Pre- and Post-Processing Algorithms for Compressed Video Enhancement

C. Andrew Segall and Aggelos K. Katsaggelos
{asegall, aggk}@ece.northwestern.edu
Department of Electrical and Computer Engineering
2145 N. Sheridan Road
Evanston, Illinois 60208-3118, USA.

Abstract

Standards-based video compression algorithms are rapidly becoming the preferred method for transmitting image sequences. Prominent examples include the MPEG and ITU family of standards. However, it is important to realize that these standards are not bit-exact, in that only the operation of the decoder is defined by the specification. Development of the rate-control mechanism and pre- and post-processing procedures is completely controlled by the system designer, and these components can introduce discernible differences between two standards compliant realizations. In this paper, we survey the fields of pre- and post-processing techniques for video compression. We then discuss our current work on compression enhancement algorithms. These algorithms are applicable to any compression standard but are discussed within the context of MPEG-2.

1. Introduction

Digital video compression algorithms are a fundamental component of a video communication system. The goal of these algorithms is to represent a video sequence with a reduced number of bits, as compared to simply utilizing the intensity values of each pixel in the sequence. If the number of bits is reduced sufficiently, then transmitting and storing the video sequence becomes feasible with modern communication channels. These medias include Internet and Intranet delivery, wireless and satellite communications systems as well as DVD or hard disk storage devices.

Compressed video sequences are usually imperfect descriptions of the original imagery, in that some of the visual data of the original sequence is discarded while encoding. Eliminating information reduces the number of bits required for the compressed representation. Unfortunately, it can also lead to visible errors in the decoded imagery. In this context, the overall visual quality of the decoded sequence then becomes a critical factor in the design and comparison of video compression systems. Many variables affect this quality. For example, rate-control algorithms allocate available bits to different parts of an image sequence, which can introduce a visible difference between similar compression algorithms. In the same way, the design and implementation of pre- and post-processing mechanisms often differentiates one product from another.

In this paper, we survey the field of pre- and post-processing algorithms for digital video compression and then discuss our current work in the area. For both types of algorithms, the goal of the operation is to reduce the severity of coding artifacts within the reconstructed video sequence. Pre-processing algorithms realize this goal by filtering the original images before (or during) compression. This improves performance by simplifying the image content that must be encoded. Post-processing algorithms operate after a sequence is compressed and do not have knowledge of the original image data. Instead, these algorithms identify and attenuate coding artifacts that were introduced by the encoder.

2. Background

Pre- and post-processing algorithms are usually designed within the framework of a successful video standard. At the current time, these standards include the ITU and MPEG compression mechanisms. In these methods, each frame of a video sequence is coded with a combination of a block-wise transformation with the Discrete Cosine Transform (DCT) and motion compensation. Compression is realized by quantizing the transform coefficients. Unfortunately, this quantization can introduce visible errors in the reconstructed video sequence. These artifacts are typically classified as blocking, ringing or temporal flicker.

The independent processing of each block results in the first class of coding artifacts. These blocking errors dominate low bit-rate applications, where the encoder typically removes all high-frequency information. The result is a series of piece-wise constant image blocks, with sharp transitions at the block boundaries. Increasing the bit-rate of the compression system helps in reducing the blocking artifacts. However, higher bit-rate applications are still plagued by the visible "grid-like" structure in the decompressed data.

This work was supported in part by a grant from the Motorola Center for Communications.

Higher bit-rate applications also suffer from ringing artifacts. At these bit-rates, more mid-frequency information is transmitted to the decoder. However, high-frequency information is still removed from the compressed representation. This introduces a strong oscillation, or ringing, in the vicinity of significant image discontinuities. Ringing becomes more complicated in dynamic image sequences, as the oscillation may change from frame to frame. When this occurs, it is sometimes referred to as a “mosquito” or “corona” artifact.

The final artifact appearing with current compression standards is temporal flicker. Temporal flicker is introduced by local decisions within a rate-control algorithm. These algorithms allocate available bits in an attempt to maximize the resulting image quality. Unfortunately, computational complexity and finite resources often restrict the measurement of quality to the current frame of interest. Thus, there is no attempt to maintain temporal fidelity across the entire image sequence, which results in temporally varying image quality.

3. Post-processing

Previous work in the area of post-processing is easily divided into two types of algorithms, image enhancement and image recovery techniques. The first class of algorithms defines a heuristic procedure (or set of filters) to process the compressed imagery. The second class of algorithm relies on a rigorous model for the original image sequence and compression mechanism, utilizing optimization techniques to find the best solution. In this section, we summarize the various methods available for post-processing. Then, we discuss our current work within the area. This work focuses on higher-rate compression systems. However, our methods are applicable to any bit-rate scenario.

3.1. Enhancement Approaches

Designing an image enhancement algorithm typically relies on a two-step procedure. First, objectionable distortions are identified and modeled within the decompressed video sequence. Then, an operator is constructed to attenuate the distortion. For low-bit applications, enhancement techniques primarily remove blocking artifacts. For example, filtering the decoded image with a low-pass filter attenuates blocking [14]. Of coarse, excessive smoothing also removes important high-frequency content. Several techniques adjust the amount of smoothing to address this problem. For example, filtering may be restricted to the block boundaries and reduced when significant intensity differences are detected between neighboring blocks [4]. This preserves significant region boundaries. Alternatively, the visibility of each block boundary may be estimated through the use of a human visual system model. The boundary is then smoothed until it is no longer visible [1].

The design of an enhancement algorithm changes as the bit-rate of the compression system increases. At higher bit-rates, ringing artifacts become a significant distortion. Removing this artifact also requires low-pass filtering. However, the filter must contain a parameter of directionality, so that it filters parallel to edges. This removes the oscillations of the artifact without eliminating the underlying edge information. In practice this is a difficult task, as it requires the estimation and detection of every edge and its orientation within the decoded image. Improved results are attainable though. For example, morphological operators are utilized in [9], while gradient operators appear in [5, 16].

Modern video compression systems are typically prone to both ringing and blocking artifacts, as the rate-control mechanism may encode the image with spatially varying quality. In these situations, an artifact identification procedure must preclude any filtering. One approach to this problem is presented in [12]. In the approach, each block in the decompressed data is classified as “ringing”, “blocking” or “none”. This identification is accomplished by looking at the structure of the DCT coefficients. Significant mid-frequency information suggests that ringing artifacts are present, while the total absence of mid-frequency and high-frequency data suggests that blocking artifacts may appear. Each of these two classes of blocks is then processed with the appropriate de-blocking or de-ringing filter, while the rest of the image is unfiltered.

3.2. Recovery Methods

As an alternative approach to improving the visual quality of compressed video, post-processing algorithms can be expressed as recovery problems. In this context, an enhanced image must satisfy some measurement of “goodness”. This metric attempts to quantify traits of the original image sequence. For example, most image sequences do not contain the blocking and ringing artifacts that appear after compression. Also, the original imagery rarely exhibits a temporal flicker.

Expressing these traits can be accomplished with either deterministic or stochastic models [19]. In the first method, an algorithm reduces compression artifacts by finding a solution that remains faithful to the decoded result while also satisfying a definition of smoothness. These definitions can include linear models within a Lagrangian approach [15, 21] or non-linear models within a projection onto convex sets (POCS) methodology [22].

In the Lagrangian techniques, one attempts to minimize the function

$$J(f) = \left\| \mathbf{f} - \mathbf{g} \right\|^2 + \lambda \left\| \mathbf{C}\mathbf{f} \right\|^2, \qquad (1)$$

where **f** denotes the post-processed image, stored row by row in a one-dimensional vector, **g** denotes the observation provided by the compressed bit-stream, **C** is an operator (or filter) that defines the measurement of goodness, and λ is a multiplier that controls the influence of the regularizing constraint. In most realizations, **C** is a high-pass filter that enforces a smoothness constraint at the block boundaries.

In some applications, it may be more convenient to express the ideal properties of the post-processed image as a combination of closed and convex sets of solutions. For example, instead of penalizing high-frequency content, we could say that the energy across the block boundaries is less than some threshold. If all of the ideal properties describe the original image sequence, a solution is guaranteed to exist within the union of the set of constraints. To find the solution, a projection operator is first defined for each set. These operators map solutions outside of the constraint into the allowable set of solutions. Then, the projection operators are applied with the iteration

$$\mathbf{f}^{\mathbf{k+1}} = \mathrm{P}_1 \mathrm{P}_2 \cdots \mathrm{P}_{n-1} \mathrm{P}_n \, \mathbf{f}^{\mathbf{k}}, \qquad (2)$$

where P_i is the projection operator for the i^{th} set. The algorithm continues until $\mathbf{f}^{\,k+1}$ and $\mathbf{f}^{\,k}$ are very close, which denotes that the solution is within the intersection of the sets.

While the Lagrangian and POCS approaches provide a method for incorporating deterministic models into the recovery procedure, stochastic definitions of image “goodness” are also allowable [7, 8, 10]. In these methods, the post-processed solution corresponds to the maximum *a posteriori* (MAP) estimate of the image sequence provided to the encoder. Thus, after applying Bayes’ rule, the post-processed image must satisfy

$$\mathbf{f} = \arg \min_{\mathbf{f}_{Allowable}} \log p\left(\mathbf{f}_{\mathbf{Allowable}}\right), \qquad (3)$$

where $\mathbf{f}_{Allowable}$ denotes the set of images represented by the compressed result and p is a probability distribution. In most applications, the distributions used for post-processing penalize images with significant high-frequency information, such as the expression

$$p(\mathbf{f}) = \frac{1}{Z} \exp\left\{ -\frac{1}{\beta} \sum_{c} \varphi\left(\mathrm{d}_{\mathrm{c}}(\mathbf{f})\right) \right\}, \qquad (4)$$

where Z is a normalizing constant, β is a non-negative number, d_c returns the simple difference between the pixel of interest and the neighbor in the c^{th} direction and φ is a potential function that returns large numbers for large differences.

3.3. Current Work

In the majority of the previous work, the blocking artifact is assumed to be the dominant distortion. Our work attempts to extend the methods of image recovery to higher bit-rate applications, specifically MPEG-2 encoders operating at broadcast quality data rates. In these applications, ringing artifacts become quite important. Also, temporal flickering increases.

Working within the framework of (1), we pose the post-processing algorithm as the minimization of

$$\begin{aligned} J(f) = \mathrm{P}_{\mathrm{DCT}}\Big(&\|\mathbf{f}-\mathbf{g}\|^2 + \lambda_1\|\mathbf{C_1 f}\|^2 + \lambda_2\|\mathbf{C_2 f}\|^2 \\ &+ \lambda_3\|\mathbf{C_3 f}\|^2 + \lambda_4\|\mathbf{f}-\mathbf{f_{MC}}\|^2 \Big), \end{aligned} \qquad (5)$$

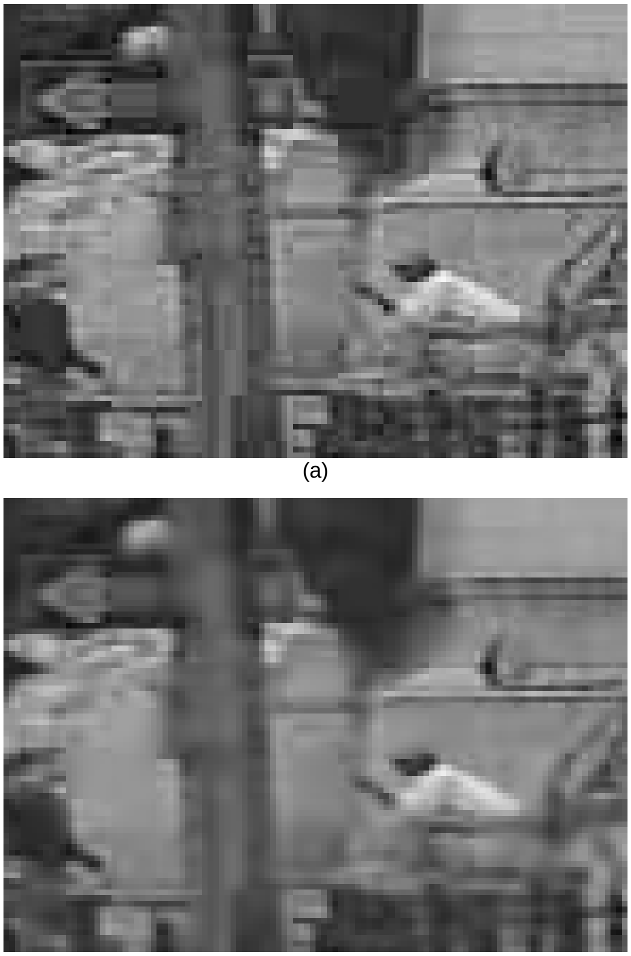
where $\mathbf{f}$ denotes the post-processed image, $\mathbf{g}$ denotes the observation provided by the compressed bit-stream, $\mathbf{C_1}$ and $\mathbf{C_2}$ are operators (or filters) that enforce smoothness across the horizontal and vertical block boundaries, respectively, $\mathbf{C_3}$ is an operator that enforces smoothness within a block, $\mathbf{f_{MC}}$ is the observation provided by the motion compensated prediction, $\mathrm{P}_{\mathrm{DCT}}$ is a projection operator that restricts the solution to the transmitted set of DCT coefficients, and λ_1, λ_2, λ_3 and λ_4 express the relative importance of each constraint.

The method of successive approximations provides a solution to (5) and results in the iteration

$$\begin{aligned} \mathbf{f}^{\mathbf{k+1}} = \mathrm{P}_{\mathrm{DCT}}\Big(\mathbf{f}^{\mathbf{k}} - \alpha\Big\{ &\mathbf{f}^{\mathbf{k}} - \mathbf{g} + \lambda_1 \mathbf{C_1^T C_1 f^k} \\ &+ \lambda_2 \mathbf{C_2^T C_2 f^k} + \lambda_3 \mathbf{C_3^T C_3 f^k} + \lambda_4\left(\mathbf{f^k} - \mathbf{f_{MC}}\right)\Big\}\Big), \end{aligned} \qquad (6)$$

where α determines the convergence and rate of convergence of the algorithm, and $\mathbf{f}^{\,\mathbf{k}}$ and $\mathbf{f}^{\,\mathbf{k+1}}$ denote the post-processed solution at iteration k and k+1, respectively.

Close inspection of (5) reveals that each of the major compression artifacts is addressed in the method. For example, increasing the parameters λ_1 and λ_2 reduces blocking artifacts. Changing the value of λ_3 removes ringing artifacts, and raising the value of λ_4 diminishes the amount of temporal flicker. Selecting values for these parameters is



(a)

(b)

Figure 1. Example of post-processing algorithm: (a) portion of original CCIR601 mage coded with MPEG-2 at 3.5Mbps; (b) post-processed result by minimizing (5).

critical for the success of the algorithm. Ideally, these parameters should vary relative to the compressed bit-stream, reflecting changes in the coding mode and bit-rate assigned to each block. Techniques that incorporate the bit-stream into the parameter selection procedure are discussed in [17, 20]. An example of the method is shown in Figure 1.

4. Pre-Processing

Pre-processing is commonly utilized in video compression systems, though it is not widely discussed in the literature. The goal of a pre-processing algorithm is to remove noise and small features from a video sequence, while preserving salient information. This affects the visual quality of the decoded sequence in two ways. First, the noise component of the original imagery is not transmitted to the decoder. Second, bits previously allocated to the noise are reassigned to more critical image areas.

4.1. Previous Work

Pre-processing algorithms appear at different locations within a video compression system. Perhaps the most obvious is to position the pre-processing algorithm before the encoding procedure. In this location, the pre-processor operates directly on the intensity data of the original image sequence. For example, applying a low-pass filter to the original imagery usually improves the quality of the encoded representation, as it removes additive noise. Heuristic methods are often employed to design the filter. However, a more sound technique is to pose the pre-processing algorithm as an operational rate-distortion problem [6]. Design of the filter then attempts to maximize the resulting image quality, given a pre-defined bit-rate.

An alternative method for pre-processing moves the procedure inside the encoder. In this approach, the motion compensated prediction is modified. Thus, the values for the transmitted error residual are changed, which requires that both the encoder and decoder implement the pre-processing method. For standards compatible encoding, only two compression formats support this type of pre-filtering. In H.261, the prediction is processed with a low-pass filter before calculating the error residual [2]. The operation is called *loop filtering*. In H.263+, the decoder filters the block boundaries of the intensity data [3], with the amount of smoothing varying relative to the quantization parameter.

Pre-processing techniques also appear after the quantization procedure. In these methods, filtering is no longer preformed on intensity data. Instead, the quantized DCT coefficients are directly modified [13]. The goal of the operation is still to remove insignificant features from the compressed bit-stream. However, the definition for insignificance exploits the structure of the variable length codes that are utilized by the encoder. Small changes in the DCT coefficient may introduce significant savings in the bits required for the representation. If the visual impact of these changes is negligible, then the pre-processing operation should adjust the quantized coefficients.

4.2. Current Work

Modifying the intensities within a video sequence before compression provides a very intuitive approach to the pre-filtering problem. Unfortunately, current video compression standards utilize a myriad of predictive coding techniques, which complicates the choice of the smoothing kernel. Under-smoothing the original video sequence may result in excessive coding artifacts at a given bit-rate. Alternatively, over-smoothing the video data may remove information that could be represented at the current bit-rate.

In an ideal application, selection of the pre-filter would be an iterative procedure, with the minimum amount of smoothing found so that coding artifacts are removed. Unfortunately, practical applications rarely have the luxury of the iterative approach. Changing the amount of smoothing in the original video data requires the complete re-encoding of the current frame. For inter-frame coding techniques, this necessitates the recalculation of the motion vectors – a requirement that precludes time-sensitive applications.

In our work, we are exploring pre-filtering procedures that operate on the residual between the predicted and current frames. This allows us to delay the smoothing decision and not effect the calculation of the motion vectors. Additionally, adjusting the smoothing parameter to minimize coding artifacts becomes feasible even in time-sensitive applications. Adapting the response of the filter to the underlying signal is an important component of the operation. A method is presented for adaptive processing in [11]. Higher bit-rate applications and non-linear filters are explored in [18].

Even a simple pre-processing approach can remove both additive noise and quantization errors. Figure 2 illustrates the improvement. In the illustration, an image is corrupted by additive, Gaussian white noise (SNR=20dB). The noisy image is then encoded with a traditional MPEG-2 compression algorithm. The result is plagued by a number of spurious DCT coefficients. Re-encoding the noisy image with a three-by-three averaging operation as the pre-filter improves the result. As can be seen in the figure, the spurious DCT coefficients are removed.

5. References

[1] F.X. Coudoux, M.G. Gazalet, P. Corlay and J.-M. Rouvean, “A Perceptual Approach to the Reduction of Blocking Effect in DCT-Coded Images,” *Journal of Visual Communications and Image Representation*, vol.8, no.4, pp.327-337, Dec. 1997.

[2] ITU-T Recommendation H.261, *Video Codec for Audio Visual Services at px64 kbits/s*, March 1993.

[3] ITU-T Recommendation H.263, *Video Coding for Low Bitrate Communications*, February 1998.

[4] A. Jacquin, H. Okada and P. Crouch, “Content-Adaptive Postfiltering for Very Low Bit Rate Video,” *Proceedings of the IEEE Conference on Data Compression*, pp.111-120, Snow Bird, Utah, March 25-27, 1997.

[5] C.J. Kuo and R.J. Hsich, "Adaptive Postprocessor for Block Encoded Images," *IEEE Transactons on Circuits and Systems for Video Technology*, vol.5, no.4, pp.322-336, Aug.1995.

[6] L.-J. Lin and A. Ortega, "Perceptually Based Video Rate Control Using Pre-filtering and Predicted Rate-Distortion Characteristics", *Proceedings of the IEEE International Conference on Image Processing*, pp.57-60, Santa Barbara, CA, Oct. 26-29, 1997.

[7] J. Mateos, A.K. Katsaggelos and R. Molina, "A Bayesian Approach for the Estimation and Transmission of Regularization Parameters for Reducing Blocking Artifacts," *IEEE Transactions on Image Processing*, vol.9, no.7, pp.1200-1215, July 2000.

[8] T.P. O'Rourk and R.L. Stevenson, "Improved Image Decompression for Reduced Transform Coding Artifacts," *IEEE Transactions on Circuits and Systems for Video Technology*, vol.5, no.6, pp.490-499, Dec. 1995.

[9] S.H. Oguz, Y.H. Hu and T.Q. Nguyen, "Image Coding Ringing Artifact Reduction using Morphological Post-filtering," *Proceedings of the IEEE Second Workshop on Multimedia Signal Processing*, pp.628-633, Redondo Beach, CA, December 7-9, 1998.

[10] T. Ozcelik, J.C. Brailean and A.K. Katsaggelos, "Image and Video Compression Algorithms Based on Recovery Techniques using Mean Field Annealing," *Proceedings of the IEEE*, vol.83, no.2, pp.304-316, Feb. 1995.

[11] T. Ozcelik, J.C. Brailean, A.K. Katsaggelos, O. Erdogan and C. Auyeund, "Method and Apparatus for Spatially Adaptive Filtering for Video Encoding," U.S. Patent 5,764,307, June 9, 1998.

[12] H.W. Park and Y.L. Lee, "A Post Processing Method for Reducing Quantization Effects in Low Bit-Rate Moving Picture Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol.9, no.1, pp.161-171, February 1999.

[13] K. Ramchandran and M. Vetterli, "Rate-Distortion Optimal Fast Thresholding with Complete JPEG/MPEG Decoder Compatibility," *IEEE Transactions on Image Processing*, vol.3, no.5, pp.700-704, Sept. 1994.

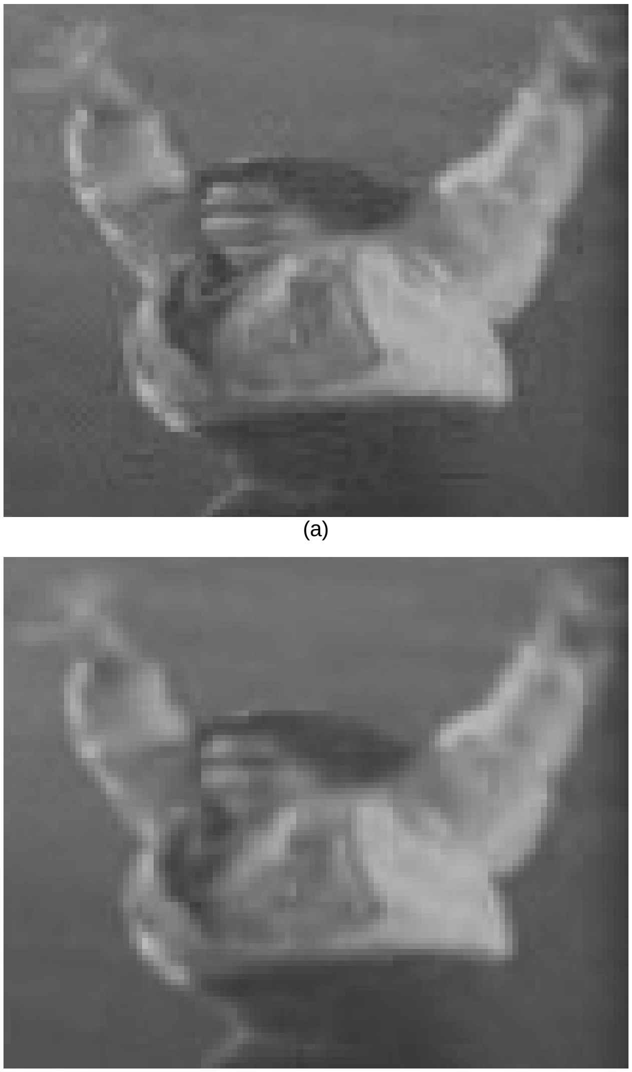
[14] H.C. Reeves and J.S. Lim, "Reduction of Blocking Effects in Image Coding," *Optical Engineering*, vol.23, pp.34-37, Jan. 1984.

[15] R. Rosenholtz and A. Zakhor, "Iterative Procedures for Reduction of Blocking Effects in Transform Image Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol.2, no.1, pp.91-94, Mar. 1992.

[16] K. Sauer, "Enhancement of low bit-rate coded images using edge detection and estimation," *Computer Vision, Graphics and Image Processing: Graphical Models and Image Processing*, vol.53, no.1, pp.52-62, Jan. 1991.

[17] C.A. Segall and A.K. Katsaggelos, "Enhancement of Compressed Video using Visual Quality Metrics," *Proceedings of the IEEE International Conference on Image Processing*, Vancouver, BC, Canada, Sept. 10-13, 2000.

[18] C.A. Segall, P. Karunaratne and A.K. Katsaggelos, "Pre-Processing of Compressed Video," *Proceedings of the SPIE Conference on Visual Communications and Image Processing*, San Jose, CA, Jan. 21-26, 2001.



(a)

(b)

Figure 2. Example of current pre-processing algorithm: (a) portion of original image corrupted by noise and coded without pre-processing; (b) same image coded after filtering the residual error.

[19] *Signal Recovery Techniques for Image and Video Compression and Transmission*, A. K. Katsaggelos and N. P. Galatsanos, editors, Kluwer Academic Publishers, 1998.

[20] C.-J. Tsai, P. Karunaratne, N.P. Galatsanos and A.K. Katsaggelos, "A Compressed Video Enhancement Algorithm," *Proceedings of the International Conf. on Image Processing*, pp.454-458, Kobe, Japan, Oct. 25-28, 1999.

[21] Y. Yang, N.P. Galatsanos and A.K. Katsaggelos, "Regularized Reconstruction to Reduce Blocking Artifacts of Block Discrete Cosine Transform Compressed Images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol.3, no.6, pp.421-432, Dec. 1993.

[22] Y. Yang and N.P. Galatsanos, "Removal of Compression Artifacts Using Projections onto Convex Sets and Line Process Modeling," *IEEE Transactions on Image Processing*, vol.6, no.10, pp.1345-1357, Oct. 1997.