A Non-linear Image Enhancement Method By Spline Basis Filter

Lung-Jen Wang
Dept.of Information Technology
National Pingtung Institute of Commerce

Kun-Rong Shieh, Ying-Lun Tang
Dept.of Information Management
National Pingtung Institute of Commerce

ABSTRACT

In this paper, we propose a spline basis filter to improve the non-linear image enhancement method. In addition, a higher-frequency component is predicted to solve the blurred problem of an enlarged image. Furthermore, the proposed enhancement method can be used along with the video coding to improve the quality of decoded image. Experimental result shows that the proposed method yields a better quality of the reconstructed image than other non-linear enhancement methods.

1: INTRODUCTIONS

With the growing interest of digital image processing, the applications in this domain such as digital high definition television (HDTV) and video-phone, are an integral part of our life. These applications relate to a scale image enlargement technique. Typically, however, the image enlargement causes a blurred image because the high-frequency component of enlarged image is not enough [1]. To improve the quality of such blurred image, the image enhancement is an indispensable post-processing method.

Image Enhancement is very good topic in the researches, the principle of image enhancement is to process an image so that the result is more suitable than the original image in many applications. A typical image enhancement is achieved througth the high-pass filter to make the image suitable. This method uses a typical principle behind unsharp masking and high-boost filtering [1], and is shown in Fig.1.

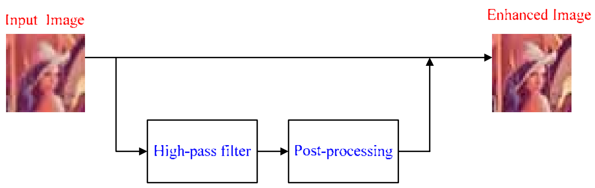


Fig.1. A typical image enhancement with high-pass filter and post-processing

Non-linear image enhancement [2]-[4] is similar to the typical image enhancement except the high-pass filter is replaced by non-linear operations. This enhancement method uses the Gaussian-pyramid [2] or FSD(filter subtract and decimate)-pyramid [3] representation of an image to extract the high-frequency component of original image shown in Fig.2. The major non-linear step involves clipping and scaling the extracted components.

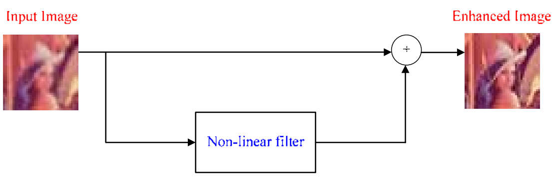


Fig.2. Non-linear image enhancement

In this paper, we want to propose a better image enhancement method that uses a cubic B-spline filter for the non-linear image enhancement. In addition, this method is used along with video coding to improve the quality of decoded image. Finally, some experimental results show that the proposed method obtains a better subjective quality and PSNR performance than other non-linear image enhancement methods.

2: SPLINE BASIS FUNCTIONS

Spline basis function [5]-[7] is a very good low-pass filter, equal to B-spline function. The B-splines (where the B may stand for basis or basic) are the basic building blocks for splines.

Definition[5]: Let $\xi : \xi_0 < \xi_1 < \cdots < \xi_n < \xi_{n+1}$ be a partition of the interval $[\xi_0, \xi_{n+1}]$ on a real axis. The spline basis function of degree n on ξ is the following piecewise polynomial:

$$B_n(\xi;\xi_0,\xi_1,\xi_2,\cdots,\xi_{n+1}) = (n+1)\sum_{k=0}^{n+1}\frac{(\xi-\xi_k)^n U(\xi-\xi_k)}{\omega(\xi_k)}, \text{ for } n = 0,1,2,\cdots \quad (1)$$

where

$$\omega(\xi_k) = \prod_{\substack{j=0 \\ j\neq k}}^{n+1}(\xi_k - \xi_j)$$

and

$$U(\xi-\xi_k) = \begin{cases} (\xi-\xi_k)^0, & \text{for } \xi > \xi_k \\ 0, & \text{for } \xi \le \xi_k \end{cases}$$

is a unit step function.

A sketch of the four lower order spline basis function for uniformly spaced data points is shown in Fig.3.

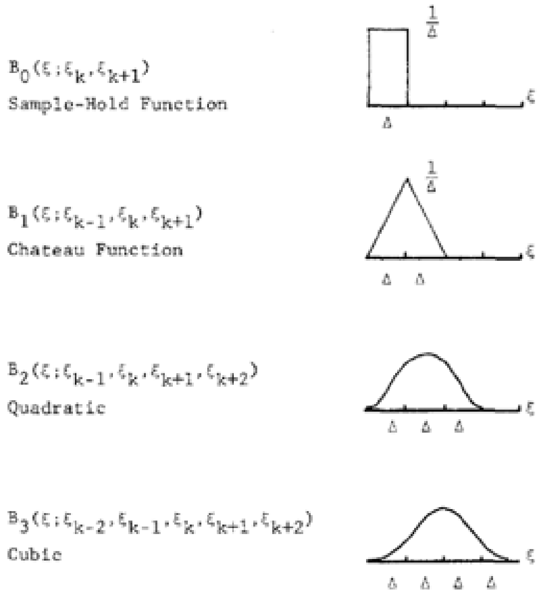


Fig 3. The spline basis function

From (1), the cubic B-spline function is given by

$$\begin{aligned} S_3(\xi-\xi_k) &\equiv B_3(\xi;\xi_{k-2},\xi_{k-1},\xi_k,\xi_{k+1},\xi_{k+2}) \\ &= [(\xi-\xi_{k-2})^3U(\xi-\xi_{k-2}) - 4(\xi-\xi_{k-1})^3U(\xi-\xi_{k-1}) \\ &\quad + 6(\xi-\xi_k)^3U(\xi-\xi_k) - 4(\xi-\xi_{k+1})^3U(\xi-\xi_{k+1}) \\ &\quad + (\xi-\xi_{k+2})^3U(\xi-\xi_{k+2})]/6\Delta^4, \end{aligned} \tag{2}$$

where $\Delta = \xi_k - \xi_{k-1}$.

An interpolation function can be expressed in the following form

$$\hat{f}(\xi) = \sum_{k=1}^{N} c_k S_k(\xi-\xi_k) \tag{3}$$

where c_k are the coefficients to be determined from the input data, $S_k(\xi-\xi_k)$ is the spline basis function, and *N* is the number of given data points.

From (1)-(3) and illustrated in Fig.4., the cubic B-spline interpolation function in one-dimension(1-D) can be obtained as

$$\begin{aligned} \hat{f}(\xi) = \{ & c_{k-1}[(\xi-\xi_{k-3})^3 - 4(\xi-\xi_{k-2})^3 + 6(\xi-\xi_{k-1})^3 - 4(\xi-\xi_k)^3] \\ & + c_k[(\xi-\xi_{k-2})^3 - 4(\xi-\xi_{k-1})^3 + 6(\xi-\xi_k)^3] \\ & + c_{k+1}[(\xi-\xi_{k-1})^3 - 4(\xi-\xi_k)^3] \\ & + c_{k+2}[(\xi-\xi_k)^3]\}/6\Delta^4, \end{aligned} \tag{4}$$

where c_{k-1}, c_k, c_{k+1} and c_{k+2} are the coefficients to be determined from the input data.

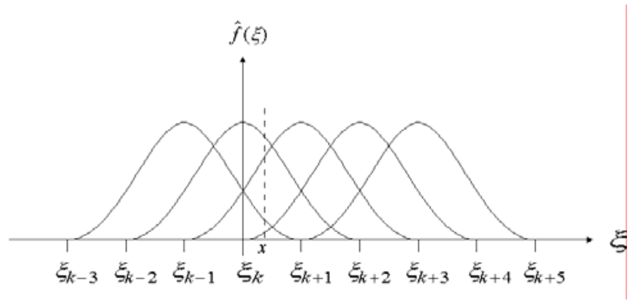


Fig.4. $\hat{f}(\xi)$ interpolated by cubic B-spline.

Let $\xi = \xi_k + x\Delta$, where $0 \le x \le 1$. The cubic B-spline interpolation function in (4) can be written by

$$\begin{aligned} \hat{f}(\xi_k + x\Delta) &= \{c_{k-1}[(3+x)^3 - 4(2+x)^3 + 6(1+x)^3 - 4x^3] + c_k[(2+x)^3 \\ &\quad - 4(1+x)^3 + 6x^3] + c_{k+1}[(1+x)^3 - 4x^3] + c_{k+2}x^3\}/6\Delta \\ &= \{x^3(c_{k+2} - 3c_{k+1} + 3c_k - c_{k-1}) + x^2(3c_{k+1} - 6c_k + 3c_{k-1}) \\ &\quad + x(3c_{k+1} - 3c_{k-1}) + (c_{k+1} + 4c_k + c_{k-1})\}/6\Delta. \end{aligned} \tag{5}$$

Then (5) can be used to find the interpolation at any point between sampled points. In particular, at the node point $\xi = \xi_k$, i.e., $x = 0$, (5) becomes

$$\hat{f}(\xi_k) = (c_{k+1} + 4c_k + c_{k-1})/6\Delta. \tag{6}$$

In other words, the 1-D filter of cubic B-spline function in (6) is [1, 4, 1] / 6.

Based on the definition of the two-dimensional (2-D) interpolation function [6], the cubic B-spline interpolation function can be extended from the 1-D interpolation function to the 2-D interpolation function. Let $\hat{f}(\xi,\eta)$ be the 2-D cubic B-spline interpolation function at the point ($\xi = \xi_k + x\Delta$, $\eta = \eta_l + y\Delta$), where $0 \le x \le 1$ and $0 \le y \le 1$. By (5), one obtains

$$\begin{aligned} \hat{f}(x,y) = \{ & \hat{f}_{l-1}(x)[(3+y)^3 - 4(2+y)^3 + 6(1+y)^3 - 4y^3] \\ & + \hat{f}_l(x)[(2+y)^3 - 4(1+y)^3 + 6y^3] \\ & + \hat{f}_{l+1}(x)[(1+y)^3 - 4y^3] + \hat{f}_{l+2}(x)y^3\}/6\Delta. \end{aligned} \tag{7}$$

In particular, at the node point (ξ_k, η_l), i.e., $x = 0$ and $y = 0$, (7) gives

$$\begin{aligned} \hat{f}(\xi_k,\eta_l) = \{ & (c_{k-1,l-1} + 4c_{k,l-1} + c_{k+1,l-1}) \\ & + 4(c_{k-1,l} + 4c_{k,l} + c_{k+1,l}) \\ & + (c_{k-1,l+1} + 4c_{k,l+1} + c_{k+1,l+1})\}/36\Delta^2, \end{aligned} \tag{8}$$

for all $k = 1, 2, \cdots, K$ and $l = 1, 2, \cdots, L$.

In other words, the 2-D filter of cubic B-spline function in (8) is [1, 4, 1; 4, 16, 4; 1, 4, 1] / 36.

3: THE PROPOSED ENHANCEMENT SCHEME

The proposed enhancement algothrim in this paper uses the same philosophy of non-linear image enhancement method in [3], which is illustrated in Fig.5. The low-frequency image I_1 is obtained using the cubic B-spline filter, and the high-frequency image K_0 is the residual image of the input-blurred image I_0 and the low-frequency image I_1. The enhanced image I_{-1} is predicted in the following:

$$I_{-1} = I_0 + NL(S_3(I_0) - I_0) \quad (9)$$

where $S_3(\bullet)$ is the cubic B-spline filter, and $NL(\bullet)$ is a non-linear operator [3], which includes both scaling and clipping steps, defined as follows:

$$NL(x) = s \times Clip(x) \quad (10)$$

where s is a scaling constant and $Clip(x)$ is given by

$$Clip(x) = \begin{cases} T, & if \ x > T \\ x, & if \ -T \le x \le T \\ -T, & if \ x < -T \end{cases} \quad (11)$$

where x is the pixel of the high-frequency image K_0, $T = c \times K_{0\max}$, $K_{0\max}$ is the maximum pixel of the high-frequency image K_0 and c is clipping constant ranging between 0 and 1.

After non-linear operator, the higher-frequency image K_{-1} can be added to the input-blurred image I_0 to enhance the input-blurred image I_0, and obtain the enhanced image I_{-1}.

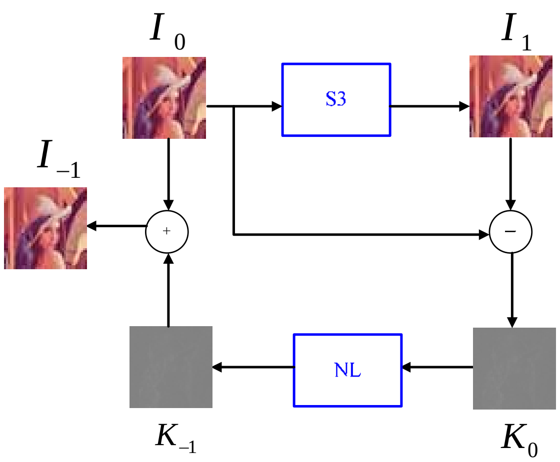


Fig.5. The proposed non-linear image enhancement

4: A 3-D DOWN-SCALING FOR VIDEO CODING

In order to obtain a very low bit-rate video, a new type of three-dimensional (3-D) down-scaling scheme is presented for video coding. This scheme applies a 3-D decimation with a compression ratio of 8 to 1 as the pre-processing step of the encoder. As a consequence, a 3-D interpolation with a ratio of 1 to 8 is used for the post-processing step of the decoder.

4.1: A 3-D Linear Decimated Scheme

Let t_1, t_2 and t_3 be the integer indices and n_1, n_2 and n_3 are also integers. The 3-D decimated scheme takes an video $X(t_1, t_2, t_3)$ as an input and produces an output of $Y(t_1, t_2, t_3)$ by a factor of 2 in each dimension as follows:

$$Y(t_1, t_2, t_3) = \text{avg}\left(\sum_{i_1=0}^{1}\sum_{i_2=0}^{1}\sum_{i_3=0}^{1} X(2t_1 + i_1, 2t_2 + i_2, 2t_3 + i_3)\right) \quad (12)$$

for $0 \le t_i \le n_i - 1, \quad i = 1,2,3.$

where $avg(\bullet)$ is returns the average (arithmetic mean) of a set of numeric values. Fig.6. shows the down-sampling of pre-processing stage using the 3-D linear method.

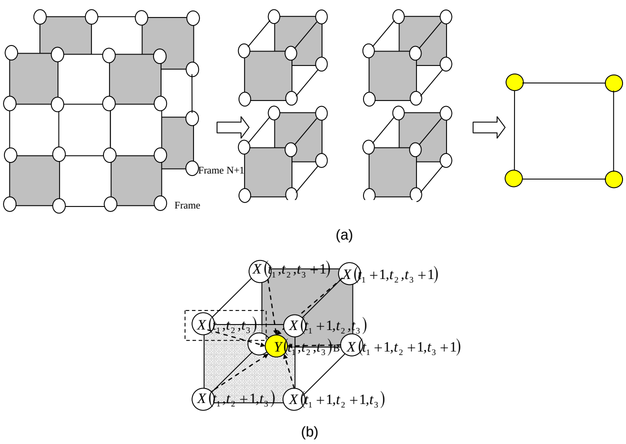


Fig.6. A 3-D linear decimated scheme

4.2: A 3-D Linear Interpolated Scheme

Using the decimated video $Y(t_1, t_2, t_3)$ obtain by (12), the 3-D reconstructed video can be calculated by a linear interpolation shown in Fig.7. and given by

$$\hat{X}(t_1, t_2, t_3) = \sum_{k_1=0}^{1}\sum_{k_2=0}^{1}\sum_{k_3=0}^{1} Y(k_1, k_2, k_3) R(t_1 - 2k_1, t_2 - 2k_2, t_3 - 2k_3) \quad (13)$$

for $0 \le t_i \le 3, \quad i = 1,2,3.$

where $R(t_1 - 2k_1, t_2 - 2k_2, t_3 - 2k_3)$ is the 3-D linear functions defined by

$$R(t_1, t_2, t_3) = R(t_1)R(t_2)R(t_3) \tag{14}$$

and $R(t)$ is the 1-D linear function given by

$$R(t) = \begin{cases} 1 - |t|/2 & , |t|/2 \\ 0 & , otherwise \end{cases} \tag{15}$$

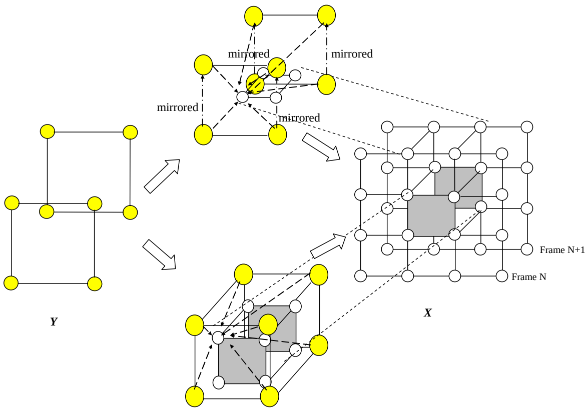


Fig.7. A 3-D linear interpolated scheme

5: 3-D DOWN-SCALING FOR VIDEO CODING USING NON-LINEAR IMAGE ENHANCEMENT BY SPLINE BASIS FILTER

The 3-D down-scaling for video coding provides a better performance at a lower bit-rate transmission, however, this method causes the blurred problem of decoded image. In this section, the proposed non-linear image enhancement with the cubic B-spline filter is used to improve the decoded quality of the 3-D down-scaling for video coding shown in Fig.8. In this figure, this algorithm applies the 3-D linear decimation (3D-LI$\downarrow$) as the encoder, and the 3-D linear interpolation (3D-LI $\uparrow$) as the decoder for video coding. As a consequence, the proposed non-linear image enhancement with cubic B-spline filter is used for the post-processing step of this decoder.

For this algorithm, an original video sequence in the RGB color space is converted into another preliminary sequence in YUV [6] color space prior to the 3D-LI$\downarrow$ processing. Finally, the proposed non-linear image enhancement is used for the Y component only, and it is not used for the U and V components.

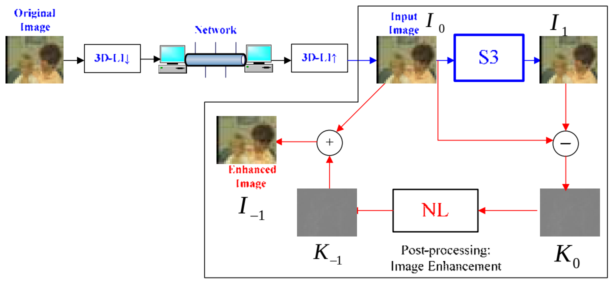


Fig.8. 3-D down-scaling for video coding using non-linear image enhancement

6: EXPERIMENTAL RESULTS

In this section, some experimental results of three gray images (Aerial, Baboon, Barbara), three color images (Lena, Peppers, Sailboat) and three video sequences (Table, Stefan, Mother) are presented. First, these gray and color images are blurred to low-resolution images, then the proposed method and the methods in [2] and [3] are used to enhance these low-resolution images, in addition, the PSNR of these experimental results are compared in Table I and Table II. Furthermore, the 3-D down-scaling for video coding is used along with the proposed method and the methods in [2] and [3] to enhance the reconstructed images of the above three video sequences, finally, the PSNR of these experimental results are also compared in Table III. Obviously, in Tables I, II and III, the quality of reconstructed images using the proposed method is better than the methods in [2] and [3].

Note that in clipping and scaling parameters, we choose s=5, c=4 for both gray and color images, but for 3-D down-scaling for video coding we choose s=3, c=0.45, because these parameters cause the better PSNR results.

TABLE I. PSNR (dB) of gray enhanced image of size 512×512 for proposed method, Gaussian[2], and FSD[3].

Image Name	Blur image	Proposed Method	FSD		Gaussian
Parameter value		s=5,c=0.4	s=3,c=0.45	s=5,c=0.4	$0.04 \times G_{0max}$
Aerial	25.5691	28.4556	27.9413	25.1179	25.8064
Baboon	21.8767	23.5041	23.4870	22.5894	22.1997
Barbara	24.6827	25.7195	25.5772	24.8687	24.5361

TABLE II. PSNR (dB) of color enhanced image (Y) of size 512×512 for proposed method, Gaussian[2], and FSD[3].

Image Name	Blur image	Proposed Method	FSD		Gaussian
Parameter value		s=5,c=0.4	s=3,c=0.45	s=5,c=0.4	$0.04 \times G_{0max}$
Lena	31.9667	34.8130	34.3645	30.7871	30.7871
Peppers	29.7998	31.4374	31.1828	29.2849	28.9743
Sailboat	27.6185	29.9329	29.6577	26.9715	27.4305

TABLE III. PSNR (dB) of enhanced sequence (Y) of size 352×288 by 3-D down-scaling for proposed method, Gaussian[2], and FSD[3].

Sequence Name	3-D linear compression	Proposed Method	FSD		Gaussian
Parameter value		s=3,c=0.45	s=3,c=0.45	s=5,c=0.4	$0.04\times G_{0\max}$
Table	24.07	24.94	24.54	22.92	24.26
Stefan	21.43	22.04	21.93	20.64	21.60
Mother	33.97	35.19	33.74	29.93	32.74

Finally, in Figs.9, 10 and 11, clearly, the proposed method obtains a better subjective quality to the original image. However, the FSD method is a clearest algorithm, but it causes the ring effect for the enhanced image because the higher-frequency component is enhanced too much, and the Gaussian method is inferior in performance to the proposed method and the FSD method.

7: CONCLUSION

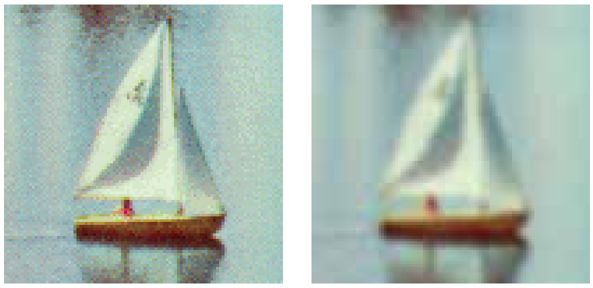
In this paper, the proposed method uses the cubic B-spline filter and non-linear image enhancement scheme to improve the blurred image and video sequence. Such a new method is not only efficient but also superior in performance to the other image enhancement methods. Experimental results also show that this method yields a better subjective quality and objective PSNR than other non-linear image enhancement methods for the reconstructed image.

(a) (b)

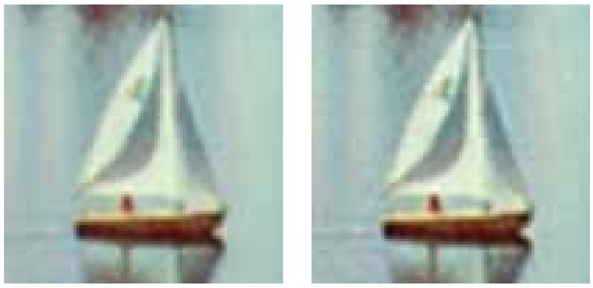
(c) (d)

(e) (f)

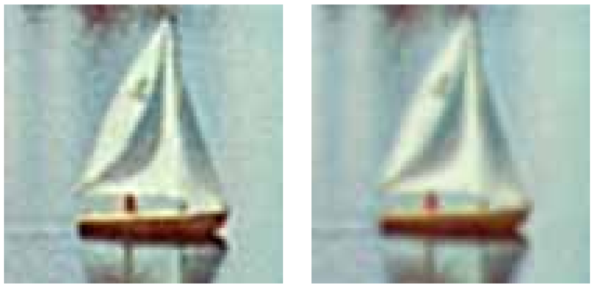
Fig.9.(a)Original Barbara image(zoom in), (b)Blurred Barbara image, (c)proposed method(s=5,c=0.4), (d)FSD method(s=3,c=0.45), (e)FSD method(s=5,c=0.4), (f)Gaussian method($0.04\times G_{0\max}$).



(a) (b)



(c) (d)



(e) (f)

Fig.10.(a)Original Sailboat image(zoom in), (b)Blurred Sailboat image, (c)proposed method(s=5,c=0.4), (d)FSD method(s=3,c=0.45), (e)FSD method(s=5,c=0.4), (f)Gaussian method($0.04\times G_{0\max}$).

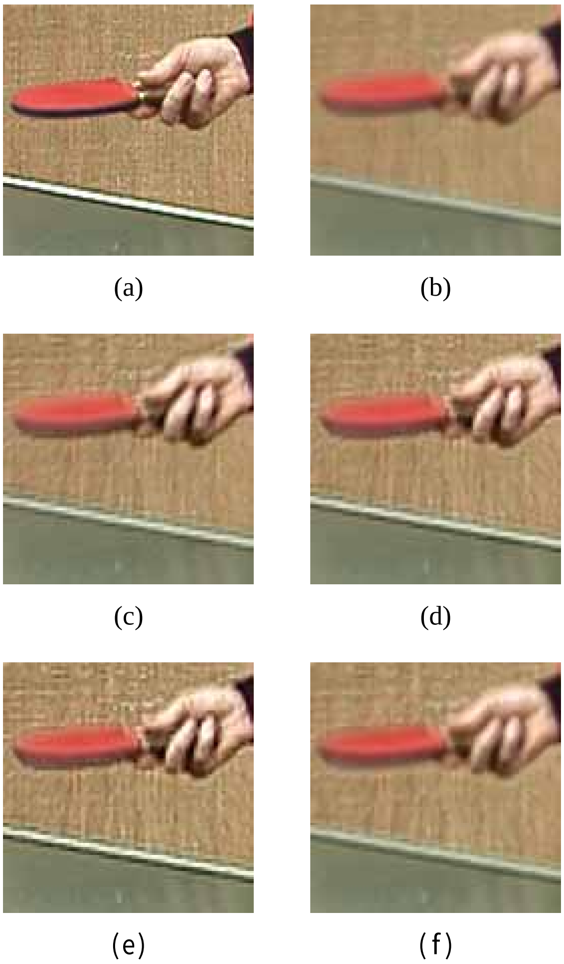


Fig.11.(a)Original Table sequence(zoom in), (b) 3-D down-scaling video coding, (c)proposed method(s=3,c=0.45), (d)FSD method(s=3,c=0.45), (e)FSD method(s=5,c=0.4), (f)Gaussian method($0.04 \times G_{0\max}$).

ACKNOWLEDGMENT

This work was supported by the National Science Council, R.O.C., under Grant NSC 95-2221-E-251-003.

REFERENCES

[1] S. Yuan, A. Taguchit, and M. Kawamata, “Arbitrary scale image enlargement with the prediction of high frequency components,” *in Proc. of IEEE International Symposium on Circuits and Systems*, vol. 6, pp.6264-6267, 23-26 May 2005.

[2] H. Greenspan, “Multi-resolution image processing and leaming for texture recognition and image enhancement,” *Ph.D. thesis, California Inst. of Technol.*, 1994.

[3] H. Greenspan, C. H. Anderson and S. Akber, “Image enhancement by nonlinear extrapolation in frequency space,” *IEEE Trans. on Image Processing*, vol. 9, no. 6, pp. 1035-1048, June 2000.

[4] Y. Yang, B. Li, “Non-linear image enhancement for digital TV applications using Gabor filters,” *in Proc. of IEEE International Conference on Multimedia and Expo*, 6-8 July 2005.

[5] H. S. Hou and H. C. Andrews,“Cubic spline for image interpolation and digital filtering,” *IEEE Trans. Acoustics Speech Signal Processing*, vol. ASSP-26, pp. 508–517, Dec. 1978.

[6] L. J. Wang, W. S. Hsieh, T. K. Truong, “A fast computation of 2D cubic-spline interpolation,” *IEEE Signal Processing Letters*, vol.11, no.9, pp.768-771, Sept. 2004.

[7] M. Unser, A. Aldroubi, and M. Eden, “B-spline signal processing: Part II - Efficient design and applications,” *IEEE Transactions on Signal Processing*, vol. 41, no. 2, pp. 834-848, Feb. 1993.