KonIQ++: Boosting No-Reference Image Quality Assessment in the Wild by Jointly Predicting Image Quality and Defects

Shaolin Su[1]
shaolin_su@mail.nwpu.edu.cn
Vlad Hosu[2]
vlad.hosu@uni-konstanz.de
Hanhe Lin[3]
h.lin2@rgu.ac.uk
Yanning Zhang[1]
ynzhang@nwpu.edu.cn
Dietmar Saupe[2]
dietmar.saupe@uni-konstanz.de

[1] School of Computer Science and Engineering
Northwestern Polytechnical University
Xi'an, China

[2] Department of Computer and Information Science
University of Konstanz
Konstanz, Germany

[3] National Subsea Centre
Robert Gordon University
Aberdeen, UK

Abstract

Although image quality assessment (IQA) in-the-wild has been researched in computer vision, it is still challenging to precisely estimate perceptual image quality in the presence of real-world complex and composite distortions. In order to improve machine learning solutions for IQA, we consider side information denoting the presence of distortions besides the basic quality ratings in IQA datasets. Specifically, we extend one of the largest in-the-wild IQA databases, KonIQ-10k, to KonIQ++, by collecting distortion annotations for each image, aiming to improve quality prediction together with distortion identification. We further explore the interactions between image quality and distortion by proposing a novel IQA model, which jointly predicts image quality and distortion by recurrently refining task-specific features in a multi-stage fusion framework. Our dataset KonIQ++, along with the model, boosts IQA performance and generalization ability, demonstrating its potential for solving the challenging authentic IQA task. The proposed model can also accurately predict distinct image defects, suggesting its application in image processing tasks such as image colorization and deblurring.

1 Introduction

The quality of visual media is an essential component of a viewer's experience when watching or interacting with media content. Accurate computational image quality assessment models provide an opportunity to scale up quantitative visual inspection in image processing, computer graphics, and computer vision [20]. For example, IQA models are applied in image compression, enhancement, and summarization. During the image acquisition process, IQA models can provide suggestions or dynamically tune camera settings to improve

the focus, exposure, or colors of shots. Image search can benefit as well from quality assessment as a filtering principle. In computer vision, object recognition and detection are affected by quality impairment. Identifying or correcting such distortions can lead to more reliable and accurate predictive models. Overall, accurate IQA opens up many possibilities for research.

There are multiple sub-types of IQA in the literature. According to the availability of pristine reference images, an IQA model can be classified as full-reference (FR), reduced-reference (RR), and no-reference (NR). NR-IQA is more challenging and practical than FR-IQA and RR-IQA since reference images are not always available in applications [21], and images that are captured in the wild contain composite distortions that differ from those in the artificial distorted images [9]. Therefore, NR-IQA in-the-wild has been drawing increasing attention in the computer vision community.

The human visual system is sensitive to many physical attributes of images such as distortion and contrast. It has been demonstrated that the performance of image quality prediction can be boosted when trained together with these attributes. For example, Liu et al. [10] proposed the RankIQA model. It first trains a Siamese network to rank the quality of a reference image and its distorted versions according to the corresponding physical distortion parameter. The quality score of a given image is estimated by fine-tuning the pretrained Siamese network. Fan et al. [2] used a CNN network for image distortion identification and multi-expert CNN networks for distortion-specific quality predictions, where the final quality of a given image is estimated by fusing the outputs of these networks. Similarly, Ma et al. [11] proposed the MEON model, which contains two sub-networks, a distortion identification network and a quality prediction network, with shared weights in early layers. By training large-scale image data for distortion identification, followed by fine-tuning for quality prediction, it allows training a deep network on small IQA databases. The aforementioned methods, however, deal with synthetic IQA tasks, and are not suitable for IQA in-the-wild. In real-world scenarios, images suffer from more complex and composite distortions. Recently, Fang et al. [3] proposed a multi-task model to simultaneously predict qualities and EXIF tags/image attribute scores/scene category labels of smartphone images captured in the wild. Though they showed an IQA accuracy improvement by learning from multiple labels, the approach is still limited for IQA in-the-wild. It only applies to images captured by smartphones, and a simple multi-head prediction model is adopted for learning image attributes. Image quality and its attributes show a much more complex interaction during the perception process. The latter has not been explored in the authentic IQA field yet.

In this paper, we explore the effectiveness of jointly learning to predict both the quality and distortion in images. We first extend one of the largest in-the-wild IQA datasets, KonIQ-10k, to KonIQ++, by collecting additional distortion annotations (Sec. 2). We further propose a novel IQA model to jointly predict image quality and distortion by recurrently refining task specific features (Sec. 3). Experiments show that our approach boosts quality prediction and generalization ability, demonstrating its potential for solving the challenging problem of NR-IQA in-the-wild (Sec. 4). Database and codes are made available [1] [2].

[1] Database is available at http://database.mmsp-kn.de

[2] Codes are available at https://github.com/SSL92/koniqplusplus

2 KonIQ-10k Dataset with Distortion Annotations

We extend the IQA dataset KonIQ-10k, which is one of the largest in-the-wild and authentically distorted datasets [3, 7], and the most diverse of its kind. The augmented set, KonIQ++, provides additional labels corresponding to the presence and types of visible degradations. There are larger in-the-wild IQA databases in terms of number of images (KonPatch-30k [17] and PaQ-2-PiQ [22]), however, these consist of image crops of relatively small size.

2.1 Brief description of KonIQ++

The data collection for KonIQ++ was performed on the same platform as that for the original KonIQ-10k, previously known as *CrowdFlower*™ or *Figure-Eight*™ and nowadays owned by *Appen*™. The experimental study is similar to [6] where the classes of degradations used during the annotation were introduced first. Thus, in addition to annotating a quality rating, the participants in the KonIQ++ study were also asked for the presence of any visible degradation, and the degradation type. There are four broad classes of degradations: blurs, artifacts, color, and contrast. Participants could select multiple classes per image. Blurs includes all types of low-pass distortions such as lens, motion, Gaussian blurs; artifacts include compression and super-resolution artifacts, and also noise and grain; degradations relating to color can be over/under-saturation, color fringing, *etc.*; contrast includes problems with either global and local contrast, *i.e.*, sharpness. Each participant was required to provide a quality score on a 5-point absolute category ratings (ACR) scale, and binary labels denoting the presence of any degradation, one for each type of degradation.

2.2 Quality control for KonIQ++

As was done for the KonIQ-10k database, we used the same test images with known ground-truth for the quality control. We first computed the annotation accuracy on test images, and participants that had an overall accuracy below 50% were disqualified, and their ratings were discarded, leaving 72.8 ratings per image on average. We then used several user rating screening methods, similar to the ones applied for creating the KonIQ-10k database. These include removing users whose ratings had a low correlation with the global average rating, known as the mean opinion score (MOS) (below 0.5 Pearson linear correlation coefficient, PLCC), line clickers and random clickers as defined in [6]. After the screening procedures, 40 to 69 ratings per image remained, 57.9 ratings per image on average. 2482 of the initial 2844 participants passed both the test images and the screening procedures. Even though a large number of users were removed, the screening procedures changed the MOS very little, the Spearman rank-order correlation coefficient (SRCC) between the raw MOS (before screening) and the screened MOS was 0.995.

We obtained an excellent correlation between the MOS of the two screened experiments of 0.960 PLCC and 0.948 SRCC. This suggests a very high replicability of the experiments.

2.3 Aggregating annotation scores

For every observed image, subjects each provided a 5-point rating for the technical quality, from *bad* (1) to *excellent* (5), and an annotation for the type of distortion which was conditioned on the presence of any distortion. Per rating for an image, each of the classes *presence, blurs, artifacts, colors, and contrast* received a binary label of either *present* (1) or

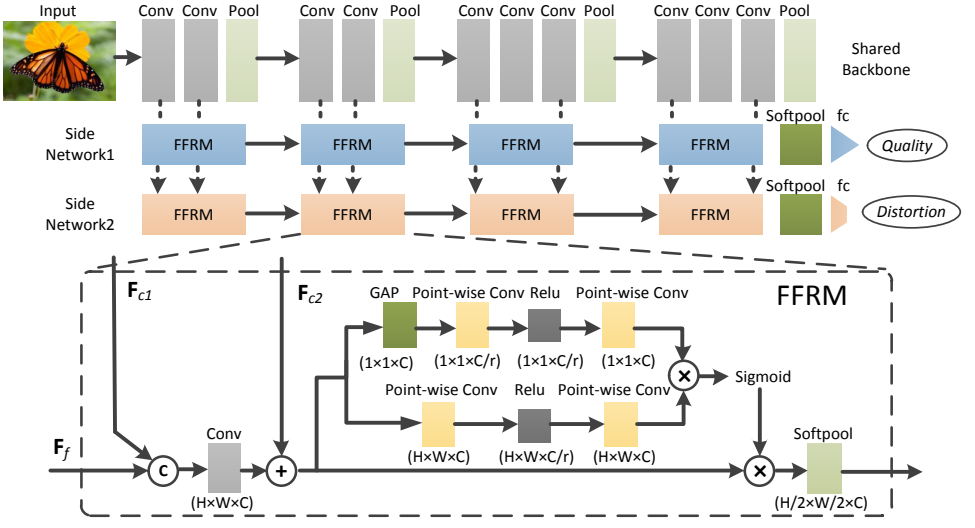

Figure 1: Our proposed model architecture for simultaneously predicting image quality and distortion. It jointly predicts image quality and distortion through two side networks which recurrently fuse and refine multi-stage image features.

absent (0). This yields MOS (in $[0, 1]$) also for each of the distortion classes. We define the distortion magnitude of an image as the mean of the MOS over the four degradation classes (*blurs, artifacts, colors, contrast*). The annotations on degradations, along with quality rating scores form the KonIQ++ database. Nonetheless, to enable us to compare models that are trained on KonIQ-10k scores, we did all the model training and testing by using the distortion labels from KonIQ++ and the quality ratings from KonIQ-10k. It is also worth noting that there is a negative correlation of -0.926 SRCC between the distortion magnitude and the quality MOS. This suggests that the quality rating cannot be derived entirely from the degradation amount, as showcased in the supplement.

3 Learning to Jointly Predict Quality and Distortion

We make use of the new KonIQ++ dataset to train models which jointly predict image quality and distortion. Generally, the perceived image quality is negatively correlated with the amount of detected distortions. However, there are exceptions in real world scenarios (see supplement). In order to precisely capture distinct quality and distortion features, as well as to understand their mutual relationship, we propose a novel model. To this end, we introduce two separate side networks which recurrently fuse and refine image features. In the following subsections, we describe the overall framework, the details including the feature fusion and refining operation, and the implementation.

3.1 Overall Model Framework

The overall model architecture is shown in Figure 1. In our model, we introduce two side networks expanded from a shared backbone network to capture image distortion and quality in a separate manner. The two side networks serve to extract distortion and quality specific

features, and they contribute together to refine the backbone network for learning general representations. Within each side network, we propose to fuse the multi-scale features from the backbone for distortion and quality prediction. In order to refine the extracted features to be task specific, we further propose a feature fusion and refining module (FFRM), which enables the two side networks to learn distortion and quality specific representations, respectively. In the last part of each side network, we extract the feature maps from the last FFRM and apply global soft pooling [14] to them for a vector representation $\tilde{a} = [\tilde{a}_1, \tilde{a}_2, ..., \tilde{a}_C]$, where C denotes the total channel number. Each element $\tilde{a}_c$ in $\tilde{a}$ is calculated by:

$$\tilde{a}_c = \sum_{i \in \mathbf{R}} w_c^i \cdot a_c^i, \quad w_c^i = \frac{e^{a_c^i}}{\sum_{j \in \mathbf{R}} e^{a_c^j}} \tag{1}$$

where a_c^i denotes the activation value of the cth channel feature indexed by i, and $\mathbf{R}$ represents the global spatial region of the feature map.

At last, fully connected layers are used to regress the vectors into the final quality score or distortion magnitude.

3.2 Feature Fusion and Refining Module

In our model, the feature fusion and refining module fulfills two functions: fusing multi-stage features and refining the features to be either distortion or quality specific. As illustrated in the bottom of Figure 1, each FFRM receives three input feature maps: the first map represents the features after the first convolution in each backbone block $\mathbf{F}_{c1}$, the second the features before the pooling operation in each backbone block $\mathbf{F}_{c2}$, and the last one is the output from the previous FFRM block $\mathbf{F}_f$. The first FFRM receives only $\mathbf{F}_{c1}$ and $\mathbf{F}_{c2}$.

When fusing multi-stage features, we first concatenate $\mathbf{F}_{c1}$ with $\mathbf{F}_f$, then apply one convolution and add $\mathbf{F}_{c2}$ to the result. After fusing the three types of features, we refine the fused feature map by sending it via two pathways. One of them calculates the channel attention map by global average pooling followed by two point-wise convolutions with a squeeze ratio r. The other conducts similar point-wise convolutions but preserves the spatial resolution of the input feature map without global average pooling. The two pathways serve to extract a global and a local feature map respectively, and the outputs are then multiplied to form a channel and spatial resolution preserved attention map, which further refines the fused feature by point-wise multiplication. At last, we adopt soft pooling [14] to the refined features to adjust their sizes matching with the next FFRM stage.

3.3 Implementation Details

In our implementation, we select three base architectures including ResNet-50 [5], EfficientNet-b4 [16], and ResNeXt-101 [18], with the pre-trained weights on the ImageNet [1] as candidate backbones, to observe their performances on the collected dataset. The two side networks share the same configurations, concretely, we set the convolution kernel size to be 3 in all FFRMs, and the numbers of channels in four FFRMs are set to be 64, 64, 128, and 256, respectively. The squeeze ratios r in the two pathways of FFRM are all set to 2. Since the feature representations are already refined to a certain extent in former FFRMs, we use only one fully connected layer to regress the pooled vectors into quality or distortion scores.

According to the collected quality and distortion label, we set the output size of two side networks to be 1 and 4 respectively, the quality prediction side network predicts a

one-dimensional quality score, and the distortion prediction side network predicts a four-dimensional vector, the components of which indicating the detected distortions including artifacts, blur, contrast, and colors. We sum the loss of both quality and distortion values for joint training by

$$\ell = \alpha L(s,\widehat{s}) + \frac{\beta}{4}\sum_{k=1}^{4} L(d(k),\widehat{d}(k)), \tag{2}$$

where $L(\cdot)$ denotes the loss function, $s = [s_1, s_2, ..., s_n]$ and $\widehat{s} = [\widehat{s}_1, \widehat{s}_2, ..., \widehat{s}_n]$ represent the predicted quality scores and ground truth scores in a mini-batch of size n. Similarly, $d(k)$ and $\widehat{d}(k)$ are also n-dimensional vectors representing the k-th type of predicted and ground truth distortion amounts. α and β are weighting factors which are both set to 1 by default.

Due to the performance boost brought by the "norm-in-norm" loss function L_{NiN}, proposed in [8], we apply it for training our models.

$$L_{\text{NiN}}(x,y) = \frac{1}{2^p N^{1-\frac{p}{q}}} \left\| \frac{x-\bar{x}\mathbb{1}}{\|x-\bar{x}\mathbb{1}\|_q} - \frac{y-\bar{y}\mathbb{1}}{\|y-\bar{y}\mathbb{1}\|_q} \right\|_p^p, \tag{3}$$

where $x-\bar{x}\mathbb{1}$ is the mean-removed vector, i.e., $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$, $\mathbb{1} = (1\cdots 1)^T$, and likewise for y. For the norms, we used $p = 1$ and $q = 2$, the best reported values in [8].

During training, images were resized to 480×640 resolution, and the batch size was $n = 8$ for all experiments. We trained the model for 25 epochs using the Adam optimizer. The learning rates were first set to 1×10^{-4} for the side networks, and 1×10^{-5} for the backbone network, then divided by 10 after every 10 epochs. Our model was implemented in PyTorch and the experiments were conducted on NVIDIA 1080Ti GPUs.

4 Experiments

4.1 Setup

We followed the same strategy conducted in [7] to evaluate model performances, where the 10,073 images contained in KonIQ-10k/KonIQ++ are split into a training subset (7,058 images), a validation subset (1,000 images) and a test subset (2,015 images). We calculated the SRCC on the validation set to select the best performing model. The selected model was used for the following evaluations on the test set. SRCC and PLCC were used for evaluating model performance.

4.2 Effectiveness of joint learning

In order to investigate the effectiveness of joint learning, we first compared models trained with single quality scores and with joint quality and distortion annotations. The compared models include: backbone model trained on single quality score by adding four fully connected layers behind, denoted as BB-s; backbone model trained on joint annotations, with four similarly added fully connected layers but with 5 outputs (1 for quality and 4 for distortions), denoted as BB-j; and backbone model integrated with the proposed side networks, denoted as BB-p. The results are shown in Table 1.

We make the following observations. First, despite using the same model architecture, models trained with joint annotations (BB-j) performed slightly better than those with a single quality label (BB-s). The result verifies that models make better predictions under joint

supervision, consistent with previous observations [8]. Second, though trained with similar joint annotations, our proposed model (BB-p) consistently outperformed BB-j, indicating that the proposed model architecture learns a better relationship between image quality and distortion than using simple multi-head predictors. Last, we observe that when ResNeXt-101 was selected as backbone, our model performed the best. Therefore, we chose ResNeXt-101 along with the proposed side networks as the final model for the following evaluations.

Backbone	ResNet-50			EfficientNet-b4			ResNeXt-101		
Model	BB-s	BB-j	BB-p	BB-s	BB-j	BB-p	BB-s	BB-j	BB-p
SRCC	0.9167	0.9170	0.9229	0.9246	0.9251	0.9316	0.9342	0.9350	0.9396
PLCC	0.9280	0.9293	0.9347	0.9387	0.9397	0.9439	0.9428	0.9436	0.9483

Table 1: Comparison of backbone models trained on single quality score (BB-s), on joint annotations but use simple multi-head predictors (BB-j), and on the proposed model (BB-p).

4.3 Comparison with SOTA IQA models

In this section, we compare our model with several state-of-the-art models on the authentic IQA task. We selected two authentic IQA datasets, *i.e.*, KonIQ-10k test dataset and LIVE Challenge (CLIVE) [4] dataset for evaluation, and show the results in Table 2. The competing models are BLIINDS-II [13], BRISQUE [12], HOSA [19], KonCept512 [7], HyperIQA [15], and Li's model [8]. All competing models were trained on the official KonIQ-10k training subset, and the model selection was done according to the performance on the KonIQ-10k validation subset. As shown, our approach outperformed the other models on both datasets. Our model not only achieved better prediction accuracy on the KonIQ-10k test dataset, but also a better generalization ability on the independent test dataset CLIVE.

The improvements over Li's method [8] for KonIQ are small, yet statistically significant by Student's t-test. The resulting p-value based on 10 re-training and testing runs for each model was 0.0153 for SRCC and 0.0052 for PLCC, where $p < 0.05$ implies statistical significance.

Dataset	Criterion	BLIINDS-II	BRISQUE	HOSA	KonCept512	HyperIQA	Li's	Ours
KonIQ	SRCC	0.585	0.705	0.805	0.921	0.917	0.938	**0.940**
	PLCC	0.598	0.707	0.828	0.937	0.923	0.947	**0.948**
CLIVE	SRCC	0.090	0.561	0.628	0.825	0.778	0.836	**0.840**
	PLCC	0.107	0.598	0.668	0.848	0.798	0.852	**0.854**

Table 2: Comparison with SOTA IQA methods on the KonIQ-10k test subset and the whole CLIVE dataset. Our proposed approach outperforms competing models on both datasets. The models are all trained on the official KonIQ-10k training subset, and the model selection is done on the validation subset.

4.4 Evaluation on distortion prediction

Since our model is also capable of predicting image distortion, we compare it with several existing models purposed for detecting multiple image distortions. We selected Yu's model [23], which was designed to detect multiple photographic defects, and the model MT-A from [8], used to predict image quality along with five image attributes (brightness, colorfulness, contrast, noisiness, and sharpness) for comparison. We trained and tested the models on the new KonIQ++ dataset, following all the same setups, and report the results in Table 3. We

also compare model performances when trained only on distortion annotations (Model_d), and on joint quality and distortion annotations ($\text{Model}_{\text{d+q}}$), to observe if image quality clues assists in image distortion prediction. Output sizes of all models were adapted for training and testing on the proposed dataset.

From Table 3, we observe that for all the competing models, distortion prediction accuracy improved when jointly trained with quality scores (except one for artifacts prediction on our proposed model). In addition, our proposed model outperformed the others on 7 out of 8 criteria, demonstrating its superiority. It is also interesting to note that when jointly trained, there were only slight improvements for Yu's and our model, but quite large gains for MT-A. This may be due to the separate prediction branches used in Yu's and our models are learning task specific features, each branch does not benefit much from the other, while for MT-A, the shared encoder and regressor are able to learn better representations with an extra quality clue.

Defects	Criterion	Yu's$_\text{d}$	Yu's$_{\text{d+q}}$	MT-A$_\text{d}$	MT-A$_{\text{d+q}}$	Ours$_\text{d}$	Ours$_{\text{d+q}}$
Artifacts	SRCC	0.6646	0.6772	0.6026	0.7209	**0.7255**	0.7234
	PLCC	0.7688	0.7743	0.6923	0.7982	0.8181	**0.8226**
Blur	SRCC	0.8247	0.8288	0.8228	0.8590	0.8708	**0.8727**
	PLCC	0.8465	0.8480	0.8173	0.8669	0.8866	**0.8875**
Contrast	SRCC	0.5928	0.5969	0.6220	**0.6783**	0.6697	0.6718
	PLCC	0.6474	0.6484	0.6260	0.6949	0.7123	**0.7177**
Colors	SRCC	0.5916	0.5989	0.6078	0.6690	0.6767	**0.6770**
	PLCC	0.7142	0.7161	0.6887	0.7427	0.7822	**0.7847**

Table 3: Performance comparison for distortion prediction models on KonIQ++.

4.5 Cross test with SPAQ

As mentioned above, a recent similar work, *i.e.*, the SPAQ dataset and several assessment models were proposed [3] for predicting quality and distortions of images captured by smartphones. In this subsection, we compare with their model and dataset by conducting cross tests. We trained our model on the KonIQ++ training subset and tested on SPAQ for image quality and attribute prediction. Similarly, we took their pretrained model on SPAQ and tested on the KonIQ++ test set. In our experiment, the MT-A model trained on SPAQ was selected for testing, and performances are evaluated on the commonly labelled attributes in both datasets, *i.e.*, image quality, blur, contrast, and colors. We also tested the two models, which were trained on their corresponding proposed datasets, on a third authentic IQA dataset, CLIVE, for a further comparison.

Model/ Dataset	Criterion	SPAQ				KonIQ++				CLIVE
		Quality	Blur	Contrast	Colors	Quality	Blur	Contrast	Colors	Qualtiy
MT-A/ SPAQ	SRCC	–	–	–	–	0.728	0.690	0.385	0.430	0.752
	PLCC	–	–	–	–	0.758	0.663	0.352	0.416	0.776
Our model/ KonIQ++	SRCC	0.875	0.843	0.603	0.503	–	–	–	–	**0.840**
	PLCC	0.873	0.854	0.576	0.545	–	–	–	–	**0.854**

Table 4: Cross-tests with the SPAQ dataset and their model. The first column indicates the model used for evaluation and the dataset it was trained on. The last nine columns correspond to the test dataset and the evaluated attributes.

We show the results in Table 4, and can find that when cross testing, our model achieved a high predictive performance for quality and blur labels on SPAQ, while MT-A performed worse on our KonIQ++ dataset. When tested on the third IQA dataset CLIVE, our model

Figure 2: Heatmaps from distortion and quality prediction side networks of the model. The distortion prediction side network mainly focuses on degraded regions, while the quality prediction side network focuses more on salient regions of the main subject.

also outperformed MT-A (trained on SPAQ) by a large margin. We attribute the superior performance of models trained on the KonIQ++ dataset to two main factors. The first is the larger diversity of images in the original KonIQ-10k dataset in contrast to only camera-phone pictures contained in SPAQ, which are domain-specific and more limited regarding the types of distortions present. The second factor is the way we have classified the distortion types. In KonIQ++, each distortion class tolerates a wide range of sub-types, whereas in SPAQ, the definition of the distortion classes are more narrow. However, it is also noticeable that both models did not perform very well on contrast and colors prediction, leaving this problem open for future research.

4.6 Visualizing features for quality and distortion predictions

Since we use two separate side networks for predicting image quality and defects, in this subsection, we show feature heatmaps from the two branches to visualize the roles they play in the model. We extracted features from the last FFRM module and visualize the heatmaps by using the approach proposed in [24], the results are shown in Figure 2.

It appears that the two branches are extracting task specific features, showing different regions of high activations. For instance, the distortion prediction branch responds strongly to either out-of-focus regions in the top row or motion blur regions in the second row, while the quality prediction branch focuses more on salient regions of the subject, such as the in-

focus part of the squirrel, and the main body of the racing car. It is also interesting to find that when the image is barely distorted (see the bottom row), the distortion prediction branch shows overall low activations (except for the over-sharpened building roof and the apparently under-exposed part of the sky), while the quality prediction branch regards most of the scene regions contributing to image quality. The results, which are consistent with human perception, further demonstrate the effectiveness and generality of our proposed approach.

4.7 Which images benefit most from the joint prediction?

In this subsection, we are interested to find out which kind of images benefit most from the joint prediction with distortion annotations. We first select ResNeXt-101-s and ResNeXt-101-p mentioned in section 4.2, and calculate prediction errors of the two models on test images in the KonIQ-10k dataset, according to MOS labels. Compared with prediction errors made by ResNeXt-101-s, images whose prediction errors made by ResNeXt-101-p are reduced over a certain threshold are selected as "benefited images". In our experiment, predicted image ratings range from 0 to 100 , and the threshold is set to 5. We show representative examples in Figure 3.

Figure 3: Examples of images which benefit most from the joint prediction. Both images contain composite distortions including blur and artifacts (noise). A possible explanation could be that without distortion prediction, the model mistakes high frequency artifacts present in blurry regions as sharp structures with good quality. When the model is trained with distortion annotations, it correctly recognizes the composite distortions, thus leads to more accurate quality predictions. The images are best viewed when zoomed in.

5 Conclusion

We proposed to boost the challenging authentic NR-IQA task by jointly learning image quality and distortion. We first extended one of the largest IQA datasets in-the-wild, KonIQ-10k, into KonIQ++, by introducing a new set of subjective labels describing the mixture of distortions. We then explored the interaction between quality and distortion and further proposed a deep model for jointly predicting multiple image attributes. Experiments conducted in both self- and cross-database scenarios validated that the model not only surpasses existing authentic IQA models, but also greatly improves the prediction of image distortions.

6 Acknowledgement

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 251654672 – TRR 161 (Project A05) and National Engineering Laboratory for Integrated Aero-Space-Ground-Ocean Big Data Application Technology.

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009.

[2] Chunling Fan, Yun Zhang, Liangbing Feng, and Qingshan Jiang. No reference image quality assessment based on multi-expert convolutional neural networks. *IEEE Access*, 6:8934–8943, 2018.

[3] Yuming Fang, Hanwei Zhu, Yan Zeng, Kede Ma, and Zhou Wang. Perceptual quality assessment of smartphone photography. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3677–3686, 2020.

[4] Deepti Ghadiyaram and Alan C Bovik. Massive online crowdsourced study of subjective and objective picture quality. *IEEE Transactions on Image Processing*, 25(1): 372–387, 2016.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[6] Vlad Hosu, Hanhe Lin, and Dietmar Saupe. Expertise screening in crowdsourcing image quality. In *Tenth IEEE International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–6, 2018.

[7] Vlad Hosu, Hanhe Lin, Tamas Sziranyi, and Dietmar Saupe. KonIQ-10k: An ecologically valid database for deep learning of blind image quality assessment. *IEEE Transactions on Image Processing*, 29:4041–4056, 2020.

[8] Dingquan Li, Tingting Jiang, and Ming Jiang. Norm-in-norm loss with faster convergence and better performance for image quality assessment. In *28th ACM International Conference on Multimedia*, pages 789–797, 2020.

[9] Hanhe Lin, Vlad Hosu, and Dietmar Saupe. KADID-10k: A large-scale artificially distorted IQA database. In *Eleventh IEEE International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–3. IEEE, 2019.

[10] Xialei Liu, Joost Van De Weijer, and Andrew D Bagdanov. RankIQA: Learning from rankings for no-reference image quality assessment. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1040–1049, 2017.

[11] Kede Ma, Wentao Liu, Kai Zhang, Zhengfang Duanmu, Zhou Wang, and Wangmeng Zuo. End-to-end blind image quality assessment using deep neural networks. *IEEE Transactions on Image Processing*, 27(3):1202–1213, 2018.

[12] A Mittal, A. K. Moorthy, and A. C. Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing*, 21(12):4695, 2012.

[13] Michele A Saad, Alan C Bovik, and Charrier Christophe. Blind image quality assessment: a natural scene statistics approach in the dct domain. *IEEE Transactions on Image Processing*, 21(8):3339, 2012.

[14] Alexandros Stergiou, Ronald Poppe, and Grigorios Kalliatakis. Refining activation downsampling with softpool. *arXiv preprint arXiv:2101.00440*, 2021.

[15] Shaolin Su, Qingsen Yan, Yu Zhu, Cheng Zhang, Xin Ge, Jinqiu Sun, and Yanning Zhang. Blindly assess image quality in the wild guided by a self-adaptive hyper network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3667–3676, 2020.

[16] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning (ICML)*, pages 6105–6114, 2019.

[17] Oliver Wiedemann, Vlad Hosu, Hanhe Lin, and Dietmar Saupe. Disregarding the big picture: Towards local image quality assessment. In *2018 Tenth International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–6. IEEE, 2018.

[18] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1492–1500, 2017.

[19] Jingtao Xu, Peng Ye, Qiaohong Li, Haiqing Du, Yong Liu, and David Doermann. Blind image quality assessment based on high order statistics aggregation. *IEEE Transactions on Image Processing*, 25(9):4444–4457, 2016.

[20] Qingsen Yan, Dong Gong, Qinfeng Shi, Anton van den Hengel, Chunhua Shen, Ian Reid, and Yanning Zhang. Dual-attention-guided network for ghost-free high dynamic range imaging. *International Journal of Computer Vision*. doi: 10.1007/s11263-021-01535-y.

[21] Qingsen Yan, Dong Gong, and Yanning Zhang. Two-stream convolutional networks for blind image quality assessment. *IEEE Transactions on Image Processing*, 28(5): 2200–2211, 2018.

[22] Zhenqiang Ying, Haoran Niu, Praful Gupta, Dhruv Mahajan, Deepti Ghadiyaram, and Alan Bovik. From patches to pictures (PaQ-2-PiQ): Mapping the perceptual space of picture quality. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3575–3585, 2020.

[23] Ning Yu, Xiaohui Shen, Zhe Lin, Radomir Mech, and Connelly Barnes. Learning to detect multiple photographic defects. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1387–1396, 2018.

[24] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2921–2929, 2016.