Text-Guided Patch Scoring and Local Distortion Guidance for Image Quality Assessment

Supplementary Material

6. Experimental Settings

Implementation details. The experiments are conducted using a single A6000 GPU on PyTorch 1.12.0. We use the Adam optimizer to train our model with a batch size of 32. The learning rate is set to 0.002 and adjusted using a cosine-annealing scheduler. We train for 30 epochs except for FLIVE [40] and AVA [24], which contains a sufficiently large number of images. For FLIVE and AVA, we use 5 epochs. The performance is evaluated using the weights that achieved the best performance in the training epochs. The input image is resized to 448×448, except for KADID-10k [21], where the original resolution is retained to preserve the synthetic distortion signals as much as possible. We use CLIP-ViT-L/14 from OpenAI [26] for our frozen CLIP model. We set w_{min} and h_{min} to 40, w_{max} and h_{max} to 150 in Eq. (8). We set $N = 6$ and $M = 18$ for RandAugment [6]. All experiments are conducted based on IQA-PyTorch [3] code. We normalize the Mean Opinion Score (MOS), the ground truth of IQA datasets, to [0,1] for calculating the loss. In Tab. 1, we present the reproduced results of MANIQA on all datasets. MANIQA does not provide the performance and experimental details for SPAQ and FLIVE, which include images at various resolutions—some even lower than 224×224. Since MANIQA crops images to a fixed size of 224×224, we follow the CLIP-IQA's approach [33] that resizes the shorter side of each image to 512.

Datasets. We evaluate the performance of our models on various datasets: KonIQ-10k [13], which is an in-the-wild dataset from YFCC100m [32]; SPAQ [9], which contains 11K images gathered from smartphones; CLIVE [11], which is also an in-the-wild dataset, and includes 1K images; AVA [24], which is an image aesthetic assessment dataset, and contains about 236K images; FLIVE [40], which is another in-the-wild dataset, and contains 160K images from AVA [24], VOC [8], EMOTIC [15], and CERTH Blur [23]; KADID-10k [21], which is a synthetic dataset; CSIQ [18], which is another synthetic dataset, and contains about 1K images; LIVE [28], which is also synthetic dataset, and contains 779 distorted images; and AGIQA-3K [19], which is an AI-generated image dataset.

Metrics Descriptions. PLCC (Pearson's linear correlation coefficient) measures the linear relationship between the predicted score and the ground truth, where a positive correlation approaches 1 and a negative correlation approaches -1. The equation is as follows:

$$\text{PLCC} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}}, \tag{12}$$

where x_i and y_i indicate the i–th predicted score and the ground truth, respectively. $\bar{x}$ and $\bar{y}$ denote the mean of the predicted score and the ground truth. N is the number of the images.

SRCC (Spearman's rank-order correlation coefficient) measures the rank relationship between the predicted score and the ground truth. If the ranks of the predicted scores and the ground truth match exactly, SRCC is 1; if the ranks are upside down, SRCC is -1. The equation is as follows:

$$\text{SRCC} = 1 - \frac{6\sum_{i=1}^{N} d_i^2}{N(N^2-1)}, \tag{13}$$

where d_i denotes the rank gap between the i–th predicted score and ground truth.

7. Text Prompt Tuning

We conduct the ablation study to examine the effects of text prompt tuning on TeMu-IQA-I. In Tab. 5, option (1) denotes that text prompt tuning is not applied. Options (2) and (3) represent the text prompt tuning of CoOp [47] and CoCoOp [46], respectively. The text prompt tuning from CoCoOp shows the best performance on KonIQ-10k [13] and CLIVE [11].

	KonIQ-10k		CLIVE	
Option	SRCC	PLCC	SRCC	PLCC
(1)	0.944	0.955	0.873	0.903
(2)	0.937	0.952	0.870	0.899
(3)	**0.946**	**0.957**	**0.911**	**0.932**

Table 5. Results on KonIQ-10k and CLIVE for different text prompt tuning methods with TeMu-IQA-I. Options are (1) no tuning, (2) CoOp, and (3) CoCoOp. **Bold** indicates best results

8. AI-Generated Image

Recent advancements in image generation have led to the development of AGIQA-3K [19], a dataset designed for assessing the quality of generated images. We train our model

Method	AGIQA-3K	
	SRCC	PLCC
CLIPIQA+	0.843	0.888
HyperIQA	0.843	0.901
CLIP-AGIQA	0.875	0.919
TeMu-IQA-S	<u>0.896</u>	<u>0.933</u>
TeMu-IQA-I	**0.898**	**0.934**

Table 6. Results on AGIQA-3K. **Bold** and <u>underline</u> indicate best and second best results.

on AGIQA-3K and compare its performance with existing methods. We average the results across 10 random splits. All results presented in Tab. 6 except for TeMu-IQA are derived from CLIP-AGIQA [10]. As shown in Tab. 6, our models surpass the performance of CLIP-IQA+ [33], HyperIQA [30], and CLIP-AGIQA on AGIQA-3K.

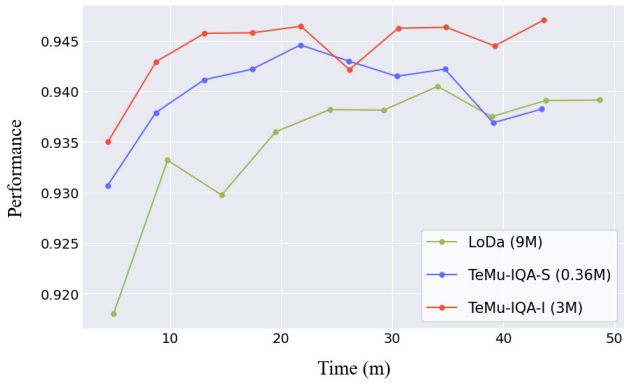


Figure 6. Performance comparison between TeMu-IQA and LoDa on KonIQ-10k for 10 epochs. The x-axis represents cumulative time for training and evaluation, while the y-axis denotes the average of SRCC and PLCC.

9. Training Time Comparison

We compare the performance and training time of TeMu-IQA-S and TeMu-IQA-I with those of LoDa [37], which contains only 9M trainable parameters. For a fair comparison, we use ViT-B/16, the backbone of LoDa. Specifically, we use a ViT pretrained on multi-modal data. By contrast, LoDa uses a ViT pretrained on ImageNet-21K [29]. We use the train–test split of LoDa and train models on KonIQ-10k for 10 epochs. The data points in Fig. 6 indicate the performance and cumulative time recorded at the end of training and evaluation for each epoch. As shown in Fig. 6, our models show faster training and outperform LoDa. In LoDa, a pretrained CNN and ViT are frozen, and trainable extractors and injectors are built to allow ViT to integrate information from low-level to high-level features from CNN. However, LoDa encounters challenges when applied to a ViT pretrained on multi-modal data, highlighting the need for further research. We believe that our study serves as an initial step toward addressing these challenges.

Method	KonIQ-10k		FLIVE	
	SRCC	PLCC	SRCC	PLCC
CLIP-IQA+	0.895	0.909	0.606	0.641
TOPIQ-ResNet50	0.928	0.941	0.633	0.722
DEIQT	0.921	0.934	0.571	0.663
ATT-IQA	0.942	0.952	0.632	0.742
TeMu-IQA-S	0.943	0.954	0.766	0.797
TeMu-IQA-S†	0.940	0.954	0.765	0.797
TeMu-IQA-I	<u>0.946</u>	<u>0.957</u>	**0.770**	<u>0.798</u>
TeMu-IQA-I†	**0.948**	**0.959**	<u>0.767</u>	**0.799**

Table 7. Results on KonIQ-10k and FLIVE with "†" models and previous works. **Bold** and <u>underline</u> indicate best and second best results.

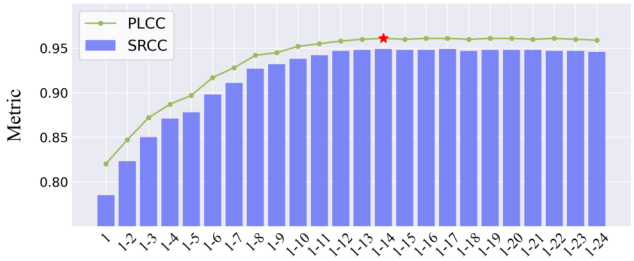


Figure 7. Layer optimization. Using the features up to the 14th layer shows the best performance on KonIQ-10k.

10. Layer Optimization

Fig. 8 shows the performance of TeMu-IQA-S trained on KonIQ-10k using the patch features from each single layer of CLIP-ViT on CSIQ [18], KADID-10k [21], KonIQ-10k test set [13], LIVE [28], CLIVE [11], and SPAQ [9] using the Q-ALIGN [36] train–test split. As shown in Fig. 8, it is hard to guarantee consistently optimal performance across all datasets using only a specific layer. For example, the 13th layer that performs best on KonIQ-10k shows lower performance on KADID-10k, where the best performance is achieved with the 20th layer. Therefore, using multi-level features can minimize discrepancies among datasets. In the main paper, we utilize patch features and [**cls**] tokens from all layers for TeMu-IQA. Here, we aim to investigate whether using only up to a specific layer may yield more effective results. As shown in Fig. 7, TeMu-IQA-I, which uses the patch features and the [**cls**] tokens up to the 14th layer, achieves optimal performance on KonIQ-10k. In Tabs. 7 and 8, our models marked with the "†" symbol use up to the 14th layer, whereas models without the "†" symbol use all layers. Although the "†" models demon-

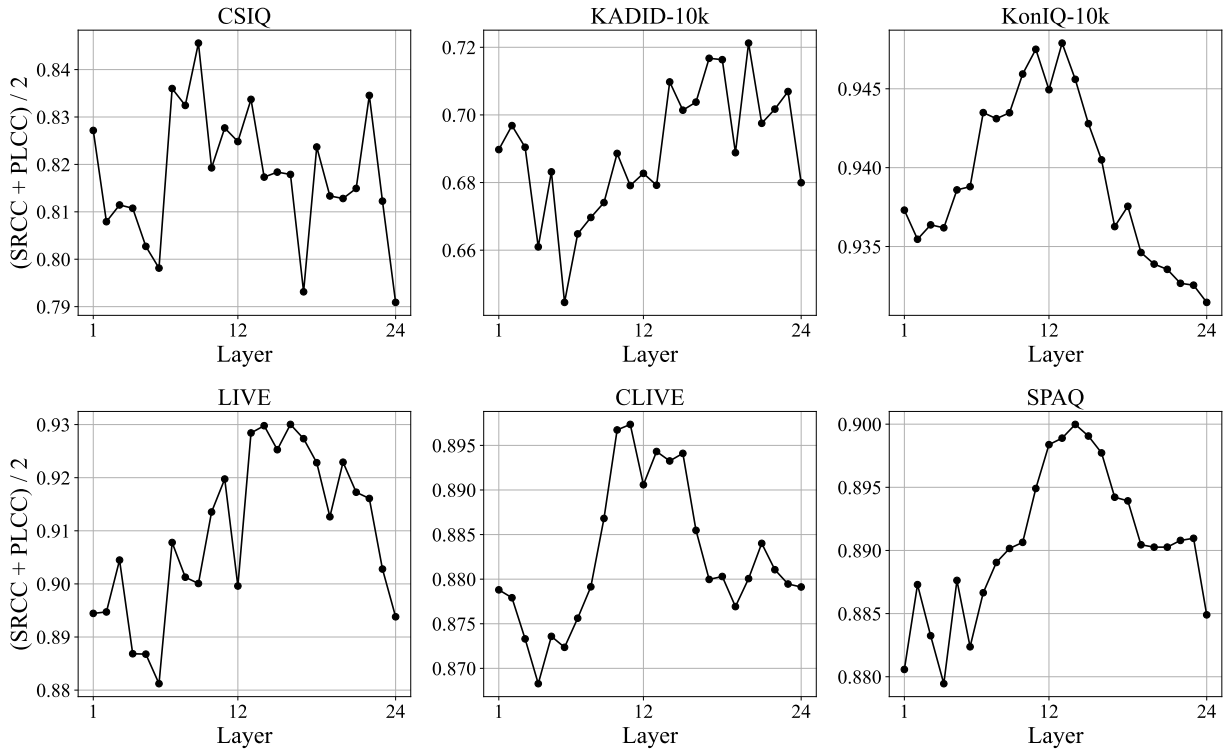


Figure 8. Performance comparison of TeMu-IQA using patch features from each single layer of CLIP-ViT.

Train	KonIQ-10k			SPAQ			CLIVE		
Test	SPAQ	CLIVE	CSIQ	KonIQ-10k	CLIVE	CSIQ	KonIQ-10k	SPAQ	CSIQ
TeMu-IQA-S	0.891/**0.891**	**0.864/0.905**	**0.869/0.898**	0.827/**0.858**	**0.857/0.880**	**0.901/0.901**	**0.819**/0.862	0.880/0.880	**0.915/0.926**
TeMu-IQA-S†	**0.892**/0.890	0.832/0.882	0.854/0.877	**0.829/0.858**	0.839/0.859	0.767/0.819	0.814/**0.866**	**0.887/0.885**	0.741/0.789
TeMu-IQA-I	0.895/**0.895**	**0.867/0.906**	**0.891/0.911**	**0.846/0.874**	**0.862/0.881**	**0.881/0.894**	0.809/**0.868**	0.888/0.890	**0.879/0.899**
TeMu-IQA-I†	**0.897/0.895**	0.847/0.896	0.849/0.877	0.834/0.866	0.834/0.862	0.812/0.842	**0.822/0.868**	**0.892/0.891**	0.835/0.853

Table 8. Results of cross-dataset evaluation. Models are trained on KonIQ-10k, SPAQ, and CLIVE. **Bold** indicates best results. Using specific layers optimized for a specific dataset (models marked with the "†" symbol) worsens the model's generalization ability. Metrics are SRCC and PLCC, respectively.

strate improved performance in Tab. 7, this effect is minimal. Moreover, as shown in Tab. 8, this optimization worsens the model's generalization ability. In particular, TeMu-IQA-I trained on SPAQ outperforms TeMu-IQA-I† by 8.5% in SRCC and 6.2% in PLCC on CSIQ. To retain the model's generalization ability, we recommend using all layers. Note that we use a single train–test split for all experiments in Tab. 8.

11. Local Distortion Guidance

We present more results of local distortion guidance in Fig. 10. As shown in Fig. 10, LDG helps the model alleviate the over-localized problem across all datasets. Here, we use CW-SSIM [27] for a fidelity score function. In contrast, Fig. 11 shows the prediction results obtained from a model trained on CLIVE [11] using LDG with $f(\cdot,\cdot) = 0, 0.3, 0.5, 0.7$. According to Eq. (10), we expect that as the value of the function decreases, the predicted quality scores of the distorted images also decline. As shown in Fig. 11, the model trained with LDG reflects the user's desired direction, thereby escaping from the over-localized problem.

12. Epochs & Generalization Ability

All experiments, except for FLIVE [40], are performed with 30 epochs as default. However, our model shows promising performance even with fewer training epochs because of a

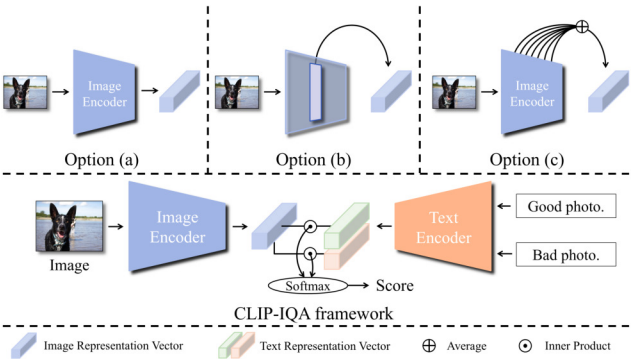
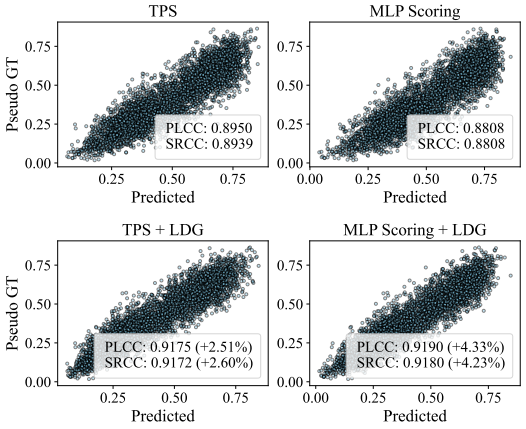


Figure 9. Illustration of options (a), (b), and (c) with CLIP-IQA framework.

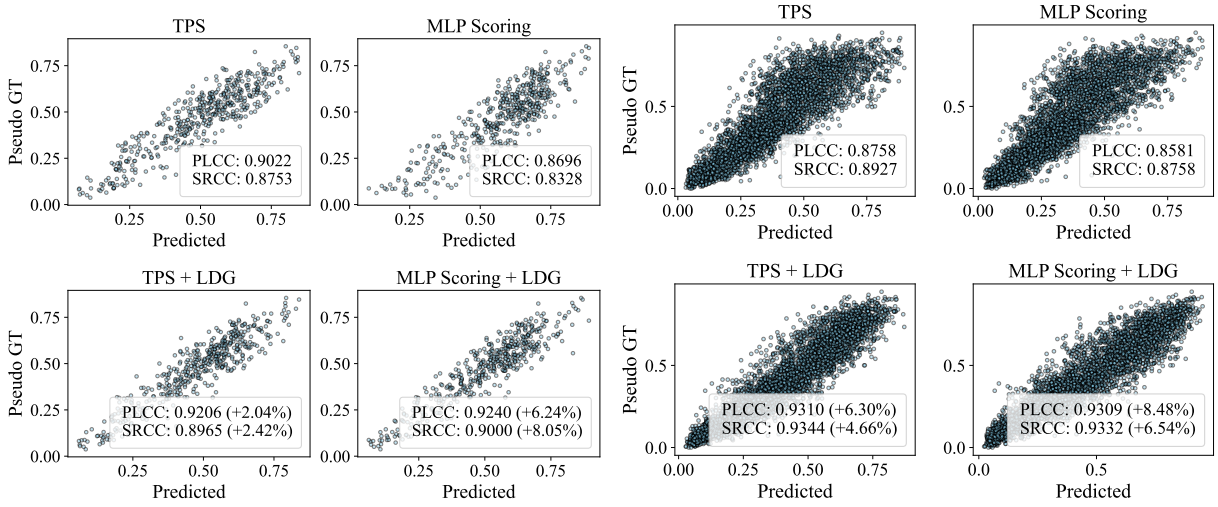
few trainable parameters. As shown in Tab. 9, TeMu-IQA-I achieves state-of-the-art performance and strong generalization ability with only 5 epochs. Additionally, as shown in Tab. 10, TeMu-IQA-I trained on SPAQ shows competitive generalization ability compared to MUSIQ [14], CLIP-IQA+ [33], LIQE [44], MANIQA [38], TOPIQ [4], and Q-ALIGN [36]. All results in Tabs. 9 and 10, except for those of TeMu-IQA, MANIQA, and TOPIQ, are derived from Q-ALIGN, and we follow the train–test split from Q-ALIGN to ensure a fair comparison.

13. Layer-wise Performance of CLIP-IQA

We present the layer-wise performance of the CLIP-IQA framework in Tabs. 11 to 13. In Fig. 9, option (a) refers to using the [**cls**] token from the last layer to create the image representation vector, as in the CLIP-IQA method. Option (b) indicates using the [**cls**] token from a specific layer. Option (c) denotes using the [**cls**] tokens from all layers. Specifically, for (c), we average the vectors created from each layer to produce an aggregated image representation vector. We evaluate the performance for three backbones: CLIP-ViT-L/14, CLIP-ViT-B/32, and CLIP-ViT-B/16.



(a) SPAQ



(b) CLIVE (c) KADID-10k

Figure 10. Scatter plots depicting the ground truth and predicted scores for generated SPAQ, CLIVE and KADID-10k.

							
	{ Predict / GT }	{ Predict / GT }	{ Predict / GT }	{ Predict / GT }	{ Predict / GT }	{ Predict / GT }	{ Predict / GT }
$f(\cdot,\cdot)=0$	0.000 / 0.000	0.000 / 0.000	0.000 / 0.000	0.030 / 0.000	0.000 / 0.000	0.000 / 0.000	0.010 / 0.000
$f(\cdot,\cdot)=0.3$	0.165 / 0.161	0.148 / 0.155	0.091 / 0.083	0.253 / 0.220	0.226 / 0.189	0.088 / 0.079	0.328 / 0.231
$f(\cdot,\cdot)=0.5$	0.319 / 0.269	0.289 / 0.259	0.156 / 0.139	0.423 / 0.368	0.444 / 0.316	0.161 / 0.132	0.501 / 0.385
$f(\cdot,\cdot)=0.7$	0.416 / 0.377	0.384 / 0.362	0.178 / 0.195	0.540 / 0.515	0.564 / 0.443	0.195 / 0.185	0.609 / 0.539

Figure 11. Results on using different fidelity score functions for LDG: $f(\cdot,\cdot) = 0, 0.3, 0.5, 0.7$.

Test	KonIQ-10k		SPAQ		CLIVE		AGIQA-3K		KADID-10k		CSIQ	
Method	SRCC	PLCC	SRCC	PLCC	SRCC	PLCC	SRCC	PLCC	SRCC	PLCC	SRCC	PLCC
MUSIQ	0.929	0.924	0.863	0.868	0.830	0.789	0.630	0.722	0.556	0.575	-	-
CLIP-IQA+	0.895	0.909	0.864	0.866	0.805	0.832	0.685	0.736	0.654	0.653	0.731	0.771
TOPIQ-ResNet50	0.927	0.942	0.869	0.872	0.803	0.816	0.660	0.729	0.547	0.568	0.717	0.731
LIQE	0.928	0.912	0.833	0.846	0.870	0.830	0.708	**0.772**	0.662	0.667	-	-
Q-ALIGN	0.940	0.941	0.887	0.886	0.860	0.853	**0.735**	**0.772**	0.684	0.674	0.700	0.759
5 epochs	0.942	0.954	0.898	0.899	0.889	0.907	0.712	0.765	**0.707**	**0.708**	0.802	0.830
10 epochs	0.942	0.953	0.900	0.900	0.889	0.910	0.713	0.769	0.695	0.696	0.800	0.832
15 epochs	0.942	0.953	**0.900**	**0.901**	0.884	0.904	0.710	0.768	0.693	0.698	0.800	0.832
20 epochs	0.945	0.956	0.900	0.900	0.885	0.906	0.715	**0.772**	0.689	0.694	**0.813**	**0.842**
25 epochs	0.945	0.956	0.899	0.900	**0.890**	**0.910**	0.714	0.771	0.688	0.693	0.812	**0.842**
30 epochs	**0.946**	**0.957**	**0.900**	**0.901**	0.889	0.909	0.715	**0.772**	0.689	0.694	0.812	**0.842**

Table 9. Results of cross-dataset evaluation. Models are trained on KonIQ-10k. Bottom shows the performance of TeMu-IQA-I for each epoch. **Bold** indicates best results.

Test	KonIQ-10k		SPAQ		CLIVE		AGIQA-3K		KADID-10k		CSIQ	
Method	SRCC	PLCC	SRCC	PLCC	SRCC	PLCC	SRCC	PLCC	SRCC	PLCC	SRCC	PLCC
MUSIQ	0.753	0.680	0.917	0.921	0.813	0.789	0.564	0.675	0.349	0.429	-	-
CLIP-IQA+	0.753	0.777	0.881	0.883	0.719	0.755	0.577	0.614	0.633	0.638	-	-
LIQE	0.826	0.847	0.922	0.919	0.805	0.866	0.672	0.722	0.639	0.627	-	-
MANIQA	0.784	0.831	0.924	0.926	0.826	0.859	0.625	0.690	0.541	0.569	0.714	0.742
TOPIQ-ResNet50	0.806	0.816	0.921	0.925	0.805	0.834	0.546	0.638	0.441	0.492	0.660	0.709
Q-ALIGN	**0.848**	**0.879**	**0.930**	0.933	0.865	0.873	**0.723**	**0.786**	**0.743**	**0.740**	0.733	0.781
TeMu-IQA-S	0.815	0.855	**0.930**	**0.934**	0.867	0.873	0.667	0.748	0.679	0.688	**0.813**	**0.849**
TeMu-IQA-I	**0.848**	0.877	0.929	0.933	**0.868**	**0.879**	0.675	0.746	0.656	0.668	0.806	0.839

Table 10. Results of cross-dataset evaluation. Models are trained on SPAQ. **Bold** and underline indicate best and second-best results, respectively.

Method	KonIQ-10k		SPAQ		CLIVE	
	SRCC	PLCC	SRCC	PLCC	SRCC	PLCC
layer1	0.2790	0.3125	-0.0082	0.0870	0.2020	0.2914
layer2	0.4249	0.4390	0.3798	0.3917	0.3517	0.4023
layer3	0.4781	0.4592	0.5438	0.5428	0.4374	0.4630
layer4	0.4522	0.4501	0.6155	0.6045	0.3258	0.3601
layer5	0.5121	0.5103	0.6806	0.6564	0.3639	0.4301
layer6	0.6307	0.6190	0.6686	0.6266	0.4314	0.4762
layer7	0.6559	0.6129	0.6974	0.6170	0.4513	0.4910
layer8	0.6477	0.6115	0.7083	0.5907	0.4042	0.4634
layer9	0.6707	0.6566	0.7050	0.6619	0.3592	0.4109
layer10	0.7062	0.7023	0.7112	0.6775	0.3580	0.4034
layer11	0.7224	0.7043	0.7221	0.7298	0.4600	0.4838
layer12	0.7230	0.7146	0.7583	0.7652	0.5772	0.5670
layer13	0.7242	0.7018	0.7696	0.7754	0.6214	0.6222
layer14	0.7411	0.7098	**0.7910**	**0.7958**	0.6793	0.6584
layer15	0.7523	0.7128	0.7805	0.7671	0.7094	0.6896
layer16	0.7661	0.6968	0.7977	0.7345	**0.7289**	**0.7233**
layer17	0.7598	0.7121	0.7842	0.7401	0.6838	0.6810
layer18	0.7526	0.7202	0.7527	0.7234	0.7294	0.7075
layer19	0.7383	0.7216	0.6686	0.6600	0.6804	0.6753
layer20	0.7280	0.6936	0.6414	0.6241	0.6261	0.6115
layer21	0.6939	0.6453	0.6296	0.5939	0.6011	0.5454
layer22	0.6101	0.5295	0.5038	0.4496	0.4649	0.3997
layer23	0.6238	0.5716	0.4276	0.4104	0.4647	0.4332
layer24 (CLIP-IQA)	0.6112	0.5681	0.3800	0.3751	0.4426	0.4185
Entire layer (Option (c))	**0.7743**	**0.7750**	0.7751	0.7732	0.6949	0.6803

Table 11. Results of the CLIP-IQA framework on KonIQ-10k, SPAQ, and CLIVE with the [**cls**] token from different layer. Here, we use CLIP-ViT-L/14 as the backbone model. **Bold** indicates best results.

Method	KonIQ-10k		SPAQ		CLIVE	
	SRCC	PLCC	SRCC	PLCC	SRCC	PLCC
layer1	0.5156	0.5267	0.3860	0.3825	0.5134	0.5164
layer2	0.5846	0.5701	0.5659	0.5531	0.4325	0.4302
layer3	0.5427	0.5346	0.6412	0.6278	0.4791	0.4904
layer4	0.5755	0.5803	0.7448	0.7411	0.5253	0.5180
layer5	0.6348	0.6379	0.7454	0.7376	0.5259	0.5258
layer6	0.6826	0.6843	0.7397	0.7202	0.5121	0.5336
layer7	0.6708	0.6655	0.7434	0.7392	0.3704	0.3876
layer8	**0.7373**	0.7176	**0.8260**	**0.8267**	0.5342	0.5579
layer9	0.7113	0.7020	0.8231	0.8187	0.5757	0.5877
layer10	0.7044	0.6816	0.6243	0.6175	0.4748	0.4483
layer11	0.5969	0.5901	0.5917	0.5759	0.4806	0.4529
layer12 (CLIP-IQA)	0.6336	0.6422	0.6038	0.5952	0.4621	0.4532
Entire layer (Option (c))	0.7319	**0.7497**	0.7803	0.7773	**0.6042**	**0.6105**

Table 12. Results of the CLIP-IQA framework on KonIQ-10k, SPAQ, and CLIVE with the [**cls**] token from different layer. Here, we use CLIP-ViT-B/32 as the backbone model. **Bold** indicates best results.

Method	KonIQ-10k		SPAQ		CLIVE	
	SRCC	PLCC	SRCC	PLCC	SRCC	PLCC
layer1	0.2913	0.2664	0.5414	0.5343	0.4198	0.3919
layer2	0.3979	0.3681	0.5714	0.5541	0.4593	0.4205
layer3	0.4522	0.3937	0.4968	0.4780	0.4559	0.4011
layer4	0.5008	0.4441	0.5099	0.4989	0.4511	0.4094
layer5	0.4839	0.4256	0.5630	0.5307	0.4508	0.3988
layer6	0.5633	0.4514	0.6639	0.5498	0.5156	0.4455
layer7	0.6296	0.5149	0.7100	0.6297	0.5138	0.4044
layer8	**0.6614**	**0.5462**	0.7139	**0.6322**	0.6134	0.4682
layer9	0.6546	0.5074	**0.7188**	0.5810	**0.6523**	**0.5304**
layer10	0.5401	0.4681	0.5207	0.4519	0.4736	0.4205
layer11	0.5148	0.4289	0.5694	0.4612	0.4734	0.4083
layer12 (CLIP-IQA)	0.4658	0.4631	0.5330	0.4732	0.3818	0.3421
Entire layer (Option (c))	0.6371	0.5438	0.6897	0.6238	0.5753	0.5100

Table 13. Results of the CLIP-IQA framework on KonIQ-10k, SPAQ, and CLIVE with the [**cls**] token from different layer. Here, we use CLIP-ViT-B/16 as the backbone model. **Bold** indicates best results.