REAL-TIME REGULARIZED ITERATIVE ENHANCEMENT OF LOW-RESOLUTION VIDEO

SUNGJIN KIM[1], JOONKI PAIK[1], JE-WOO KIM[2], JONG-HO PAIK[2], AND HYEOK GOO JUNG[2]

[1]Department of Image Engineering, The Graduate School of Imaging Science
Chung-Ang University, Seoul 156-756, Korea
[2]Korea Electronics Technology Institute, Kyunggi-Do, 451-865, Korea

Abstract: Regularized iterative image restoration is proven to be a successful technique in restoring degraded images. However, its application is limited to still images or off-line video enhancement due to its slow convergence. In order to enable this iterative restoration algorithm enhance the quality of video in real-time, each frame of video is considered as the constant input and the processed previous frame is considered as the previous iterative solution. Each frame of a video is segmented into two regions: still background and moving objects. These two regions are processed differently by using a segmentation-based spatially adaptive restoration and a background generation algorithms. The proposed framework enables real-time video enhancement at the cost of image quality only in the moving object area of dynamic shots, which is relatively insensitive to the human visual system.

Key-Words: real-time image processing, image restoration, interpolation, video enhancement, regularization

1. INTRODUCTION

Among various types of degradation, compression and subsampling are the major factors that degrade the quality of video. Compression is almost always necessary for video communication due to the limited bandwidth of the communication channel. Subsampling is another way to reduce the amount of video data and computational overhead at the cost of resolution. Both compression and subsampling can be mathematically modeled, and therefore their degradation effects can be removed by the regularized iterative restoration algorithm. Mathematical model of degradation due to video compression and the corresponding restoration algorithms have also been proposed in the literature [1][2]. In this section we present the image degradation model due to combined compression and subsampling and the corresponding restoration results using the proposed real-time framework with particular emphasis on the video enhancement aspect.

High-resolution (HR) image restoration refers to restoration of a low-resolution (LR) image by removing degradation due to subsampling. Application areas of HR image restoration include, but are not limited to: digital high-definition television (HDTV), aerial photography, medical imaging, video surveillance, and remote sensing [3].

Many algorithms have been proposed to obtain an HR image from LR images. Conventional interpolation algorithms, such as zero-order or nearest neighbor, bilinear, cubic B-spline, and the DFT-based interpolation, can be classified by basis functions [4][5][6]. Since these algorithms focus on just enlargement of an image without consideration of the degradation factor, they cannot restore the original HR image. In order to improve the performance of the previously mentioned algorithms, a spatially adaptive cubic interpolation method has been proposed in [7]. It is well-known that image interpolation is an ill-posed problem. More specifically, sub-sampling process can be regarded as a general image degradation process. Therefore, the regularized image restoration algorithm can successfully find the inverse solution defined by the subsampling process with *a priori* constraint [8][9]. Many regularization-based or similar interpolation methods have been proposed in the literature [10][11][12][13][14].

2. IMAGE DEGRADATION MODEL FOR COMBINED SUBSAMPLING AND COMPRESSION

Enlargement of compressed video has wide application areas. For example, when we look at a JPEG coded picture by using an image viewer, we can selectively enlarge or reduce the size of the picture by interpolation and subsampling, respectively. Consider an LR image of size $M \times N$, which can be obtained by subsampling the original $pM \times pN$ image x. This LR image is then compressed by block discrete cosine transform (BDCT). The original $pM \times pN$ image can be reconstructed from the subsampled-compressed image by regularized image restoration. For the reconstruction or enhancement we need to define the image degradation model for both subsampling and compression.

Figure 1 shows the combined subsampling-compression degradation process, where H_S represents subsampling, C and C^{-1} respectively represents forward and the inverse DCT, and Q and D^{-1} respectively represents the quantization and inverse quantization matrices [1].

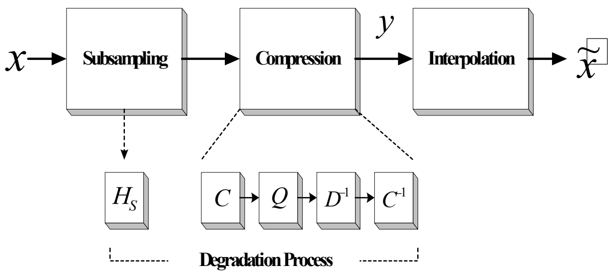

Figure 1: Image degradation process for combined subsampling-compression

Based on the degradation process shown in Figure 1, the corresponding mathematical model is given as

$$H = H_C H_S\,, \tag{1}$$

where $H_C = C^{-1}D^{-1}QC$ represents the degradation matrix due to compression. x, H_S and H_C respectively represent the lexicographically ordered, $p^2MN \times 1$, $MN \times p^2MN$, and $MN \times MN$ vectors. The image degradation model for combined subsampling and compression is given as

$$y = Hx\,, \tag{2}$$

where y represents the degraded image.

3. REAL-TIME RESTORATION OF SUBSAMPLED-COMPRESSED VIDEO

Given an image degradation model such as (2), the general image restoration process based on the regularized constrained optimization approach is to find the minimum solution of an objective function given as,

$$f(x) = \|y - Hx\|^2 + \lambda\|Cx\|^2\,, \tag{3}$$

where C and λ respectively represent a high pass filter for incorporating *a priori* smoothness constraint and the regularization parameter that controls the fidelity to the observed data and smoothness of the restored image. To find the solution for minimizing the objective function given in (3), we apply the following iterative method,

$$x^{(k+1)} = x^{(k)} + \beta\left[H^T y - \left(H^T H + \lambda C^T C\right)x^{(k)}\right], \tag{4}$$

where $x^{(k+1)}$ and β respectively represent the restored image after $k+1$ st iterative solution and the step length which controls the convergence rate.

Equation (4) can be rewritten as

$$x^{(k+1)} = \beta H^T y + \left[I - \left(H^T H + \lambda C^T C\right)\right]x^{(k)}\,. \tag{5}$$

Since H and C respectively represents the degradation and *a priori* smoothness constraint matrices, both of them are given before the iteration starts. Hence the k th iteration in (5) can be considered as a single filtering operation. Most hardware-based image filtering operations are performed by visiting every pixel in the raster scanning order, which enable real-time video processing.

If we use the original regularized restoration algorithm, which converges after M iterations, for enhancing a video, each frame delays by M frames. If the video consists of N frames, total delay becomes MN frames, which disables real-time processing.

The restored i th frame, denoted by $\hat{x}_i$, for $i = 1, \ldots, N$, is obtained as

$$\begin{aligned} x_i^{(0)} &= H^T y_i \\ x_i^{(k+1)} &= x_i^{(k)} + \beta\left[H^T y_i - \left(H^T H + \lambda C^T C\right)x_i^{(k)}\right], \end{aligned}$$

$$\text{for } k = 0, 1, \ldots, M-1, \text{ and } \hat{x}_i = x_i^{(M)}\,. \tag{6}$$

However, if a shot of the input video sequence has little motion, we can assume that N input frames in the motionless shot are approximately the same, such as

$$y_1 \approx y_2 \approx \cdots \approx y_N\,. \tag{7}$$

Based on (7), the first frame is used as the initial guess of the regularized iteration as

$$x_1^{(0)} = H^T y_1 \text{, and } \hat{x}_1 = x_1^{(0)}, \tag{8}$$

and the i th frame, for $i = 2,\ldots,N$, is updated as,

$$\begin{aligned} x_1^{(1)} &= x_1^{(0)} + \beta\left[H^T y_1 - \left(H^T H + \lambda C^T C\right)x_1^{(0)}\right] \\ \hat{x}_2 &= x_1^{(1)} \\ &\vdots \\ x_1^{(N-1)} &= x_1^{(N-2)} + \beta\left[H^T y_1 - \left(H^T H + \lambda C^T C\right)x_1^{(N-2)}\right] \\ \hat{x}_1 &= x_1^{(i-1)} \end{aligned}. \tag{9}$$

The proposed iteration scheme for a motionless shot is illustrated in

Figure 2.

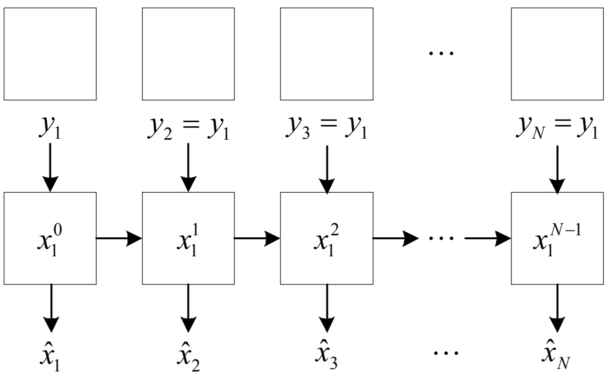

Figure 2: The real-time iterative restoration scheme for a motionless shot

In general, we can hardly find exactly the same frames in a video, which means the frame-level decomposition is almost impractical. For the block and pixel-level decompositions, a necessary condition for the proposed real-time framework is that the video must be captured by a stationary camera. This condition ensures that the stationary background of each frame is the same. In the video captured by a fixed camera the background region can be enhanced by the proposed regularized iteration and the moving region is either enhanced by simple filtering or remains unprocessed. The proposed region-adaptive real-time restoration consists of two steps. First, we extract the background using the method described in the previous section, and then the stationary background is enhanced by using the proposed restoration method. Second, the restored background and the moving regions are merged.

Since the extracted background from a fixed camera is the same, we have

$$y_{b_1} = y_{b_2} = \ldots = y_{b_N}, \tag{10}$$

where y_{b_N} represents the N th reconstructed background. Since (10) is equivalent to (7), (8) and (9) can be respectively rewritten as

$$\begin{aligned} x_{b_1}^{(0)} &= H^T y_{b_1} \\ \hat{x}_{b_1} &= x_{b_1}^{(0)} \end{aligned}, \tag{11}$$

and

$$\begin{aligned} x_{b_1}^{(1)} &= x_{b_1}^{(0)} + \beta\left[H^T y_{b_1} - \left(H^T H + \lambda C^T C\right)x_{b_1}^{(0)}\right] \\ \hat{x}_{b_2} &= x_{b_1}^{(1)} \\ &\vdots \\ x_{b_1}^{(N-1)} &= x_{b_1}^{(N-2)} + \beta\left[H^T y_{b_1} - \left(H^T H + \lambda C^T C\right)x_{b_1}^{(N-2)}\right] \\ \hat{x}_{b_N} &= x_{b_1}^{(N-1)} \end{aligned}. \tag{12}$$

The proposed algorithm is summarized as follows.

Algorithm 1: Decomposition-based real-time regularized restoration

(step 1) Generate the background image using the first T input frames, as described in Sec. 3. After T frames pass, go to the next step.
(step 2) Extract moving regions at each frame by comparing with the previously generated background image.
(step 3) Apply the real-time regularized restoration procedure given in (11) and (12) to the background and a simple filter to the moving region.
(step 4) Combine the restored background and the filtered moving region. □

4. EXPERIMENTAL RESULTS

In order to test the proposed real-time restoration algorithm, we used a video sequence of size 640×480, captured by a fixed-view surveillance camera. One frame of the video sequence is shown in Figure 3. This video sequence is appropriate for the experiment because the camera was fixed and the localized target object can easily be extracted from the stationary background. The four times subsampled version of the original video was used as an input to the proposed real-time restoration algorithm.

For objective comparison of two different degradation model, we computed peak-to-peak signal-to-noise ratio (PSNR) as

$$PSNR[dB] = 10\log_{10}\frac{L^2 \times 255^2}{\|x-\hat{x}\|^2}, \qquad (13)$$

Based on Algorithm 1, the background image was generated first. The background image consists of four evenly spaced vertical regions. Both the first and the second regions were obtained from the third frame. The third and the fourth regions were obtained from the 17th and the 35th frames, respectively. The background generation process is illustrated in Figure 4. In other words, the background image was completely generated after the 35th frame, and then the rest part of Algorithm 1 can be performed.

Figure 3: One frame of the original video sequence captured by a fixed-view surveillance camera. Note that the stationary background is dominant throughout the image, and the moving person is localized.

Figure 4: The background generation process. Background images with: (a) the first region from the third frame, (b) the first and second regions both from the third frame, (c) the first, second, and third regions (the third region is obtained from the 17th frame,) and (d) all four regions (the fourth region from the 35th frame.)

After the LR background image is generated, it must be enlarged to obtain the original HR background image. Two differently enlarged background images are shown in Figure 5 and Figure 6. Figure 5 shows the four times enlarged image using simple zero-order interpolation algorithm. Due to the nature of the interpolation algorithm, the enlarged image shows blocking artifacts and does not have enhanced resolution. On the other hand Figure 6 shows the 50th background frame, which is enhanced by using the proposed real-time regularized restoration algorithm. This background image is obtained after 15 iterations, and shows 0.49dB improvement over Figure 5. If we keep iterating the background image shown in Figure 6, both the subjective quality and the PSNR value also keep increasing.

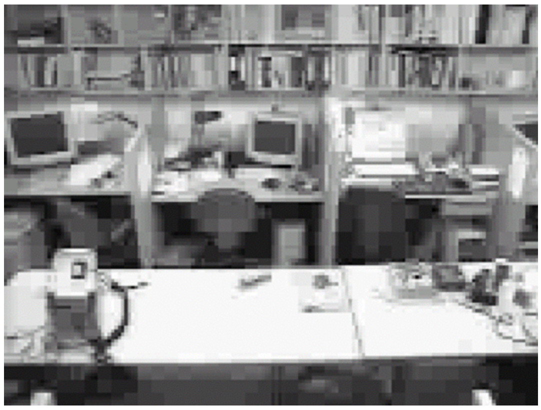

Figure 5: The enlarged 50th background frame by using the zero-order interpolation algorithm (PSNR=20.45dB)

Figure 6: The enlarged 50th background frame by using the proposed real-time regularized restoration algorithm. (PSNR=20.94dB)

After generating and restoring the background image, the moving region is to be extracted from the input frame. The extracted moving region from the 88th frame is shown in Figure 8, and the combined restored background and filtered moving region is shown in Figure 9. If we compare Figure 9 with the enlarged image without restoration shown in Figure 7, the restored image gives approximately 0.5dB improvement in PSNR. The proposed real-time video enhancement algorithm is compared with simple enlargement method in the sense of PSNR versus the number of iterations, as shown in Figure 10. Until the 35th frame no restoration process occurs, and therefore two methods give the same result. After the background image is generated at the 35th frame, the real-time restoration result outperforms simple enlargement method.

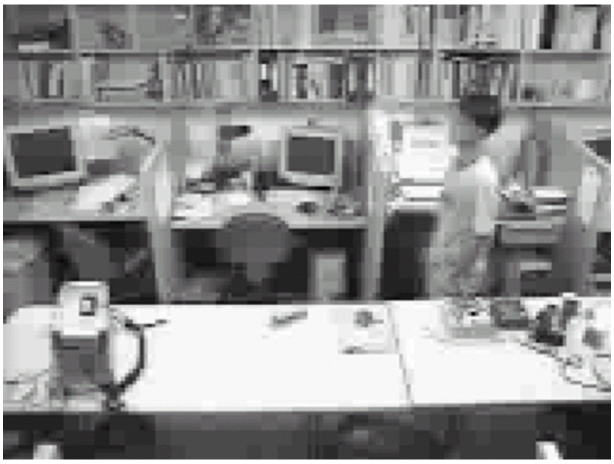

Figure 7: The enlarged 88th frame by using the zero-order interpolation algorithm. (PSNR=20.57dB)

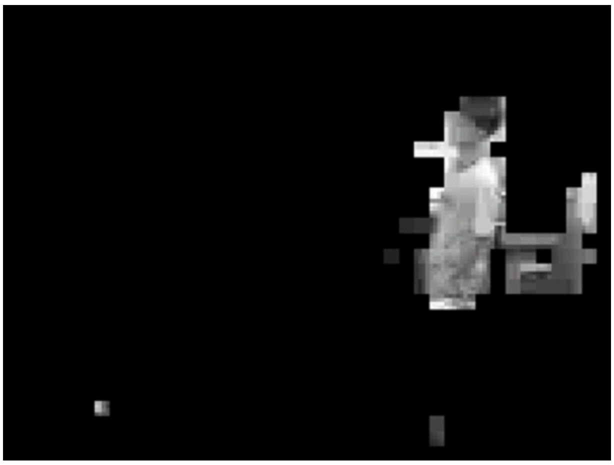

Figure 8: The moving region extracted from the 88th frame

Figure 9: The combined restored background and filtered moving region. (PSNR=21.01dB)

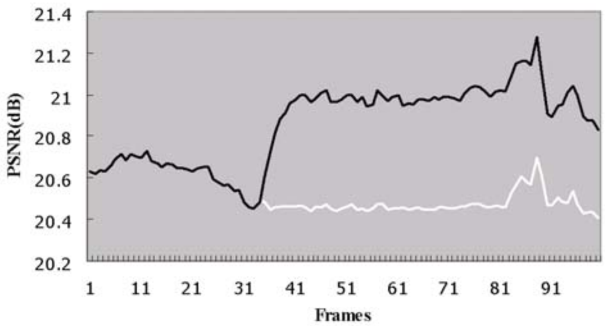

Figure 10: Comparison of two different enlarged sequences in the sense of PSNR. The black curve represents the enhanced sequence by using the proposed real-time restoration, and the white curve represents just enlarged sequence without restoration. Until the 35th frame two sequences are the same because no restoration process occurs.

To ensure the real-time processing, we computed the processing time of the proposed algorithm. The 100 frame 320×240 video sequence was processed by using IDL (Interactive Data Language) in a Pentium III 500MHz personal computer. The proposed real-time restoration algorithm took 182 seconds to process 100 frames, while the original form of iterative restoration algorithm took 3860 seconds.

5. CONCLUSIONS

We proposed a real-time framework for regularized iteration and its application to restoration of subsampled-compressed degradation. In order to restore a subsampled-compressed image, the combined image degradation model was proposed. The proposed real-time framework first generates the background image, extracts the moving region by subtracting the background image from the current frame, and finally combines the restored background and the filtered moving region. The proposed restoration framework can efficiently enhance both quality and resolution of video in real-time at the cost of slight performance degradation especially in the moving region, which is relatively insensitive to the human visual system.

Extensive experiment was conducted to demonstrate the feasibility of the proposed image degradation model and the real-time restoration framework. Experimental results showed the combined degradation model for subsampling-compression outperformed the degradation model for only subsampling in restoring the compressed video. Each step of the proposed real-time restoration algorithm was tested and the corresponding results were presented.

Application of the proposed real-time framework is restricted to surveillance or the equivalent quality video systems because of the assumption that the input video must be captured by a fixed-view camera. However, if the background image generation method is modified to deal with panned or tilted video input, the proposed algorithm can be used to extended application areas.

ACKNOWLEDGMENT

This research was supported in part by Korea Electronics Technology Institute and in part by Brain Korea 21 project.

11. REFERENCES

[1] T. K. Kim and J. K. Paik, "Fast image restoration for reducing block artifacts based on adaptive constrained optimization," *Journal of Visual Communication and Image Representation*, vol. 9, no. 3, pp. 234-242, September 1998.

[2] T. K. Kim, J. K. Paik, C. S. Won, Y. S. Choe, J. Jeong, and J. Y. Nam, "Blocking effect reduction of compressed images using classification-based constrained optimization," *Signal Processing: Image Communication*, vol. 5, pp. 869-877, 2000.

[3] R. A. Schowengerdt, *Remote Sensing: Models and Methods for Image Processing*, 2nd ed., Academic Press, 1997.

[4] J. S. Lim, *Two-Dimensional Signal and Image Processing*, Prentice-Hall, 1990.

[5] M.Unser, "Fast b-spline transforms for continuous image representation and interpolation," *IEEE Trans. Pattern Analysis, Machine Intelligence,* vol. 13, pp. 277-285, March 1991.

[6] J. A. Parker, R. V. Kenyon, and D. E. Troxel, "Comparison of interpolating methods for image resampling," *IEEE Trans. Medical Imaging* vol. 2, pp. 31-39, March 1983.

[7] K. P. Hong, J. K. Paik, H. J. Kim, and C. H. Lee, "An edge-preserving image interpolation system for a digital camcoder," *IEEE Trans. Consumer Electronics*, vol. 42, pp. 279-284, August 1996.

[8] M. C. Hong, M. G. Kang, and A. K. Katsaggelos, "An iterative weighted regularized algorithm for improving the resolution of video sequences," *Proc. 1997 Int. Conf. Image Processing*, vol. 2, pp. 474-477, October 1997.

[9] B. C. Tom and A. K. Katsaggelos, "An iterative algorithm for improving the resolution of video sequences," *Proc, SPIE Visual Comm., Image Proc.*, pp. 1430-1438, March 1996.

[10] R. R. Schultz and R. L. Stevenson, "Extraction of high-resolution frames form video sequences," *IEEE Trans. Image Processing*, vol. 5, pp. 996-1011, June 1996.

[11] J. H. Shin, J. H. Jung, and J. K. Paik, "Spatial interpolation of image sequences using truncated projections onto convex sets," *IEICE Trans. Fundamentals of Electronics, Communications, Computer Sciences*, June 1999.

[12] R. C. Hardie, K. J. Barnard, and E. E. Armstrong, "Joint map registration and high-resolution image estimation using a sequence of undersampled images," *IEEE Trans. Image Processing*, vol. 6, pp. 1621-1633, December 1997.

[13] R. C. Hardie, K. J. Barnard, J. G. Bognar, E. E. Armstrong, and E. A. Watson, "High-resolution image reconstruction from a sequence of rotate and translated frames and its application to an infrared imaging system," *Optical Engineering*, vol. 37, pp. 247-260, January 1998.

[14] A. J. Patti, M. I. Sezan, and A. M. Tekalp, "High-resolution standards conversion of low resolution video," *Proc.1995 Int. Conf. Acoust., Speech, Signal Processing*, pp. 2197-2200, 1995.

[15] J. K. Paik and Y. C. Park, "An edge detection approach to digital image stabilization based on tri-state adaptive linear neurons," *IEEE Trans. Consumer Electronics*, vol. 37, no. 3, pp. 521-530, August 1991.

[16] J. K. Paik, Y. C. Park, and D. W. Kim, "An adaptive motion decision system for digital image stabilizer based on edge pattern matching," *IEEE Trans. Consumer Electronics*, vol. 38, no. 3, pp. 607-616, August 1992.

[17] A. M. Tekalp, *Digital Video Processing*, Prentice-Hall, 1995.