Multimedia Tools and Applications (2022) 81:42607–42619
https://doi.org/10.1007/s11042-022-13477-9

1221: DEEP LEARNING FOR IMAGE/VIDEO COMPRESSION AND VISUAL QUALITY ASSESSMENT

Multiscale convolutional neural network for no-reference image quality assessment with saliency detection

Xiaodong Fan[1] · Yang Wang[1] · Changzhong Wang[1] · Xiangyue Chen[1]

Received: 23 May 2021 / Revised: 6 September 2021 / Accepted: 13 July 2022 /
Published online: 25 July 2022

Abstract
In recent years, Convolutional Neural Network (CNN) has been gradually applied to Image Quality Assessment (IQA). Most CNNs segment the image into patches for training, which lead to increase of data and affect calculation speed of the model. Meanwhile, the parameters of CNN usually reach millions, which is the root cause of overfitting. In this paper, a multiscale CNN for NR-IQA is established to solve these problems. Since IQA simulates the perception of Human Visual System (HVS) on image quality, salient areas are more valuable for reference. Therefore a patch sampling method was designed based on saliency detection. Firstly, patches with salient values between given thresholds are retained as training data. Secondly, the sampled patches are fed into multiscale CNN. The network consists of three branches with multiscale convolutional kernels. Finally, the weighted average of the quality scores from the salient patches is the final score. The CNN was trained on LIVE dataset and cross-validated on CSIQ dataset. The experimental results show that the proposed method can achieve better performance with fewer parameters compared with state-of-the-art NR-IQA algorithms.

Keywords Convolutional neural network · No-reference image quality assessment · Human visual system · Saliency detection · Multiscale network

1 Introduction

Image Quality Assessment (IQA) is an emerging artificial intelligence technology to simulate human perception of image quality. Full-Reference IQA (FR-IQA) takes the known original image as a reference to evaluate the quality of the distorted image. However, ideal reference images are actually unavailable. Reduced-Reference IQA (RR-IQA) methods are proposed

✉ Xiaodong Fan
bhdxfxd@163.com

[1] College of Mathematical Sciences, Bohai University, Jinzhou 121000 Liaoning, China

when only partial information of the reference image is known. In practical applications, the features of reference images are often difficult to obtain. No-Reference IQA (NR-IQA) mainly predicts image quality without any information from reference images. Therefore, it is one of the most widely used and challenging tasks.

The deep neural networks have demonstrated a strong ability to capture the basic attributes of images, which provides a new solution to NR-IQA. Convolutional Neural Network (CNN) can extract image features more quickly and accurately. For example, Kang et al. [11] were pioneers in applying CNN to IQA. Inspired by CORNIA method [22], they proposed a meaningful framework and achieved excellent results. However, Kang's network contains only one convolutional layer, which makes the expression of features incomplete. Sun et al. [20] and Bosse et al. [3] added convolutional layers on the basis of Kang. In [20], they proposed a branching framework based on global and local perception. Local and global features were combined to estimate the overall image quality. Reference [3] estimated the quality without employing any domain knowledge. Pan et al. [17] proposed an improved CNN combined saliency detection. This algorithm was based on the free energy neural model [7] to detect saliency map, then applied the saliency map as a weighting mask to output the quality score of the whole image. Squeeze-and-Excitation Network [8] was designed to enhance features and improve accuracy. When the existing model cannot meet our specific needs, we may not be allowed to customize a new architecture at the cost of heavy human effort or numerous GPU hours [27]. Zhou et al. [26] divided the training images into high-confidence distorted images and low-confidence distorted images, and reasonably assigned different local quality scores to each patch through specific Gaussian functions with the global quality score as the mean value and the undetermined hyperparameter as the standard deviation. By mimicking the active inference process of IGM, Ma et al. [15] established an active inference module based on the generative adversarial network (GAN) to predict the primary content, the image quality is predicted according to the correlation between the distorted image and its primary content. However, the size of the convolutional kernels in the above networks are single, the local features under different scales cannot be extracted well, the computation complexity is relatively high. Therefore, a multiscale CNN is proposed to form a more rapid and effective IQA model by using three different convolutional kernels.

In the CNN for NR-IQA, most methods are to process small patches, and then use the average score of patches to predict the whole image quality [18]. Therefore, how to select the appropriate patches is a topic that we should focus on. Since IQA simulates the perception of Human Visual System (HVS) on image quality, salient areas are more valuable for reference. Therefore, the saliency detection can be combined to select patches. Saliency detection can help humans quickly and accurately select the most important areas from complex images. There were many classic algorithms, such as the earliest visual attention algorithm Itti [9], LC algorithm [23], HC algorithm [6], AC algorithm [1] and FT algorithm [2]. Itti applied the multiple characteristics, multiscale decomposition and filtering to get saliency map; LC algorithm used the difference in pixel values as the saliency value to generate saliency map; HC algorithm took color information into account, instead of gray information as LC algorithm did; AC algorithm obtained the final saliency map by adding the saliency of multiscale fuzzy images; FT algorithm proposed five indicators to detect the saliency of the image. The SDSP algorithm [24] integrated the following three priors. First, HVS in detecting salient objects can be well simulated through band-pass filtering. Second, people tend to focus their attention on the center of the image. Third, warm colors attract people's attention more easily than cool colors.

In this paper, in order to relieve the overfitting problem and be more consistent with HVS, we propose a multiscale CNN for NR-IQA with saliency detection. Firstly, the SDSP algorithm will be applied to generate saliency map to select appropriate patches, patches with salient values between given thresholds are retained as training data. Secondly, the sampled patches are fed into multiscale CNN to extract features. Different features will be extracted by different convolutional kernels, so the designed network consists of three branches with multiscale convolutional kernels. Finally, the weighted average of the quality scores from the salient patches is the final score. The rest of paper is organized as follows. Section 2 describes the designed NR-IQA algorithm in detail. Section 3 provides a comparison experiment and evaluate the performance of our method. The conclusions are contained in Section 4.

2 Method for NR-IQA

In this section, the implementation of the algorithm will be elaborate. It is divided into two parts. Firstly, a saliency detection algorithm is adopted to select non-overlapping patches. Secondly, the normalized patches are fed into the multiscale CNN to extract features and train the network.

2.1 Patch sampling based on saliency detection

The CNN for NR-IQA takes the patches segmented from the images as training data. However, patches have different reference values for IQA. In order to make the extracted patches representative, a patch sampling strategy will be proposed based on saliency detection.

In this paper, we adopted a novel salient region detection method, namely Saliency Detection by combining Simple Priors (SDSP, for short). It was used to generate saliency map, the computation process is shown in Fig. 1.

Given an image $\{f(x)|x \in \Omega\}$, where $\Omega \subset R^2$ denotes the image spatial domain, $x = (i, j)$ is the position coordinate. $f(x)$ is actually a vector, containing three values representing R, G, and B intensities at the position x.

The image $f(x)$ in the RGB color space can be converted to CIEL*a*b* color space. In the transformed image, $f_L(x)$, $f_a(x)$ and $f_b(x)$ denoted L*-channel, a*-channel, and b*-channel, respectively. L*-channel represents the brightness of the pixel, a*-channel represents green-red information, b*-channel represents blue-yellow information. The frequency prior maps $S_F(x)$ is defined as

$$S_F(x) = \left((f_L(x)*g(x))^2 + (f_a(x)*g(x))^2 + (f_b(x)*g(x))^2 \right)^{\frac{1}{2}} \tag{1}$$

where * denotes the convolution operation, $g(x)$ is the transfer function of a log-Gabor filter.

The location prior maps is expressed as a Gaussian map:

$$S_D(x) = exp\left(-\frac{\|x-c\|_2^2}{\sigma_D^2} \right) \tag{2}$$

where the center of the image $f(x)$ is denoted by c, and the location variance by σ_D.

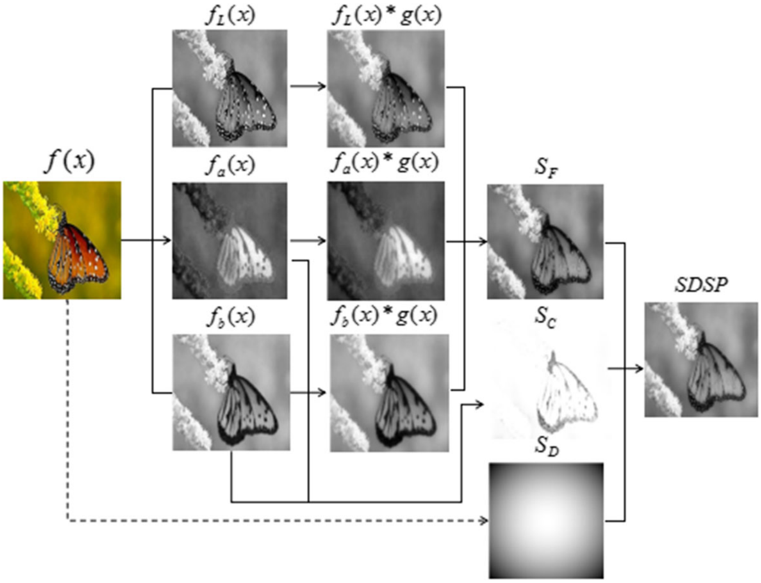

Fig. 1 Illustration for the computation process of SDSP

The color prior maps is defined as

$$S_C(x) = 1-exp\left(\frac{-f_{\boldsymbol{an}}{}^2(x)+f_{bn}^2(x)}{\sigma_C^2}\right) \tag{3}$$

where σ_C is a color variance, $f_{\boldsymbol{an}}(x)$ and $f_{bn}(x)$ can be computed as:

$$f_{\boldsymbol{an}}(x) = \frac{f_a(x)-minf_a}{maxf_a-minf_a}, f_{bn}(x) = \frac{f_b(x)-minf_b}{maxf_b-minf_b}, \tag{4}$$

$minf_a(maxf_a)$ is the minimum (maximum) value of $f_a(x)$,and $minf_b(maxf_b)$ is the minimum (maximum) value of $f_b(x)$.

The image's final saliency map can be naturally defined as:

$$S(\mathrm{x}) = S_F(\mathrm{x})\cdot S_D(\mathrm{x})\cdot S_C(\mathrm{x}) \tag{5}$$

where $S_F(\mathrm{x})$, $S_D(\mathrm{x})$ and $S_C(\mathrm{x})$ are the maps corresponding to frequency prior, location prior and color prior. They can be computed as Eqs. (1), (2) and(3), respectively.

Figure 2 shows an example of patch sampling. Firstly, for each distorted image, saliency map is generated by SDSP algorithm. Secondly, the average saliency value from the saliency map is calculated as

$$Sa = \frac{\sum_{i=1}^{H}\sum_{j=1}^{W} S(i,j)}{H \times W} \tag{6}$$

where $S(i, j)$ is the value of the saliency map at the position (i, j), H and W are the height and width of the saliency map respectively.

Thirdly, If $S(i, j) < Sa$, let $S(i, j) = 0$, otherwise, $S(i, j) = 1$. In this way, a binary image B of saliency map is generated.

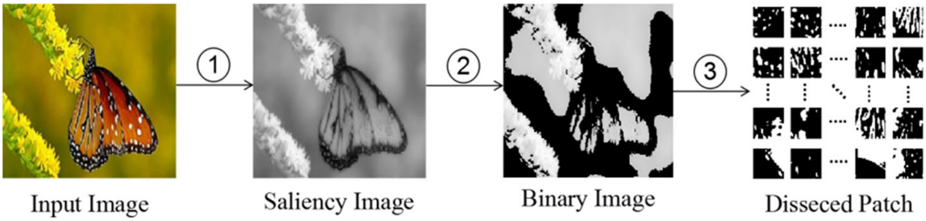

Fig. 2 Example of patch sampling

Fourthly, the binary images are segmented into 32 × 32 patches. For each patch B_p, it's average saliency value is defined as

$$Sa_p = \frac{\sum\limits_{i=1}^{32}\sum\limits_{j=1}^{32} B_p(i,j)}{32 \times 32} \tag{7}$$

where $B_p(i, j)$ is the value at the position (i, j) in binary patch B_p.

If the average saliency value Sa_p of the patch p is between the given thresholds Ta_1 and Ta_2, i.e., $Ta_1 < Sa_p < Ta_2$, the patch is considered to be salient and retained. Otherwise, abandoned.

2.2 Multiscale CNN

In this section, we will elaborate the data preprocessing process, CNN architecture and parameter learning in detail.

2.2.1 Local normalization

In practice, the value range of data is not uniform, which makes the learning process time-consuming. Therefore, each patch is normalized before training. In order to improve the normalization efficiency, the patches are locally normalized to the standard normal distribution. For the value $f(i, j)$ of a pixel at the position (i, j), its normalized value $\widetilde{f}(i,j)$ is calculated as follows:

$$\widetilde{f}(i,j) = \frac{f(i,j)-\mu(i,j)}{\delta(i,j)+C} \tag{8}$$

$$\mu(i,j) = \frac{\sum\limits_{m=-M}^{M}\sum\limits_{n=-N}^{N} f(i+m,j+n)}{(2M+1)(2N+1)} \tag{9}$$

$$\delta(i,j) = \sqrt{\sum\limits_{m=-M}^{M}\sum\limits_{n=-N}^{N} (f(i+m,j+n)-\mu(i,j))^2} \tag{10}$$

where a small constant C is set to maintain numerical stability, M and N represent the normalization window sizes,$\mu(i, j)$ and $\delta(i, j)$ are the mean value and standard deviation of the pixel value $f(i, j)$ respectively.

2.2.2 Network structure

The network consists of three parts: patch sampling, CNN model and quality evaluation as shown in Fig. 3.The designed CNN includes three branches with multiscale convolutional kernels. The sizes of convolutional kernels are chose as 3 × 3, 5 × 5, and 7 × 7.Eachbranchinclude five convolutional and five pooling layers. Three branches are fused after the last pooling. In order to merge the three scales, zero fillings are executed for the convolution, the stride size is 1 pixel. All pooling layers adopt 2 × 2 max pooling. Dropout is added in the first full connected layer with ratio of 0.5 to improve the generalization ability of the model and reduce the overfitting effect. We use Rectified Linear Unit (ReLU) in the two full connected layers as the activation function.

Table 1 shows the detailed parameters of the CNN, the depth of convolutional kernels and the number of feature maps. First, the patches with size 32 × 32 × 1 pass through the first convolutional layer (C1) with kernels of size 3 × 3 × 32, 5 × 5 × 32, and 7 × 7 × 32,three groups of feature maps are produced with the size 32 × 32 × 32. The 2 × 2 maximum pooling follows to reduce the feature maps to 16 × 16 × 32. Through the second same convolutional (C2) and pooling layer, three groups of feature maps with the size 8 × 8 × 32 are produced. After the five similar convolutions and pooling, three groups of features with the size of 1 × 1 × 32 are obtained in the end. Finally three feature maps are fused as a 1 × 1 × 96 feature, the quality score is predicted through two full connection layers with 128 nodes and a simple linear regression layer with 1 node.

2.2.3 Parameter learning

For training the network, 32 × 32 non-overlapping patches are assigned a quality score as its source image's ground truth score. The loss function is defined as:

$$\boldsymbol{Loss} = \frac{\boldsymbol{1}}{N}\sum_{i=\boldsymbol{1}}^{N}\|y_i - F(p_i\boldsymbol{;}\,\omega)\|_{l_1} \tag{11}$$

where y_i denotes the ground true score, $F(p_i;\ \omega)$ denotes the estimated score of the input patch p_i, and N is the total number of image patches.ω is the network parameter to be learned. Parameter learning is achieved through minimizing loss function. The parameters are updated by the following optimization problem:

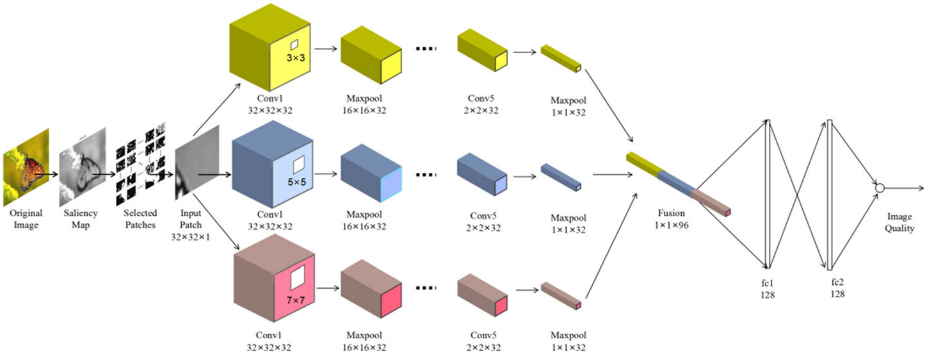

Fig. 3 Architecture of multiscale CNN

Table 1 Parameters of proposed multiscale CNN

Layer	Input	C1	P1	C2	P2	C3	P3	C4	P4	C5	P5	Fusion	FC	FC	Out put
Kernel size	32×32×1	3×3×32 5×5×32 7×7×32	2×2 2×2 2×2	3×3×32 5×5×32 7×7×32	2×2 2×2 2×2	3×3×32 5×5×32 7×7×32	2×2 2×2 2×2	3×3×32 5×5×32 7×7×32	2×2 2×2 2×2	3×3×32 5×5×32 7×7×32	2×2 2×2 2×2	1×1×96	128	128	1
Patch size	32×32×1	32×32×32	16×16×32	16×16×32	8×8×32	8×8×32	4×4×32	4×4×32	2×2×32	2×2×32	1×1×32				

$$\omega' = \min_{\omega} \boldsymbol{Loss} \tag{12}$$

where ω' is the updated parameter. The loss function is optimized by the following Adam method.

1) Calculate the gradient of the loss function with respect to the parameter ω_t at time t:

$$\frac{d_t = \nabla \boldsymbol{Loss} = \partial \boldsymbol{Loss}}{\partial(\omega_t)} \tag{13}$$

2) Calculate the first-order momentum at time t:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) d_t \tag{14}$$

Correct the first order momentum:

$$\widehat{m}_t = \frac{m_t}{1-\beta_1^t} \tag{15}$$

3) Calculate the second-order momentum at time t:

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) d_t^2 \tag{16}$$

Correct the second order momentum:

$$\widehat{v}_t = \frac{v_t}{1-\beta_2^t}\eta_t = lr\cdot\frac{\widehat{m}_t}{\sqrt{\widehat{v}_t}} = lr\cdot\frac{\frac{m_t}{1-\beta_1^t}}{\sqrt{\frac{v_t}{1-\beta_2^t}}} \tag{17}$$

4) Calculate the descent gradient at time t:

$$\eta_t = lr\cdot\frac{\widehat{m}_t}{\sqrt{\widehat{v}_t}} = lr\cdot\frac{\frac{m_t}{1-\beta_1^t}}{\sqrt{\frac{v_t}{1-\beta_2^t}}} \tag{18}$$

5) Update the parameters ω_{t+1} at time $t + 1$:

$$\omega_{t+1} = \omega_t - \eta_t = \omega_t - lr\cdot\frac{\frac{m_t}{1-\beta_1^t}}{\sqrt{\frac{v_t}{1-\beta_2^t}} + \varepsilon} \tag{19}$$

where β_1 and β_2 are constants between 0 and 1, the parameter ε is a very small number to prevent dividing by zero in the implementation, the learning rate is denoted by lr.

Table 2 SROCC of different methods on LIVE

Method	JP2K	JPEG	WN	BLUR	FF	ALL
Kang [11]	0.952	0.977	0.978	0.962	0.908	0.956
CORNIA [22]	0.943	0.955	0.976	0.969	0.906	0.942
Pan [17]	0.955	0.981	0.953	0.927	**0.983**	0.968
SFOSR [21]	0.932	0.947	0.982	0.951	0.946	0.953
Li's method [13]	0.964	0.935	0.988	0.941	0.945	0.958
SOM [25]	0.947	0.952	0.984	0.976	0.937	0.964
Deep CNN [14]	0.973	0.955	0.981	**0.984**	0.955	0.956
Ours	**0.976**	**0.982**	**0.989**	0.932	0.900	**0.973**

3 Experimental results

In this section, we first describe the experimental setups, including the datasets, evaluation indicators and the experimental parameters. Then the performance of multiscale CNN is compared with other IQA methods on LIVE dataset. To investigate generalization ability of our multiscale CNN, the trained CNN on LIVE dataset are validated on the CSIQ dataset.

3.1 Experimental setups

The experiment were implemented on two universal IQA datasets, LIVE [19] and CSIQ [5]. The LIVE dataset consists of 29 source reference images and 982 distorted images with five distortions: JPEG2000 compression (JP2K), JPEG compression (JPEG), White Gaussian (WN), Gaussian blur (BLUR) and fast-fading (FF). Differential Mean Opinion Scores (DMOS) in the range of [0,100] represent the subjective quality of the image. Higher DMOS corresponds to lower image quality. The CSIQ dataset consists of 30 source reference images and 866 distorted images with six distortions: JP2K, JPEG, WN, BLUR, FN and CONTRAST. DMOS in the range of [0,1] is associated with each image.

The two measures of Spearman Rank Order Correlation Coeffificient (SROCC) and Linear Correlation Coeffificient (LCC) were employed to evaluate the performance of IQA algorithms. SROCC assessed the relationship between the estimated scores and the ground true scores. LCC measured the degree of linear dependence between the two scores.

The super parameters involved in the experiment are set as follows. In the patch sampling, $\sigma_C = 0.25$, $\sigma_D = 114$. The fixed threshold$Ta_1 = 1.1$, $Ta_2 = 1.8$. In local normalization, $C = 1$, $M = N = 3$. In parameter learning, the relevant optimization parameters were selected as $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 10^{-8}$, $lr = 0.001$. The experiment results obtained from 95 train-test

Table 3 LCC of different methods on LIVE

Method	JP2K	JPEG	WN	BLUR	FF	ALL
Kang [11]	0.953	0.981	0.984	0.953	0.933	0.953
CORNIA [22]	0.951	0.965	0.987	0.968	0.917	0.935
Pan [17]	0.961	0.989	0.954	0.948	**0.987**	0.969
SFOSR [21]	0.939	0.951	0.949	0.945	0.925	0.964
Li's method [13]	0.978	0.977	0.993	0.945	0.960	0.966
SOM [25]	0.952	0.961	0.991	**0.974**	0.954	0.962
Deep CNN [14]	0.945	0.941	0.964	0.969	0.907	0.935
Ours	**0.984**	**0.992**	**0.996**	0.910	0.847	**0.973**

Table 4 SROCC and LCC on the the shared distortions

Methods	LCC	SROCC
Kang [11]	0.913	0.923
CORNIA [22]	0.914	0.899
Sen's method [10]	0.930	0.934
SFOSR [21]	0.729	0.740
Jie's method [12]	0.916	0.874
Ours	**0.965**	**0.947**

iterations. In each iteration, the batch size was 64, 60% of data was randomly selected for training, 20% for validation, and the remaining 20% for test. The experimental results showed that this setting could almost save half of the training time and achieve better performance.

3.2 Performance on LIVE dataset

The network was trained with five distortions, the evaluation indicators SROCC and LCC are compared with the seven advanced NR-IQA algorithms as shown in Tables 2 and 3. The comprehensive experimental results for all distortions were listed in the last column of the Table. The best results among the algorithms were in bold. Our method worked well on each of the five distortions, especially on JP2K, JPEG, and WN. Although the results on the distortion types BLUR and FF were slightly lower than other algorithms, our algorithm outperformed others in terms of the comprehensive distortion.

3.3 Cross-dataset validation

To evaluate generalization ability of our multiscale CNN, we trained the model on the LIVE dataset and tested on the CSIQ dataset, which contained only the four distortions (JPEG, JP2K, BLUR, WN) shared with LIVE. Table 4 shows the SROCC and LCC on the four distortions. The result shows that our method has excellent robustness and outperforms other algorithms.

The CSIQ dataset contains two types of distortion (FN, CONTRAST) that are not shared with the LIVE. Therefore, the two distortions are often overlooked in the references. In this paper, our multiscale CNN was tested on all distortions from CSIQ. Since the network is not trained on distortion types FN and CONTRAST, it can be found from Table 5 that all algorithms performed worse on the full distortions than the shared distortions. However, our algorithm overmatchs other algorithms for both shared and full distortions.

Table 5 SROCC on all distortions

Methods	CSIQ	
	shared	all
CORNIA [22]	0.899	0.663
DIQaM-NR [4]	0.908	0.681
WaDIQaM-FR [4]	0.866	0.704
BRISQUE [16]	0.899	0.557
Ours	**0.947**	**0.871**

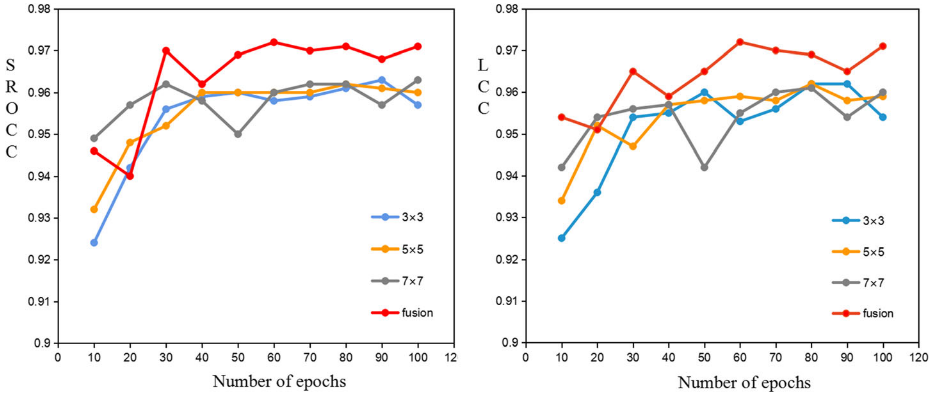

Fig. 4 SROCC and LCC under different sizes of convolutional kernels

3.4 Effect of the size of the convolutional kernels

Different sizes of convolutional kernels can extract different features from images. Comparative experiments are designed to analyzed the effect of the size of the convolutional kernels. The experimental results were shown in Fig. 4. In the network, the single size convolutional kernels are compared with their multiscale fusion under the same conditions. The blue line represented the network with a convolutional kernel size of 3 × 3, the orange line represented 5 × 5, the gray line represented 7 × 7, the red line represented their fusion. As can be seen, the multiscale CNN shows the best performance in SROCC and LCC indicators.

3.5 Effect of the patch sampling

The CNN was trained and tested on LIVE dataset with saliency sampling and without sampling, respectively. The experiment results in Table 6 show that the network with saliency sampling results in better performance.

Some conclusions follow from synthetically analyzing the experimental results. First, multiscale CNN can effectively improve the accuracy of the model. Second, the saliency sampling can increase the efficiency of CNN. Therefore, the proposed multiscale CNN based on saliency detection can effectively evaluate the quality of distorted images and improves the performance of NR-IQA.

Table 6 Performance comparison under salient sampling

Patch Sampling	LCC	SROCC
Salient sampling	**0.973**	**0.973**
Without Sampling	0.963	0.961

4 Conclusion

In this paper, a multiscale CNN for NR-IQA with Saliency detection was proposed. Human vision always focuses on the salient area of the image. In the proposed model, the saliency detection techniques were used to filter the training data, and the filtered data were fed into CNN for training, which was consist with the characteristics of Human Vision System (HVS) and improved the training efficiency. In addition, the field of human vision usually is usually fluid, so the three-scale convolution kernels were designed to extract richer features. Experiments show that the proposed network not only has fewer parameters, but also can achieve higher performance.

However, the patch size and convolution kernel size in our model are empirically determined. It is unfair to use the same size for different types and different sizes of input images. In the future work, it will be a promising topic to design an adaptive mechanism to identify the patch size and kernel size according to the type and size of image.

Acknowledgements This work was supported by the National Natural, Science Foundation of China under Grant 61976027, Liaoning Provincial Department of Education under Grant LJ2019011, LJKZ1026, Liaoning Natural Foundation Guidance Plan under Grant 2019-ZD-0502, and Liaoning Revitalization Talents Program under Grant XLYC2008002.

References

1. Achanta R, Estrada F, Wils P (2008) Salient region detection and segmentation. In: IEEE international conference on computer vision systems (ICVS), pp 66-75. https://doi.org/10.1007/978-3-540-79547-6_7
2. Achanta R, Hemami S, Estrada F, Susstrunk S (2009) Frequency-tuned salient region detection. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 1597-1604. https://doi.org/10.1109/CVPR.2009.5206569
3. Bosse S, Maniry D, Wiegand T, Samek W (2016) A deep neural network for image quality assessment. In: IEEE international conference on image processing (ICIP), pp 3773-3777. https://doi.org/10.1109/ICIP.2016.7533065
4. Bosse S, Maniry D, Müller K, Wiegand T, Samek W (2018) Deep neural networks for no-reference and full-reference image quality assessment. IEEE Trans Image Process 27(1):206–219. https://doi.org/10.1109/TIP.2017.2760518
5. Chandler EC, Larson DM (2010) Most apparent distortion: full-reference image quality assessment and the role of strategy. Journal of Electronic Imaging 19(1):011006. https://doi.org/10.1117/1.3267105
6. Cheng M, Zhang G, Mitra NJ, Huang X, Hu S (2011) Global contrast based salient region detection. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 409-416. https://doi.org/10.1109/CVPR.2011.5995344
7. Gu K, Zhai G, Lin W, Yang X, Zhang W (2015) Visual saliency detection with free energy theory. IEEE Signal Process Lett 22(10):1552–1555. https://doi.org/10.1109/LSP.2015.2413944
8. Hu J, Shen L, Albanie S, Sun G, Wu E (2020) Squeeze-and-excitation networks. IEEE Trans Pattern Anal Mach Intell 42(8):2011–2023. https://doi.org/10.1109/TPAMI.2019.2913372
9. Itti L, Koch C, Niebur E (1998) A model of saliency-based visual attention for rapid scene analysis. IEEE Trans Pattern Anal Mach Intell 20(11):1254–1259. https://doi.org/10.1109/34.730558
10. Jia S, Zhang Y (2018) Saliency-based deep convolutional neural network for no-reference image quality assessment. Multimed Tools Appl 77(12):14859–14872. https://doi.org/10.1007/s11042-017-5070-6
11. Kang L, Ye P, Li Y, Doermann D (2014) Convolutional neural networks for no-reference image quality assessment. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 1733-1740. https://doi.org/10.1109/CVPR.2014.224
12. Li L, Zhou Y (2017) Visual saliency based blind image quality assessment via convolutional neural network. In: IEEE international conference on neural information processing (ICONIP), pp 550-557. https://doi.org/10.1007/978-3-319-70136-3_58

13. Li J, Zou L, Yan J, Deng D, Qu T, Xie G (2015) No-reference image quality assessment using prewitt magnitude based on convolutional neural networks. Signal Image Vid Process 10:609–616. https://doi.org/10.1007/s11760-015-0784-2
14. Li Y, Po L, Feng L, Yuan F (2016) No-reference image quality assessment with deep convolutional neural networks. In: IEEE international conference on digital signal processing (DSP), pp 685-689. https://doi.org/10.1109/ICDSP.2016.7868646
15. Ma J, Wu J, Li L, Dong W, Lin W (2021) Blind image quality assessment with active inference. IEEE Transactions on Image Processing, pp 99:1–1. https://doi.org/10.1109/TIP.2021.3064195
16. Mittal A, Moorthy AK, Bovik AC (2012) No-reference image quality assessment in the spatial domain. IEEE Trans Image Process 21(12):4695–4678. https://doi.org/10.1109/TIP.2012.2214050
17. Pan C, Xu Y, Yan Y, Gu K, Yang X (2016) Exploiting neural models for no-reference image quality assessment. In: IEEE visual communications and image processing (VCIP), pp1–4. https://doi.org/10.1109/VCIP.2016.7805524
18. Po LM, Liu M, YuenWilson YF et al (2019) A novel patch variance biased convolutional neural network for no-reference image quality assessment. IEEE Transactions on Circuits and Systems for Video Technology 29(4):1223–1229. https://doi.org/10.1109/TCSVT.2019.2891159
19. Sheikh H, Wang Z, Cormack L, Bovik A (2004) LIVE image quality assessment dataset release 2. http://live.ece.utexas.edu/research/quality
20. Sun C, Li H, Li W (2016) No-reference image quality assessment based on global and local content perception. In: IEEE visual communications and image processing (VCIP), pp 1-4. https://doi.org/10.1109/VCIP.2016.7805544
21. Xiong Y, Shao F, Meng Y, Zhou B, Ho YS (2019) Sparse representation of salient regions for no-reference image quality assessment. IEEE Access 13(5):106295–106306. https://doi.org/10.1177/1729881416669486
22. Ye P, Kumar J, Kang L, Doermann D (2012) Unsupervised feature learning framework for no-reference image quality assessment. In: IEEE conference on computer vision and pattern recognition (CVPR), pp1098–1105. https://doi.org/10.1109/CVPR.2012.6247789
23. Yun Z, Shah M (2006) Visual attention detection in video sequences using spatiotemporal cues. Proceedings of the 14th ACM international conference on multimedia, pp 815-824. https://doi.org/10.1145/1180639.1180824
24. Zhang L, Gu Z, Li H (2013) SDSP: a novel saliency detection method by combining simple priors. In: IEEE international conference on image processing (ICIP), pp 171-175. https://doi.org/10.1109/ICIP.2013.6728036
25. Zhang P, Zhou W, Wu L, Li H (2015) SOM: semantic obviousness metric for image quality assessment. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 2394-2402. https://doi.org/10.1109/CVPR.2015.7298853
26. Zhou ZN, Zhou Z, Huang J (2021) Gauss-guided patch-based deep convolutional neural networks for no-reference image quality assessment. J Intell Fuzzy Syst 41(1):1–10. https://doi.org/10.3233/JIFS-210063
27. Zoph B, Vasudevan V, Shlens J, Le QV (2017) Learning transferable architectures for scalable image recognition. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 8697–8710. https://doi.org/10.48550/arXiv.1707.07012

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.