. While we train with diverse-content and holistic properties, our method is on par with LPIPS trained specifically on this task.

For the 2AFC experiment, we follow Zhang *et al.* [62] and report the 2AFC score $qp + (1-q)(1-p)$, where q is the ground truth probability and p is the model prediction.

A.2.1 Pairwise comparisons

Tab. S3 compares our method using the BAPPS [62] dataset in a two-alternative forced choice (2AFC) setting. Given two 64×64 image patches with different distortions, yet originating from the same reference image, the 2AFC task consists of correctly predicting which patch has the best quality. Following Ding *et al.* [11], we report results on the test set of distortions, that originate from real algorithms for colorization, video deblurring, frame interpolation and super-resolution. As LPIPS has been trained on a separate set of BAPPS, it provides a strong baseline. Even though this evaluation relies on comparing images with similar content, our method yields the best overall performance; despite being trained on the broader task of comparing differing image content.

A.2.2 Task-specific distortions

Tab. S4 further compares our approach on distortions arising in specific tasks: Liu13 [29] on deblurring, Ma17 [31] on super-resolution, Min19 [34] on dehazing, and Tian18 [52] on rendering. In other words, each dataset is task-specific as they only include distortions originating from methods related to the respective tasks. For example, Liu13 provides mean opinion scores for images pertaining to five different deblurring methods. A similar behavior to the broad distortions results may be observed here; consistent performance across all datasets is challenging. For example, the performance suddenly drops for LPIPS and PieApp methods on Tian18 rendering images. They tend to struggle with the domain gap present in this dataset and cannot quantify distortions, regardless of the image domain. In this context,

Method	Liu13[29] (deblurring)			Ma17[31] (SR)			Min19[34] (dehazing)			Tian18[52] (rendering)		
	PLCC	SRCC	KRCC	PLCC	SRCC	KRCC	PLCC	SRCC	KRCC	PLCC	SRCC	KRCC
PSNR	0.807	0.803	0.599	0.611	0.592	0.414	0.754	0.740	0.555	0.605	0.536	0.377
SSIM [56]	0.763	0.777	0.574	0.654	0.624	0.440	0.715	0.692	0.513	0.420	0.230	0.156
MS-SSIM [55]	0.899	0.898	0.714	0.815	0.795	0.598	0.699	0.687	0.503	0.386	0.396	0.264
VSI [61]	0.919	0.920	0.745	0.736	0.710	0.514	0.730	0.696	0.511	0.512	0.531	0.363
MAD [21]	0.901	0.897	0.714	0.873	0.864	0.669	0.543	0.605	0.437	0.690	0.622	0.441
VIF [44]	0.879	0.864	0.672	0.849	0.831	0.638	0.740	0.667	0.504	0.429	0.259	0.173
FSIMc [60]	0.923	0.921	0.749	0.769	0.747	0.548	0.747	0.695	0.515	0.496	0.476	0.324
NLPD [20]	0.862	0.853	0.657	0.749	0.732	0.535	0.616	0.608	0.442	0.594	0.463	0.316
GMSD [58]	0.927	0.918	0.746	0.861	0.851	0.661	0.675	0.663	0.489	0.631	0.479	0.329
PieAPP [41]	0.752	0.786	0.583	0.791	0.771	0.591	0.749	0.725	0.547	0.352	0.298	0.207
LPIPS [62]	0.853	0.867	0.675	0.809	0.788	0.687	**0.825**	0.777	0.592	0.387	0.311	0.213
DISTS [11]	**0.940**	**0.941**	**0.784**	**0.887**	**0.878**	**0.697**	0.816	**0.789**	**0.600**	**0.694**	**0.671**	**0.485**
Ours	**0.937**	**0.941**	**0.786**	**0.898**	**0.883**	**0.700**	**0.819**	**0.801**	**0.607**	**0.720**	**0.662**	**0.482**

Table S4: **Comparison on task-specific distortions** with top-2 results highlighted in bold. While learning-based method generalize well across datasets, they struggle when there is a domain gap as in image rendering. DISTS and our method achieve a consistent performance.

only DISTS and our method can provide consistent and high correlation scores on all four datasets.

A.2.3 GAN distortions

We provide additional quantitative comparisons with LPIPS [62] and DISTS [11] on the training split of PIPAL [13]. The NTIRE'21 challenge [14] relied on the PIPAL dataset as it provides more recent and challenging distortions coming for example from GAN-based restotation methods. As the ground truth labels are not available for the testing split, we instead rely on the training split for evaluation. Tab. S5 shows that our training scheme outperforms the original DISTS on this challenging dataset. Note that none of the models evaluated in Tab. S5 have been trained on any splits of PIPAL (*i.e.*, this is a similar setting to experiments in Tab. 4 or Tab. S4).

Method	PIPAL train split[13]		
	PLCC	SRCC	KRCC
LPIPS [62]	0.611	0.573	0.405
DISTS [11]	0.592	0.579	0.408
Ours	**0.701**	**0.671**	**0.484**

Table S5: **Comparison on more challenging distortions** from the PIPAL dataset. Our training scheme improves upon the original DISTS and LPIPS in all metrics.

A.3 Qualitative application results

We provide additional qualitative results when using our trained model as a training objective for a super-resolution application. Similar to the main paper, we select images from Set14 [59] while ERSGAN models are trained on DIV2K [3]. Figure S1 shows other benefits of our model: we notably observe a better enhancement of shapes (*e.g.*, letters or zebra stripes) and the absence of an unwanted color cast, that may appear in grayscale images when employing an alternative perceptual loss. This additional brief exploration further illustrates the potential to use our approach as an objective for downstream tasks, such as

Figure S1: **Qualitative application** of our image quality assessment as a training objective for a $\times 4$ super-resolution task, using Set14 [59] images. Best viewed in color with digital zoom. We evaluate different training objectives for ESRGAN [53] (columns 2–4) and compare with a naive bicubic interpolation (column 1). *Top row:* While still not perfect, letter shapes appear more realistic with our perceptual loss. *Middle row:* This is confirmed on the zebra stripes. *Bottom row:* Moreover, compared with perceptual loss from Johnson *et al.* [16], we do not exhibit a color cast and can handle a grayscale image.

super-resolution, where perceptual quality is of prime importance.

A.4 Dataset licenses and implementation

We rely on nine image quality assessment datasets: one for training and eight for evaluation. No new dataset has been collected in this paper, and datasets we rely on for our experiments have been previously published: Kadid-10k [25] mentions the Pixabay license; other datasets (LIVE [45], CSIQ [21], TID [40], BAPPS [62], Liu13 [29], Ma17 [31], Min19 [34], Tian18 [52]) provide a copyright for research usage (*i.e.*, non-commercial purposes). Future work will aim to provide implementation under popular learning frameworks *e.g.* [1, 2, 37].

A.5 Societal impact and limitations

Automating image quality assessment may induce the eradication of related manual tasks, that currently constitute valid paid work for humans. However we concur with Bhardwaj *et al.* [4]; human verification is likely to continue to be required in both the short and medium term. Given the difficulty of image quality assessment, human verification remains the gold standard.

Learning-based perceptual image metrics typically rely on manually-rated images as training data. Such data include human rater variance, and may reflect the biases of human annotators. Reducing or removing the dependency on human labels provides one direction towards mitigating this particular form of bias.

Finally, as highlighted in the paper, image content matters in image quality assessment. An important question could be to assess whether some contents create a systematic source of bias either from the annotations (*i.e.*, data bias) or the learned metrics (*i.e.*, model bias).