Color Image Compression Through Rough PCA and Feed Forward Neural Networks
彩色影像壓縮使用概略主要元素及前向神經網路

Jzau-Sheng Lin(林灶生), Tzu-Chiang Shen(沈自強),
Yao-Te Huang(黃耀德), and Jia-Yuan Jean(江佳遠)

Department of Electronic Engineering,
National Chin-Yi Institute of Technology, Taichung, Taiwan, R.O.C.
國立勤益技術學院　電子工程系
Tel: +886-4-3924505 ext 7311　Fax: +886-4-3926610
e-mail: jslin@chinyi.ncit.edu.tw

Abstract

In this paper color image compression using a hybrid neural-network model consisted of rough principal component analysis (RPCA) for separating an image into rough principal components and a feed forward network (FFN) to restore rough principal channels into the original information was proposed. In this system, a training vector is only consisted of two pixels with maximum and minimum gray values in a subdivided block for the input of RPCA net in the encoder. The proposed RPCA generates several channels with black and white image as the original color image but each sample represented by a scale value instead of a three-coordinate vector samples. Then these rough principal components are directly fed into FFN to train the synaptic weights between input and output neurons. Finally, the rough principal components and the synaptic weights are transmitted to the decoder. The experimental results show that the proposed RPCA and feed forward networks can obtain the promising reconstruction performance.

Keyword: Color Image Compression, Rough Set, PCA.

1. Introduction

Multimedia data, consisted of text, audio, and video data, are the major information in computers or various networks such as the Internet. Since the video data occupy the most information in the multimedia data, the researchers have being interest in the field of video compression strategies for the purpose of efficient transmission and less storage size. In a color image the RGB components contain redundant information that can be reduced. The efficient approaches of color image compression is necessary for the purpose of the more efficient storage and faster transmission in the processing of the multimedia data. The applications of color images are used only three primary colors, usually red, green, and blue (RGB). The straightforward way of color image compression is to compress each of red, green and blue gray-scale images that compose the image. A transform coder converts signal vectors to a new coordinate domain for the purpose of reducing the statistical redundancy between vector components.

PCA, often used in data analysis and transform coding, is a popular statistical technique with a linear transformation applied to decorrelate components of vector signals [1-3]. It is a variance maximization process that diagonalizes the covariance matrix of the input pattern distribution. Pearson [4] first introduced and used it in a biological context to recast linear regression analysis into a new form.

Rough set theory, proposed by Zdzislaw Pawlak [5] [6], provides a systematic framework for the study of the problems arising from imprecise and insufficient knowledge. Instead of fuzziness dealing with vagueness between the overlapping sets [7] [8] and fine concepts, the rough sets deal with coarse non-overlapping concepts [9]. In fuzzy sets, each training sample can have only one membership value to a particular class. However, rough sets declare that each training sample may have different membership values to the same class. Therefore, rough sets are of interest to deal with a classification system, in which knowledge about the system is unrefined.

In this paper, the concept of rough sets was embedded in to PCA network to construct a RPCA in order to transform the 3-D color information in spatial domain into several small-size 2-D principal components. In the proposed RPCA, rough neurons make it to process training samples with a range of values instead of a single precise values. Each rough neuron stores the upper and lower bounds of the input and output values. Then, the single output of RPCA, calculated by the upper and lower bounds, is directly fed into FFN to train the synaptic weights between input and output neurons.

The remainder of this paper is organized as follows. Section 2 reviews the rough set theory; Section 3 discusses the PCA and FFN networks; The proposed RPCA is demonstrated in Section 4; Section 5 shows several experimental results; Finally, Section 6 gives the discussion and conclusions.

2. Rough Set Theory

Let *R* be an equivalence relation on a universal set *X* and *X*/*R* denote the family of all equivalence classes induced on *X* by *R*. There exits an equivalence class in *X*/*R*, designated by $[x]_R$, that contains a training sample $x \in X$. For any output class $A \subseteq X$, the lower approximation $\underline{R}(A)$ and upper approximation $\overline{R}(A)$ are defined and they approach *A* as closely as possibly from inside and outside in the set respectively [10]. Therefore, $\underline{R}(A)$ can be defined as the union of all equivalence classes in *X*/*R* that are contained in *A* such that

$$\underline{R}(A) = \cup\{[x]_R \mid [x]_R \subseteq A, \;\; x \in X\}, \tag{1}$$

whereas $\overline{R}(A)$ can be also defined as the union of all equivalence classes in *X*/*R* that overlap with A like the following equation

$$\overline{R}(A) = \cup\{[x]_R \mid [x]_R \cap A \neq \phi, \;\; x \in X\}. \tag{2}$$

A rough set can be represented by $\underline{R}(A)$ and $\overline{R}(A)$ with the given set *A* as

$$R(A) = \left\langle \overline{R}(A), \underline{R}(A) \right\rangle. \tag{3}$$

And the rough boundary of *A* by the equivalence classes *X*/*R* is distinct as

$$RB(A) = \overline{R}(A) - \underline{R}(A). \tag{4}$$

3. PCA and FFN Learning Networks

Principal component analysis is a strategy used to summarize the properties of multivariate analysis in a set of data patterns. It is a linear transformation method often used for feature extraction or data compression. PCA, also known as the Karhunen-Loeve transformation [11] in communication theory, maximizes the rate of decrease of variance. From the perspective of statistical pattern recognition, the practical value of principal components analysis is that it supplies an effective method for dimensionality reduction. The number of features needed for effective data representation can be reduced. On the other hand, it discards those terms that have small variances and retains only those terms that have large variances [12]. Given input training vector in a $3\times\ell\times\ell(=\alpha)$-dimensional (48 dimensions in a 4×4 block) vector space for a RGB color image, we want to find some subset $\beta < 3\times\ell\times\ell$ of the $3\times\ell\times\ell$ variants that account for as much of the variance of data as possible. To accomplish this scheme, the $3\times\ell\times\ell$ -dimensional observed space is projected onto β-dimensional feature space to choose each component in the direction of maximum variance and such that the components are mutually orthogonal. Therefore we have smaller dimensionality in the feature space than those in the observed data that is the total processing time can be greatly reduced if the data processing is performed in the feature space.

PCA technique can be modeled by the following. There exists a α-dimensional input vector $\mathbf{X} \in \mathrm{R}^{\alpha}$ and a β-dimensional output component **Y,** a matrix **W** can be defined such that $\mathbf{Y} = \mathbf{WX} \in \mathrm{R}^{\beta}$ ($\beta < \alpha$). Then an inverse $\mathbf{Z} = (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T$ can also be created so as to estimate the input vector such that $\mathbf{X}' = \mathbf{ZY}$. Therefore, matrix **Z** should be processed such that the square error, defined in Eq. (5), is minimized.

$$J = \sum (\mathbf{ZY} - \mathbf{X})^T (\mathbf{ZY} - \mathbf{X}) \tag{5}$$

A neural network based on PCA [13] can determine matrix **W**. The output state of neuron y_i at iteration *t* can be expressed by the input vector $\{x_j \mid j = 1,2,3,\ldots,\alpha\}$ as

$$y_i(t) = \sum_{j=1}^{\alpha} w_{i,j}(t)x_j(t) \tag{6}$$

Sanger [14] indicated in 1989 that synaptic weight $w_{i,j}(t)$ can be updated with the following equation.

$$\Delta w_{i,j}(t) = \eta \left[y_i(t)x_j(t) - y_i(t)\sum_{k=1}^{i} w_{k,j}(t)y_k(t) \right] \tag{7}$$

where η is the learning-rate parameter. In the learning process of PCA net, the stable state can be reached when we increase *t* and iteratively update y_i and $w_{i,j}(t)$ respectively.

Matrix **Z** can then be determined by using a FFN. The purpose of FFN is to preserve the features of principal components so as to estimate the original image. The hybrid neural network consists of PCA and FFN (called PCAF) was proposed by [15] to apply in the color image compression and shown in Figure 1.

4. Rough PCA Neural Network

In this paper neurons in the PCA are replaced by rough neurons to create the RPCA net in which rough patterns can be processed. The value in a rough pattern is a pair of upper and lower bounds. A rough neuron *s* can be viewed as a pair neurons called *upper neuron*($\overline{s}$) and *lower neuron*($\underline{s}$). In 1998, Lingras [16] proposed three models of rough neurons. A *full-connected* model that displays a rough neuron *r* connects to a proper rough neuron *s* with four connections. If the rough neuron *r* *excites* the activity of *s*, the connections from $\overline{r}$ to $\overline{s}$ and from $\underline{r}$ to $\underline{s}$ are created respectively. On the other hand, if *r* *inhibits* the activity of *s*, the connections from $\overline{r}$ to $\underline{s}$ and from $\underline{r}$ to $\overline{s}$ are created respectively. In this paper, the excited-model rough neuron, shown in Figure 3, is applied in to RPCA. Therefore, the inputs in the excited-model rough neuron are

$$input_{\overline{s}} = \sum_{r=1}^{m} w_{\overline{s},\overline{r}} \cdot output_{\overline{r}} \tag{8}$$

and

$$input_{\underline{s}} = \sum_{r=1}^{m} w_{\underline{s},\underline{r}} \cdot output_{\underline{r}} \ . \tag{9}$$

The outputs of a rough neuron *s* are expressed using an activation function as

$$output_{\overline{s}} = \max(\tau(input_{\overline{s}}), \tau(input_{\underline{s}})) \tag{10}$$

and

$$output_{\underline{s}} = \min(\tau(input_{\overline{s}}), \tau(input_{\underline{s}})) \ , \tag{11}$$

where activation function can be defined as

$$\tau(x) = \frac{1}{1+e^{-cx}} \tag{12}$$

and *c* is a constant. To simplify the architecture of the rough neuron in the RPCA, the outputs in a rough neuron are directly transmitted by their inputs and defined as

$$output_{\overline{s}} = input_{\overline{s}} = \sum_{r=1}^{m} w_{\overline{s},\overline{r}} \cdot output_{\overline{r}} \tag{13}$$

and

$$output_{\underline{s}} = input_{\underline{s}} = \sum_{r=1}^{m} w_{\underline{s},\underline{r}} \cdot output_{\underline{r}} \ . \tag{14}$$

Therefore the upper and lower training samples in a range of values are also directly fed into the upper and lower bounds in a rough neuron *r*. These rough patterns are multiplied their own synaptic weights and fed to the rough neuron *s* in the next layer (principle component layer). Finally, the output of a neuron y_s in the RPCA is also simplified as

$$y_s(t) = \frac{1}{2}(output_{\overline{s}} + output_{\underline{s}}) \ . \tag{15}$$

The synaptic weights between neurons *s* and *r*, $w_{\overline{s},\overline{r}}(t)$ and $w_{\underline{s},\underline{r}}(t)$, can be updated with the following equations.

$$\Delta w_{\overline{s},\overline{r}}(t) = \eta \Big[output_{\overline{s}}(t) output_{\overline{r}}(t) - output_{\overline{s}}(t) \sum_{k=1}^{s} w_{\overline{k},\overline{r}}(t) output_{\overline{k}}(t) \Big] \tag{16}$$

and

$$\Delta w_{\underline{s},\underline{r}}(t) = \eta \Big[output_{\underline{s}}(t) output_{\underline{r}}(t) - output_{\underline{s}}(t) \sum_{k=1}^{s} w_{\underline{k},\underline{r}}(t) output_{\underline{k}}(t) \Big] . \tag{17}$$

The completed system for the proposed RPCA is shown in the left picture of Figure 3. Moreover, the proposed RPCA approach can reduce the computation time in compression process. Not only do the proposed technique preserve the reconstruction performances, it can also reduce one half training information in the input data.

5. Experimental Results

In simulations, the conventional PCA + FFN (PCAF) and the proposed RPCA + FFN (RPCAF) are used to compare their performances. A color image was first blocked 4×4 pixels in which 4 subblocks with 2×2 pixels (m=12) were separated in the proposed RPCA whereas the image just blocked 4×4 pixels ($\alpha = 48$) in the original PCA net. In each subblock a pair of pixels with lower and upper bounds were selected as training samples in the RPCA. Thus an input vector consisted of $3\times4\times2=24$ bytes in the proposed RPCA instead of $3\times4\times4=48$ bytes in the original PCA net. Finally, 4 principal components with gray value were transformed and fed to train and update the synaptic weights in the FFN. In the RPCA a output component occupies lower and upper bound value to be combined as a single component. Therefore the architecture of the RPCA can be simplified and only owns $2\times12\times4=96$ interconnect weights whereas the conventional PCA processes $48\times4=192$ interconnect weights. The training vectors were extracted from 512$\times$512 real color images with 24-bit true colors in RGB system and simulated in a Pentium-III computer.

The reconstructed images are displayed in Figures 4 and 5 with distinct PCA techniques. From Figures 4 and 5, the reconstructed image using the RPCA + FFN is almost sensed same as the one reconstructed by the conventional PCA + FFN. To demonstrate the reconstruction performance, the root mean squared error (RMSE) and peak signal to noise ratio (PSNR) for each coordinate, evaluated in the reconstructed image and original image, are defined as following equations

$$RMSE = \sqrt{\frac{1}{N^2}\sum_{x=1}^{N}\sum_{y=1}^{N}\left[f(x,y)-\hat{f}(x,y)\right]^2} \quad (18)$$

and

$$PSNR = 10\log_{10}\frac{255\times255}{RMSE^2} \quad (19)$$

where $f(x,y)$, $\hat{f}(x,y)$, and 255 are original coordinate, reconstructed coordinate, and peak value of pixels in a coordinate. The global channel RMSE $RMSE_T$ [16] is denoted as follows:

$$RMSE_T = \sqrt{\frac{1}{3}(RMSE_R^2 + RMSE_G^2 + RMSE_B^2)} \quad (20)$$

where $RMSE_R$, $RMSE_G$, and $RMSE_B$ are the RMSE for red, green and blue components, respectively. Tables 1 shows the reconstruction performances (PSNR) for each coordinate and global channel in different images by using the proposed PCA + FFN and RPCA + FFN approaches respectively. The last column in Tables 1 named T defines the global RMSE. Generally these two methods can also obtain almost the same reconstruction performances those are shown in pictures (b) and (c) of Figure 1. The experimental results for different images with PSNRs are shown in Table 1. For example, in image “Lenna” the PSNRs for red, green, and blue coordinates are 29.01, 28.54, and 30.64 dBs using PCAF as well as 28.94, 28.45, and 30.53 dBs using RPCAF respectively. Additionally, the PSNRs for the global channel combining all of coordinates are 29.30 and 29.22 dBs for PCAF and RPCAF individually. The distances for all PSNRs between these two methods are so small (less than 0.1 dBs) that they can be ignored. That is these two techniques can almost obtain the same reconstruction performances.

6. Discussion & Conclusions

A hybrid neural network consisted of RPCA and FFN nets is proposed to the color image compression in this paper. The original color images can be divided into several subblocks to obtain the upper and lower bounds and feed into the input of rough neurons. In addition to reducing the training samples, the architecture in the synaptic interconnections are also simplified in the proposed RPCAF to decrease the computation time in learning process. Although the architecture is simplified with reducing data, the reconstructed performances are also preserved like obtained by the conventional PCAF.

Acknowledgement

This work was partially supported by the National Science Council of the R.O.C. under grand NSC 90-2213-E-167-003 and the Lab. of image compression in the Dept. of Electronic Engineering, National Chin-Yi Institute of Technology, Taichung, Taiwan.

References

[1] R. W. Preisendorfer, *Principal Component Analysis in Meteorology and Oceanography*, New York: Elservier, 1988.

[2] I. T. Jollife, *Principal Component Analysis*, New York: Springer-Verlag, 1988.

[3] M. Tipping and C. Bishop, "Mixture of Probabilistic Principal Component Analysis," *Neural Computation*, vol. 11, pp. 443-483, 1999.

[4] K. Pearson, "On Linear and Planes of Closest Fit to System of Points in Space," *Philosophical Magazine*, vol. 2, 1901.

[5] Z. Pawlak, 'Rough Sets," *Int. J. of Information and Computer Science*, vol. 11, pp. 145-172, 1982.

[6] Z. Pawlak, 'Rough Classification," *Int. J. of Man-Machine Studies*, vol. 20, pp. 469-483, 1984.

[7] A. Kandle, *Fuzzy Mathematical Techniques with Applications*, Reading, Massachusetts: Addison-Wesley Publishing Company, 1986.

[8] J. C. Bezdek, "The Thirsty Traveller Visits Gamont: A Rejoinder to "Comments on Fuzzy Set-What Are They and Why?"," *IEEE Trans. on Fuzzy Systems*, vol. 2, pp. 43-45, 1994.

[9] D. Dubois and H. Prade, "Rough Fuzzy Sets and Fuzzy Rough Sets," *Int. J. of General Systems*, vol. 17, pp. 191-209, 1990.

[10] G. S. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic – Theory and Applications*, Englewood Cliffs, NJ: Prentice-Hall, 1995.

[11] M. Effros and P. Chou, "Weighted Universal Transform Coding: Universal Image Compression with the Karhunen-Loeve Transform," Proc. International Conf. on Image Processing, pp. 61-64, 1995.

[12] E. Oja, "Subspace Methods of Pattern recognition," Letchworth, UK: Research Studies Press, 1983.

[13] S. Haykin, *Neural Networks: A Comprehensive Foundation*, New York: Macmillan College Publishing Company, pp. 363-394, 1994.

[14] T. D. Sanger, "Optimal Unsupervised Learning in a Single-layer Linear Feed forward Neural Network," *Neural Networks*, vol. 2, pp. 459-473, 1989.

[15] C. Clausen and H. Wechsler, "Color Image Compression Using PCA and Backpropagation Learning," *Pattern Recognition*, vol. 33, pp. 1555-1560, 2000.

[16] P. Lingras, "Comparison of Neofuzzy and Rough Neural Networks," *Information Sciences*, vol. 110, pp. 207-215, 1998.

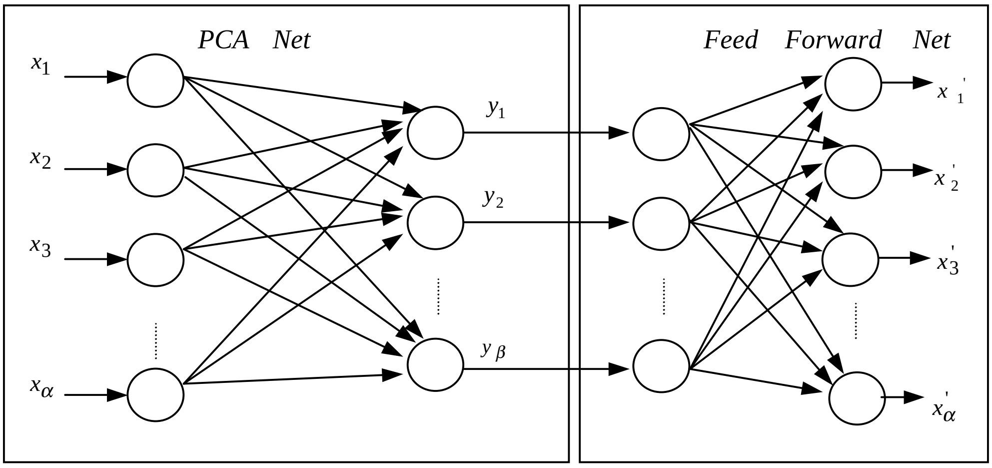


Figure 1. The completed architecture of PCA and FFN in color image compression.

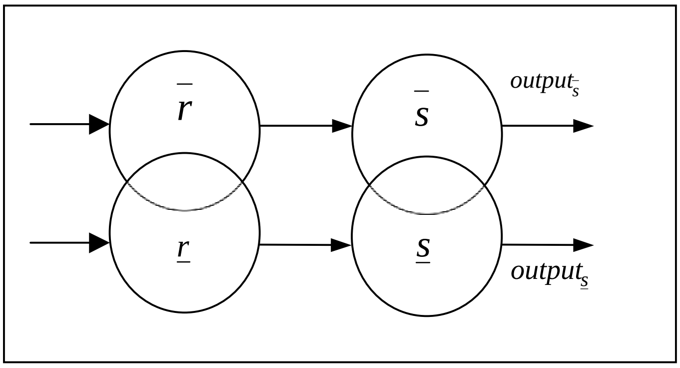


Figure 2. Interconnection in the architecture of rough neurons.

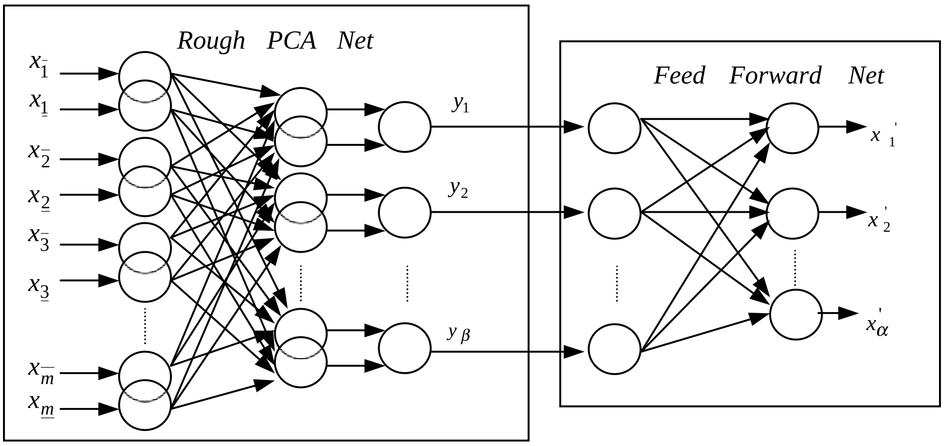


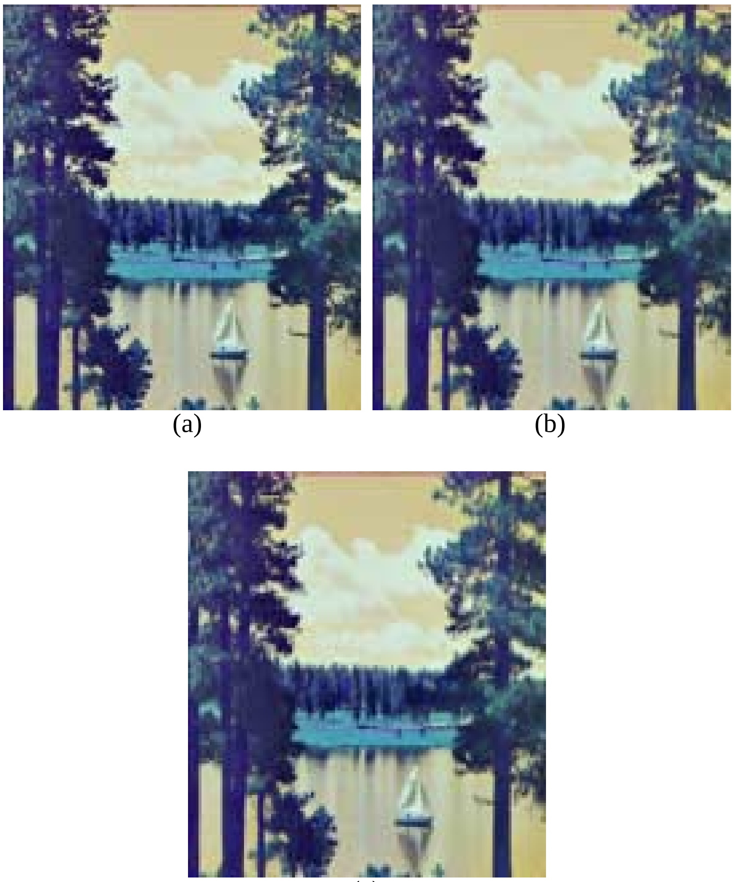
Figure 3. The completed architecture of RPCA and FFN in color image compression.


(a)


(b)


(c)

Figure 4. Image "Lenna" and its reconstructed images with 4×4 pixels and 4 principal components: (a) Original image. (b) Reconstructed image using conventional PCA (c) Reconstructed image using RPCA.



(a) (b)

(c)

Figure 5. Image “Sailboat” and its reconstructed images with 4×4 pixels and 4 principal components: (a) Original image. (b) Reconstructed image using conventional PCA (c) Reconstructed image using RPCA.

Table 1. The reconstruction performance PSNRs for different pictures by using the PCA + FFN and RPCA + FFN nets with 4×4 pixels 4 principal components.

Picture	R		G		B		T	
	PCAF	RPCAF	PCAF	RPCAF	PCAF	RPCAF	PCAF	RPCAF
Lenna	29.01	28.95	28.54	28.45	30.63	30.53	29.30	29.22
Peppers	27.81	27.76	25.51	25.42	27.71	27.68	26.87	26.81
F16	27.46	27.43	25.18	25.09	26.08	26.07	26.14	26.09
Sailboat	23.62	23.56	23.13	23.07	26.92	26.89	24.26	24.21
Average PSNR	**26.98**	**26.93**	**25.59**	**25.51**	**27.84**	**27.79**	**26.64**	**26.58**