An Improved Color Video Super-Resolution Using Kernel Regression and Fuzzy Enhancement

Manna Elizabeth Philip and G Santhosh Kumar
Department of Computer Science and Technology
Cochin University of Science and Technology
Cochin, Kerala, India
manna20philip@gmail.com, san@cusat.ac.in

Abstract— **An improved color video super-resolution technique using kernel regression and fuzzy enhancement is presented in this paper. A high resolution frame is computed from a set of low resolution video frames by kernel regression using an adaptive Gaussian kernel. A fuzzy smoothing filter is proposed to enhance the regression output. The proposed technique is a low cost software solution to resolution enhancement of color video in multimedia applications. The performance of the proposed technique is evaluated using several color videos and it is found to be better than other techniques in producing high quality high resolution color videos.**

Keywords- Super-resolution; kernel regression; fuzzy image enhancement

I. INTRODUCTION

High resolution images are essential in many imaging applications like video surveillance, medical imaging, high-definition television, remote sensing, computer vision etc. Improving imaging systems is not a cost effective method to create high resolution images due to hardware cost of underlying optics and sensor resolution. But software approach in which Super-Resolution reconstruction algorithms are used, is a most promising technique that helps to overcome these limitations. Super-resolution (SR) is a process of obtaining a high resolution (HR) image from low quality low resolution (LR) image/images [1].

There are many techniques available for super-resolution reconstruction, which mainly fall under two classes: single image super-resolution and multiple image super-resolution [2-5]. The former methods are mainly based on interpolation and smoothing techniques as no additional information is available and they have the inherent problem of blurring in the reconstruction [2]. Machine learning techniques are applied in single image super-resolution where LR patches and corresponding HR patches from training images are used for generating the high frequency details which do not exist in the LR image [3]. These methods have huge memory requirements for storing the training data and are computationally costly too. In multi image super-resolution, different methods like, Iterative back projection, and kernel regression based techniques are proposed [5]. In Kernel Regression based techniques, each output HR pixel is computed as a weighted sum of neighbouring observed pixels from the nonuniform HR grid generated from LR images having subpixel level shifts with respect to each other. Data dependent kernel technique for super-resolution is presented in [6]. A super-resolution reconstruction for LR video using a zero-order data adaptive kernel estimator is presented in [7], in which only one pixel each from an LR frame is considered for reconstruction. An improved method taking into account the small global motions between LR frames and using all the known neighbouring pixels for regression is presented in [8].

This paper reports an improved super-resolution method for low resolution color videos using kernel regression and fuzzy smoothing. Experimental results show that the proposed method give much better results in terms of PSNR value, Image Quality index, as well as subjective evaluation. Also, it avoids the blurring out of the edge details and retains the color information in the video. The rest of the paper is organized as follows: An introduction to kernel regression is presented in section II. Section III discusses the proposed super-resolution algorithm and the fuzzy smoothing filter. The performance analysis of the proposed method is given in section IV. Concluding remarks are given in section V.

II. KERNEL REGRESSION

Kernel regression is an effective method for both de-noising and interpolation. It is a non parametric technique used to estimate the value of an unknown function at any given point based on observations. The data measurement model in 2-D is given as:

$$Y_i = f(x_i) + \varepsilon_i,\ \ i= 1, .,N,\ and\ \ x_i=[x_{1i},x_{2i}]^T \tag{1}$$

where, $\{x_i,\ i=1,2,....N\}$ are the estimation points, $\{Y_i,\ i=1,2,....N\}$ are the values of the variable Y. In the case of images these variables are pixel positions and pixel vales at those positions. f is a regression function and $\{\varepsilon_i,\ i=1,2,....N\}$ are independant identically distributed random errors, and N is the number of samples(number of frames). The generalized kernel estimate $\tilde{f}(x)$ is given by the following minimization problem[11]

$$\min_{p,q_1,..q_l} \sum_{i=1}^{N} [Y_i - (p + q_1(\mathrm{x_i - x}) + \cdot + q_l(\mathrm{x_i - x})^l)] K\left(\frac{x_i - x}{h}\right) \tag{2}$$

where K(.) is the kernel function with bandwidth *h*, and *l* is a positive integer which is the order of the kernel estimator. The regression coefficients *p, q1, q2,…,ql* are determined by solving the equation. A computationally simple zero order kernel estimator known as Nadaraya-Watson kernel

978-0-7695-4723-7/12 $26.00 © 2012 IEEE
DOI 10.1109/ICACC.2012.25

estimator [8] is obtained by solving the equation (2) for l = 0. Since this kernel is not dependent on the image characteristics, a data dependent kernel which depends on the sample values (pixel values) also is proposed in [6]. The modified adaptive kernel function is given below, which gives better results.

$$f_{NW}(x)=\frac{\sum_{i=1}^{N} Y_i K\left(\frac{x_i-x}{h}\right)K\left(\frac{Y_i-Y}{h_r}\right)}{\sum_{i=1}^{N} K\left(\frac{x_i-x}{h}\right)K\left(\frac{Y_i-Y}{h_r}\right)} \tag{3}$$

where h_r is an intensity dependent smoothing parameter. A Gaussian kernel, which is differentiable and computationally simple, is used for regression in this work.

III. THE PROPOSED METHOD

The proposed technique employs kernel regression, and a fuzzy smoothing filter. This technique can be applied to both video sequences as well as images of the same scene taken with slight camera movement without causing significant displacement of object regions. A buffer is used to save the *N* sample frames. On receiving the *N* frames the regression is applied to get the HR frame. All the available 4-neighbours from each frame are used for regression in this method and it is explained in the following subsection. The HR output resulting after regression will have block artifacts and the edges reconstructed may not be smooth. Usually a lowpass Gaussian filter is applied to smooth out this as in [6]. This will result in loss of edge information and blurring problem. Hence in this proposed method, a fuzzy colour image filter proposed in [10] is applied to obtain the final HR image, which retains the smoothness of edges without block effects and the colour clarity of the image to a good level. The following subsections explain the super-resolution algorithm and the fuzzy filter.

A. Proposed Super Resolution Method

Given LR images/frames of size $m \times n$, the requirement is to obtain an $rm \times rn$ image/frame, where r is the resolution factor. The first frame of the N samples is taken as the reference frame, and required number of empty rows and columns are inserted between rows and columns of the reference image to form the HR skeleton. For making the concept simpler let's assume r = 2. If the image samples are of size 2 × 2 we have to obtain a 4× 4 image; i.e. corresponding to each pixel in the LR image three additional pixels are to be generated in the HR image. Hence 12 additional pixel values are to be computed in each HR frame. Figure 2 shows the 2 × 2 LR image samples and the 4× 4 HR image skeleton; the three additional unknown pixels corresponding to the known pixel $K1_{11}$ in the LR frame are UK_1, UK_2, and UK_3 as shown.

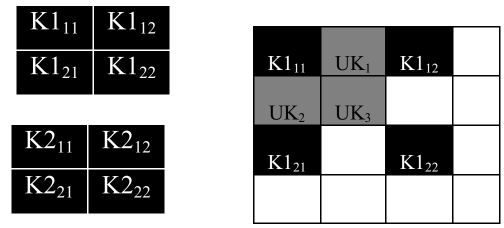


Figure 2: (a) Two 2×2, low resolution images/frames, K1, K2 (b) 4× 4 High resolution image skeleton (r=2), the grey boxes refer to the additional pixels to be computed.

Each unknown pixel value is calculated from the known pixel values in the LR frames. For example, consider the generation of the unknown pixel UK_1, the pixel value at this location depends on the neighbouring pixel values $K1_{11}$ and $K1_{12}$ in the first frame K1; $K2_{11}$ and $K2_{12}$ in the second frame K2; and so on for the remaining LR frames. The pixel value of UK_2 depends on known pixel values $K1_{11}$ and $K1_{21}$ of K1, $K2_{11}$ and $K2_{21}$ of K2 and similarly for the other sample frames. The pixel at UK_3 has four neighbouring known pixel values per frame, so it depends on $K1_{11}$, $K1_{12}$, $K1_{22}$ and $K1_{21}$ of K1, $K2_{11}$, $K2_{12}$, $K2_{22}$ and $K2_{21}$ of K2 and similarly for the remaining LR frames. The known neighbouring pixels in one frame for UK_1, UK_2, and , UK_3 are shown in figure 3(a).

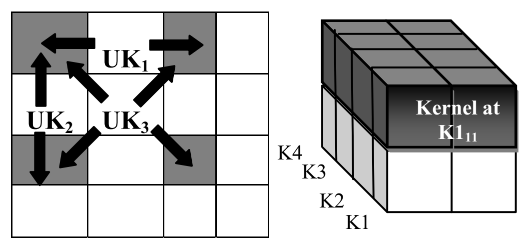


Figure 3: (a) Spatial location of neighbourhood pixels used to estimate unknown pixel values $UK_1(K_{11},K_{12})$, $UK_2(K_{11},K_{21})$, and , $UK_3(K_{11},K_{12},K_{21},K_{22})$. (b) Pixels to be considered from the four LR frames for the estimation of the unknown pixel UK_1

The unknown pixels are computed by placing the kernel at $K1_{11}$ and using all the respective known neighbouring pixels from all the 4 LR frames. Figure 3(b) shows the computation for UK_1. Similarly all other unknown pixels in the HR frame are computed placing the kernel at the other three known pixel positions and using the known neighbouring pixels from the LR frames for regression. To limit the registration error, only four LR frames are used in this work. This regression method is applied to colour images by computing the Red, Green, and Blue components of unknown pixels in the HR frame separately.

B. Fuzzy Filter

In this fuzzy filter [10] a pixel is modified using other pixel values from its neighborhood, but simultaneously take into account the important image structures such as edges and color component distances, which should be retained by the filter. RGB color model is used for this filtering. The difference between this new proposed filter and other vector based approaches for color image filtering is that, instead of calculating the 3-D distances between two color pixels, three 2-D distances (distances between red–green, red–blue, and green–blue components of two neighboring pixels) are used, together with three fuzzy rules to calculate the weights in the averaging window.

The red, green, and blue component at a certain pixel position of a noisy input image N is denoted as $N_{i,j,1}$, $N_{i,j,2}$ and $N_{i,j,3}$, respectively. For each pixel position (i,j) we define the following couples: the couple red and green denoted as $rg_{i,j}=(N_{i,j,1},N_{i,j,2})$, the couple red and blue denoted as $rb_{i,j}=(N_{i,j,1},N_{i,j,3})$, and couple green and blue denoted as $gb_{i,j}=(N_{i,j,2},N_{i,j,3})$,. To filter the current image pixel at position (i,j), we use a window of size (2K+1) × (2K+1) centered at (i,j). To each of the pixels in the window certain weights are assigned, that is, we have $w_{i+k,j+l,1}$, $w_{i+k,j+l,2}$ and $w_{i+k,j+l,3}$ for the red, green and blue component at position (i+k,j+l), respectively, where k,l Є {-K,..,K}. These weights are computed based on the following fuzzy rule:

Fuzzy rule for red component "IF the distance between the couple $rg_{i,j}$ and $rg_{i+k,j+l}$ is ***small*** *AND the distance between the couple $rb_{i,j}$ and $rb_{i+k,j+l}$ is* ***small*** *THEN the weight for the red component, $w_{i+k,j+l,1}$ is* ***large***.*"*

The weights for the green and blue components are computed using similar rules with corresponding color couples. The idea behind these simple fuzzy rules is to assign large weights to the neighbors that have similar colors as the center. For the red-green couple, the distance is computed as: $D(rg_{i,j}, rg_{i+k,j+l})=[(N_{i+k,j+l,1}-N_{i,j,1})^2 +(N_{i+k,j+l,2}-N_{i,j,2})^2]^{1/2}$. The fuzzy set ***small*** whose membership function given below is used to express the degree to which the distance of two couples is small. The membership function ***small*** is defined as:

$$\mu_s(x)=\begin{cases}\left(\dfrac{p-x}{p}\right)^2, & x\le p\\ 0, & x>p\end{cases}$$

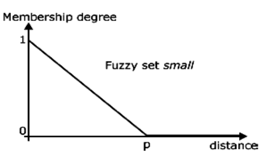


Figure 4: Membership function ***small***

The output of the fuzzy filter for the red component, where the output image denoted as F, is given by:

$$F_{i,j,1} = \frac{\sum_{k=-K}^{+K}\sum_{l=-K}^{+K} w_{i+k,j+l,1}.N_{i+k,j+l,1}}{\sum_{k=-K}^{+K}\sum_{l=-K}^{+K} w_{i+k,j+l,1}} \quad (4)$$

The filtering method for the green and blue component is done analogously to the one above. The final filtered output is obtained after a second filtering applying a correction factor to each pixel in the window as below. The red component is corrected as

$$Out(i,j,1) = \frac{\sum_{k=-L}^{+L}\sum_{l=-L}^{+L} F_{i+k,j+l,1} + \varepsilon_{k,l}}{(2L+1)^2} \quad (5)$$

where, the correction term is given by,

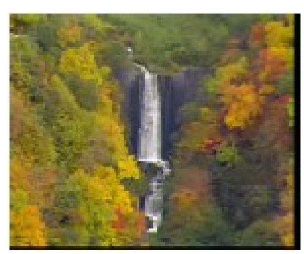
$\varepsilon_{k,l} = 1/3\ (LD_R(k,l)+ LD_G(k,l)+ LD_B(k,l))$

LD_R, LD_G, and LD_B are the local differences or gradients of the red, blue and green components. The gradient for red component is computed as, $LD_R(k,l) = F_{i+k,j+l,1}-F_{i,j,1}$.

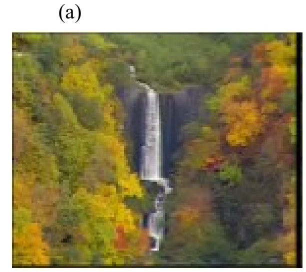
IV. PERFORMANCE ANALYSIS

The proposed method is tested with several LR colour videos. Four consecutive LR frames are used for regression to construct a HR frame with 2×2 scaling factor. To do an objective evaluation of the performance of the proposed method, 4 original consecutive HR video frames are down sampled and then reconstructed to the original size using the proposed method. The performance is measured in comparison with the original HR frame using PSNR (Peak Signal to Noise Ratio) and IQI (Image Quality Index) measures. The IQI [9] measures any distortion as a combination of three different factors; loss of correlation, luminance distortion, and contrast distortion and gives a general idea of how good the image is. Figure 5 shows the four LR frames of **water fall** video and the reconstructed HR images before and after applying the fuzzy filter. The PSNR and IQI values are shown below each reconstructed image and also given in Table 1.

(a)



(b) PSNR:27.7065;IQI:0.9956



(c) PSNR:27.7640 ;IQI: 0.9992

Figure 5: (a) 4 LR frames of **water fall** video (b) HR image with proposed regression method before applying the fuzzy filter (c) after applying the fuzzy filter.

It can be noticed that these performance measures are improved with the fuzzy filter. It can also be noted that the edges are smoother, blurring effect is less and colour clarity is closer to that of the LR frames in the case of the proposed method with fuzzy filter. Figure 6 shows the case of

tempete video and improved performance can be noticed in PSNR and IQI values and as well as in the subjective evaluation with the proposed method.

(a)

(b) PSNR:23.2368; IQI:0.9926 (c) PSNR:23.2676; IQI: 0.9931

Figure 6: (a) 4 LR frames of **tempete** video (b) HR image of the target region with proposed regression method before applying the fuzzy filter (c) after applying the fuzzy filter.

Figure 7 illustrates reconstruction of video **News**. The super-resolution result with fuzzy filter is shown (b). The PSNR value increases on applying the fuzzy filter and the IQI remains same. Figure 7(c) gives the results claimed in [8]. The method discussed in [8] claims to be better than many of the state of the art techniques. It can be seen that the proposed method gives much better PSNR value than the previous method. IQI value is not measured in [8]. It can be seen from the figure 7(d) that, it is blurred with block effect and the colour clarity is also not good even though the global motion effect is also taken care of in the method in [8]. This shows the superiority of the proposed method which is simple as no motion evaluation is needed, but it gives good quality HR videos for practical multimedia applications.

(a)

(b) PSNR:24.8587;IQI:0.9978 (c) PSNR:22.53

Figure 7: (a) 4 LR frames of **News** video(b) HR image with proposed regression method after applying the fuzzy filter.(c) HR image using the method in [8]

TABLE I PERFORMANCE COMPARISON

Video	PSNR Value		IQI Value	
	Without filter	With filter	Without filter	With filter
Water fall	27.7065	**27.7640**	0.9956	**0.9992**
Tempete	23.2368	**23.2676**	0.9926	**0.9931**
News	24.8443	**24.8587**	**0.9982**	0.9978

V. CONCLUSION

A simple kernel regression method for colour video super-resolution and a fuzzy filter for enhancing the regression output is presented in this paper. It is computationally simple as no motion computation is required and is suitable for low cost multimedia applications. The fuzzy filter enhances the output with smooth edges and also retains the colour clarity. Experimental results show that the proposed method performs better than existing super-resolution techniques. The method must be imporved for videos with considerable motions between successive frames.

VI. REFERENCES

[1]. Nguyen, N., Milanfar, P.: “A Computationally Efficient Superresolution Image Reconstruction Algorithm”. IEEE Transactions on Image Processing 10, 573–583 (2001)

[2]. Chang, H., Yeung, D.Y., Xiong, Y.: “Super-resolution through Neighbor Embedding”. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 275–282 (2004)

[3]. Freeman, W.T., Jones, T.R., Pasztor, E.C.: “Example-based Super-resolution”. IEEE Computer Graphics and Applications 22, 56–65 (2002)

[4]. Li, D., Simske, S., Mersereau, R.M.: “Single Image Super-resolution Based on Support Vector Regression”. In: IEEE Proceedings of IJCNN, pp. 2898–2901 (2007)

[5]. Elad, M., Feuer, A.: “Restoration of a Single Superresolution Image from Several Blurred, Noisy, and Undersampled Measured Images”. IEEE Transactions on Image Processing Vol.6, 1646–1658 (1997)

[6]. Takeda, H., Farisu, S., Milanfar, P.: “Kernel Regression for Image Processing and Reconstruction”. IEEE Transactions on Image Processing vol. 16, pp. 349–366 (2007)

[7]. Mohammed M.I., Vijayan K. A., Mohammed N.I., and Mohammed A. K.: “Video Super-Resolution by adptive Kernel Regression”. In Proceedings of ISVC 2009, Part II, LNCS 5876, pp.799-806, 2009.

[8]. Mohammed M.I., Vijayan K. A., Mohammed N.I., and Mohammed A. K.: “Super-Resolution Enhancement Technique for Low Resolution Video”, IEEE Transactions on Consumer Electronics, Vol. 56, No.2, pp. 919-924., 2010.

[9]. Z. Wang and A.C. Bovik: “A Universal Image Quality Index, IEEE Signal Processing Letters”, Vol. XX, No. Y, March 2002

[10]. Schulte, S., et al.: “Fuzzy noise reduction method for color images”. IEEE Transaction on Image Processing 16(5), 1425-1436 (2007).

[11]. Qiu, P.: “Image Processing and Jump Regression Analysis”. Wiley Series in Probability and Statistics. Wiley, New Jersey(2005).