DENOISING INTRA-CODED MOVING PICTURES USING MOTION ESTIMATION AND PIXEL SHIFT

Jiro Katto [†], *Junya Suzuki* [†], *Shusei Itagaki* [†], *Shinichi Sakaida* [‡] *and Kazuhisa Iguchi* [‡]

Department of Computer Science, Waseda University [†]
NHK Science & Technical Research Laboratories [‡]

ABSTRACT

This paper presents a denoising method of intra-coded pictures using motion estimation and pixel shift. Firstly, we show that pixel-aligned mixture of distorted images which are spatially shifted and differently encoded brings reduction of quantization errors. We show that this effect can be formulated as a special case of Wiener-Hopf equation and independence of quantization errors affects the performance. We then consider its application to denoising of intra-coded pictures by using motion estimation and pixel shift. Experiments using actual image sequences verify that motion estimation is effective in moving regions, pixel shift is effective in static regions and favorable PSNR gains are achieved.

Index Terms— Denoising, Motion Estimation, Pixel Shift, Wiener-Hopf Equation.

1. INTRODUCTION

Denoising (or image enhancement) of distorted images has a long history in the fields of image coding and image restoration. For compressed still images, typical examples are spatial filtering along block boundaries [1,2], POCS (projection onto convex sets) [3,4], oversampled wavelet transform [5,6] and re-application of shifted JPEG transforms [7,8]. The last two spatially shift a decoded JPEG image, re-apply JPEG compression and decompression, and mix inversely shifted images in a static manner [7] or in an edge-adaptive manner [8]. They bring PSNR gains in addition to subjective quality improvement.

For moving pictures, there are motion compensated temporal filtering [9,10,11], in-loop (deblocking) filters in H.261 and H.264/AVC, and fractional-pel accuracy motion estimation adopted since MPEG-1 [12]. Motion compensated temporal filtering applies a spatio-temporal filter along a motion trajectory to suppress noises added to image sequences. Its performance depends on motion tracking accuracy as well as adequate weight assignment to the spatio-temporal filter (as a least-square estimator).

On these backgrounds, this paper presents a denoising technique of intra-coded moving pictures exploiting encoder independence (i.e. quantization error independence). We firstly show that pixel shift reduces quantization error correlation and mixture of differently shifted images brings PSNR improvement. This effect is formulated as a special case of Wiener-Hopf equation which has been widely used in linear prediction and adaptive filters [13]. We then consider usage of motion estimation and pixel shift among intra-coded frames to reduce their quantization errors. This is a combination of the motion compensated temporal filtering [10] and the JPEG shifted transform [7]. Motion estimation is carried out at a decoder to find translated blocks in other frames, which is equivalent to pixel shift. However, since motion estimation does not contribute to static regions (having the same quantization errors), intentional pixel shift is applied at an encoder to produce different quantization results.

Most recently, Vo and Nguyen has proposed a similar and interesting method, in which motion compensated blocks of neighboring three frames are averaged with transform domain noise analysis and improvements in visual quality and PSNR are observed [14]. However, differences lie in problem formulation. In their work, an optimum solution has not yet been provided in an explicit form, for example.

This paper is organized as follows: Sec 2 describes pixel shift and mixture effects with analytical formulation. Sec 3 presents its denoising application to intraframe video codec like Motion JPEG. Sec 4 shows experimental results using actual image sequences and Sec 5 concludes this paper.

2. PIXEL SHIFT AND MIXTURE EFFECTS

Figure 1(a) shows a diagram of an auxiliary experiment, in which multiple sub-images are extracted from an original image, encoded by JPEG and JPEG-2000, and mixed up to construct a reconstructed image. Sub-images are generated by spatial pixel shift from (0,0) to (7,7), which correspond to JPEG block size and maximally up to 64 (8x8) images. Mixture operation is simply done by taking average of sub-images. Then, Figure 1(b) presents a result of pixel mixture. As expected, PSNR gains to normal compression are observed for each image and for each compression. But, PSNR gains show saturation as the number of mixed sub-images increases.

Next, we formulate pixel mixture effect. Let x be a pixel value of an original image, $\hat{x}(k)$ be a quantized pixel value of the k-th compression, $q(k)$ be its quantization error and K be the number of mixed sub-images, respectively. In addition, let $\tilde{x}$ be an estimated pixel value and $w(k)$ be a weighting coefficient for generalization. Using these notations, $\tilde{x}$ is given by

$$\tilde{x} = \sum_{k=1}^{K} w(k)\hat{x}(k) = x + \sum_{k=1}^{K} w(k)q(k) \quad (1)$$

where

$$\sum_{k=1}^{K} w(k) = 1 \cdot \quad (2)$$

Then, an optimization problem is formulated as follows: minimize variance of quantization errors

$$E\left[(x-\tilde{x})^2\right] = E\left[\left(\sum_{k=1}^{K} w(k)q(k)\right)^2\right] \quad (3)$$

subject to Eq.(2). This problem can be solved by applying Lagrange multiplier method. Let $\mathbf{q} = (q(1),q(2),\cdots,q(K))^t$, $\mathbf{w} = (w(1),w(2),\cdots,w(K))^t$ and

$$R = E\left[\mathbf{q}\cdot\mathbf{q}^t\right] = \begin{bmatrix} E[q(1)^2] & E[q(1)q(2)] & \cdots & E[q(1)q(K)] \\ E[q(2)q(1)] & E[q(2)^2] & & E[q(2)q(K)] \\ \vdots & & \ddots & \\ E[q(K)q(1)] & E[q(K)q(2)] & & E[q(K)^2] \end{bmatrix} \cdot \quad (4)$$

Then, optimum weighting coefficients are given by

$$\mathbf{w}_{opt} = C \cdot R^{-1} \mathbf{u} \quad (5)$$

where $\mathbf{u} = (1,1,\cdots,1)^t$ and C is a normalization factor given by $C = 1/\mathrm{sum}(R^{-1}\mathbf{u})$, and minimum quantization error variance is given by

$$\sigma_{q,\min}^2 = C^2 \cdot \mathbf{u}^t R^{-1} \mathbf{u} \quad (6)$$

Formulations above are basically similar to Wiener-Hopf equation used in linear prediction and adaptive filters [13]. A small difference comes from the fact that input signal x is omitted in our formulation because we only focus on quantization errors but ignore sensing noises.

Into Figure 1(b), we add theoretical curves assuming

$$R = \begin{bmatrix} 1 & a & a & \cdots & a \\ a & 1 & a & & a \\ a & a & 1 & & a \\ \vdots & & & \ddots & \\ a & a & a & & 1 \end{bmatrix} \quad (7)$$

where a represents cross-correlation between sub-images. This figure suggests that independence of quantization errors around a=0.6 brings favorable PSNR gains even if they are not completely independent (a=0).

3. DENOISING USING MOTION ESTIMATION AND INTENTIONAL PIXEL SHIFT

By considering neighboring frames as sub-images, pixel mixture in the previous section can be applied to a denoising method of intra-coded image sequences. Figure 2
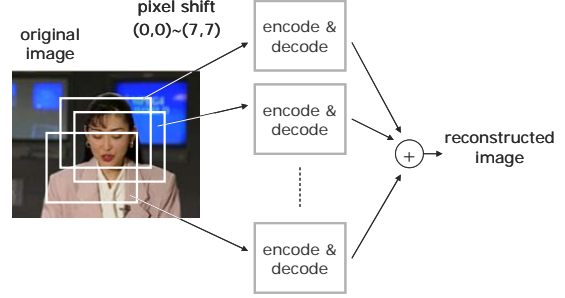
(a) A diagram of auxiliary experiment.
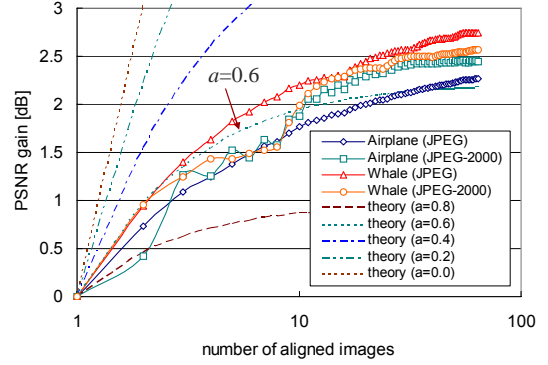
(b) PSNR gains by pixel mixture and its theoretical gains.
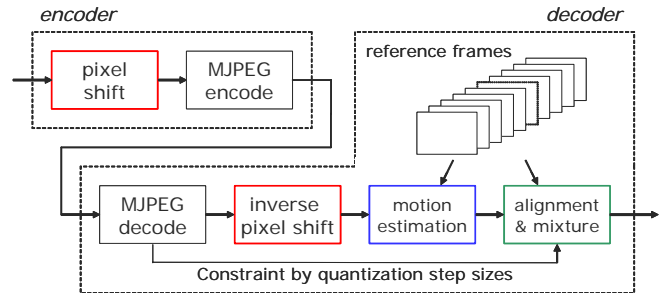Figure 1: Pixel shift and mixture effects.

Figure 2: Block diagram of the proposed intraframe codec.

depicts a block diagram of coded structure. Pixel mixture is conducted by taking a weighted sum of pixel blocks of current and neighboring frames according to Eq.(1).

Motion estimation is carried out at a decoder side only. This operation is used to find similar blocks in neighboring (past and future) frames to be mixed, and is equivalent to pixel shift effect as long as illumination change or object deformation in neighboring frames are small enough. However, when completely the same block might appear in static regions in neighboring frames, no PSNR gains are expected by motion estimation because quantization errors are also completely the same. In order to alleviate this deficiency, intentional pixel shift is inserted into an encoder and a decoder. This operation is to cause perturbation of quantization errors in static regions as well as in moving regions. There are two drawbacks in this intentional pixel shift at an encoder. The first one is an image border problem.

Circular shift can solve it but possibly degrades compression efficiency at the border. Discarding pixel slices at the border is simple but may not be permitted by some applications. The second drawback is to lose opportunity to be applied to existing Motion JPEG streams. Nevertheless, pixel shift effectiveness is promising as shown later.

On the other hand, pixel mixture is not perfect and sometimes causes quality degradation especially when occluded regions appear in a reference frame. To alleviate this problem, we force a next constraint per block by

$$\sum_{(m,n)\in block}\left|I_k(m,n)-I_{k'}(m+v_x,n+v_y)\right| \le \alpha \cdot \mathrm{quant}(k) \quad (8)$$

where $I_k(m,n)$ is a pixel value of frame k, $I_{k'}(m+v_x,n+v_y)$ is a pixel value of reference frame k' where (v_x,v_y) is an estimated motion vector of the block, $\mathrm{quant}(k)$ is a quantization step size of DC component of frame k, and α is a constant to control pixel mixture ($\alpha=2$ in most experiments as explained later). Only when Eq.(8) is satisfied, pixel mixture of the block of the reference frame k' is carried out.

4. EXPERIMENTS AND EVALUATIONS

We carry out experiments using nine CIF image sequences (and four SD image sequences). Motion estimation is done by 16x16 block matching with half-pel accuracy. Intentional pixel shift is done by horizontal 4 pixel shift for odd-number frames except the last experiment. α is fixed to 2 except the second experiment. We had used the jpeglib library to execute Motion JPEG compression, in which parameter *quality* ranging from 1 to 100 is utilized to determine compression ratio (1 is the lowest quality) and *dB* in each figure shows average PSNR of each image sequence. Firstly, Figure 3 shows an example of PSNR gains achieved by the proposed method (average PSNRs of each sequence of *quality*=10 are given in parentheses). In this experiment, two neighboring (past and future) frames are used for pixel mixture. In the figure, *fwdME* means forward motion estimation using a past frame, *biME* means bidirectional motion estimation using two frames, and *shift* means intentional pixel shift. From this figure, we can recognize that (1) bidirectional motion estimation works well for pictures having large moving regions like Flower&Garden and Mobile&Calendar, (2) intentional pixel shift works well for pictures having static regions like Akiyo, and (3) bidirectional motion estimation combined with pixel shift always performs the best. Though omitted, we observed similar tendency for the Motion JPEG-2000 case.

Secondly, Figure 4 shows effect of parameter α on denoising performance (*quality*=25). Generally speaking, when α is too small, pixel mixture operation is frequently avoided even if PSNR gains are expected. But, when α is too large, drastic quality degradation might happen even if its frequency is low. To balance this tradeoff, we chose $\alpha=2$ for other experiments.
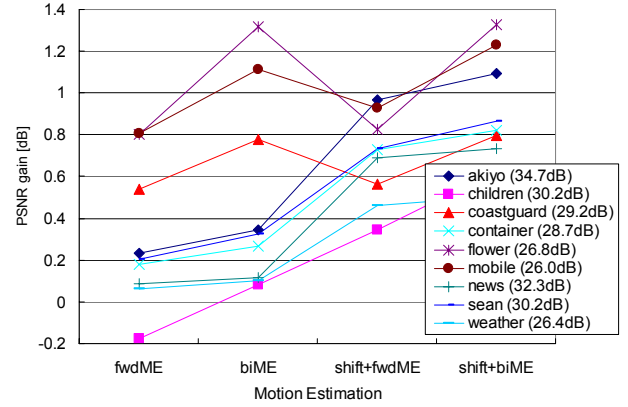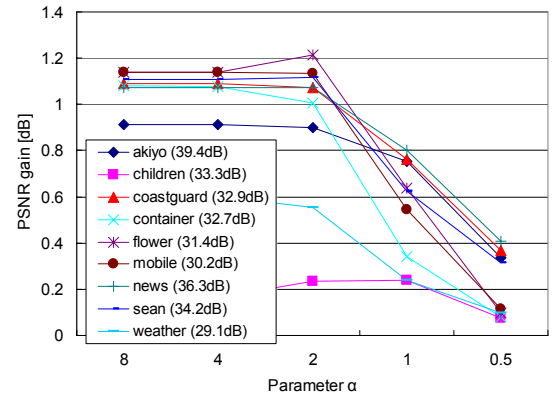

Figure 3: PSNR gains achieved by the proposed method.

Figure 4: Effect of parameter α on denoising performance.

Finally, we show correspondence between theory and experiments, effect of pixel shift patterns and subjective comparison of reconstructed images. Here we consider three pixel shift patterns of which shift values for the k-th frame are given by $8/m*\mathrm{mod}(k,m)$ where m=2,4,8, corresponding to $(0,4,0,4,\ldots)$, $(0,2,4,6,\ldots)$ and $(0,1,2,3,4,\ldots)$, respectively. Figure 5 shows averaged cross-correlation matrix R normalized by quantization error variance of the current frame for the case $m=8$, in which x-axis and y-axis represent frame distance (center is the current frame) and z-axis denotes cross-correlation given by Eq.(4). Different from Eq.(7) assuming the same image, quantization error variance (i.e. diagonal component) increases as frame distance increases (larger than that of the current frame). However, since cross-correlations of reference frames to the current frame are reduced below 0.8 thanks to the pixel shift effect, PSNR gains by pixel mixture can be still achieved.

Once cross-correlation matrix R is achieved, optimal weighting coefficients for each frame and expected PSNR gains can be calculated by Eqs.(5) and (6). Figure 6 presents a comparison result of expected PSNR gains (theory) and actually achieved PSNR gains (sim) using optimal weighting coefficients for Akiyo sequence. When the frame distance (x-axis) is n, this means that we use maximally 2*n

(*n* past and *n* future) reference frames for the pixel mixture. Although there are gaps between theory and experiments, performance order and behavior are indeed the same.

Figure 7 shows subjective comparison of a Motion JPEG image and a mixture image by the proposal for Flower &Garden (*quality*=5). We can notice much distortions in the Motion JPEG image, but they are reduced in the proposal. Furthermore, in temporal direction, we observe that flicker noises can be impressively suppressed by the proposal.
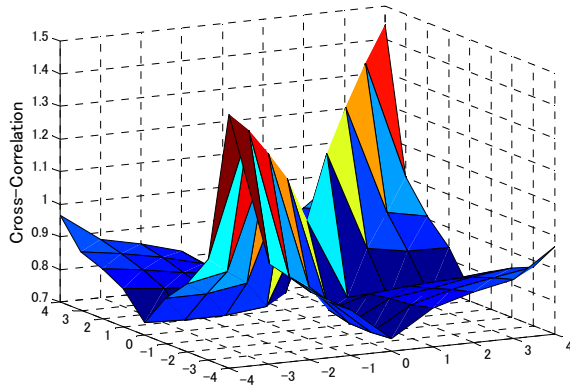

Figure 5: Normalized cross-correlation matrix *R* of intra-coded image sequences in case of *m*=8 pixel shift pattern.
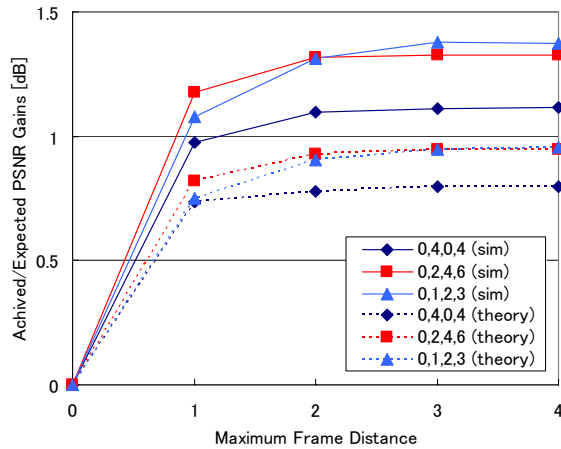

Figure 6: Comparison of expected PSNR gains (theory) and achieved PSNR gains (sim) for Akiyo sequence.

(a) Motion JPEG   (b) Proposal
Figure 7: Subjective comparison of reconstructed images.

5. CONCLUSIONS

This paper presented a denoising method of intra-coded image sequences using motion estimation and pixel shift. Mathematical formulation of the pixel mixture effect was derived as a special case of Wiener-Hopf equation, and experiments using actual image sequences were carried out to validate effectiveness of the proposed method and to verify theoretical performance expectations. As future work, in addition to theory refinement to fill the current theory-simulation gap, locally adaptive mixing schemes like [8,14] should be considered. Current weighting operation is applied in a global manner to each block similarly per frame. However, possibly due to motion estimation slip, blurs are sometimes observed (e.g. on a moving tree in Flower&Garden) even if PSNR gains are achieved. Furthermore, we will also consider extension of our work to the interframe case.

5. REFERENCES

[1] H.C.Reeve and J.S.Lim: "Reduction of Blocking Effect in Image Coding," IEEE ICASSP'83, Apr.1983.
[2] B.Ramamurthi and A.Gersho: "Nonlinear Space-Variant Postprocessing of Block Coded Images," IEEE Trans. on ASSP, Vol.34, No.5, pp.1258-1268, Oct.1986.
[3] A.Zakhor: "Iterative Procedures for Reduction of Blocking Effects in Transform Image Coding," IEEE Trans. on CSVT, Vol.2, No.1, pp.91-95, Mar.1992.
[4] Y.Yang, N.P.Galatsanos and A.K.Katsaggelos: "Regularized Reconstruction to Reduce Blocking Artifacts of Block Artifacts of Block Discrete Cosine Transform Compressed Images," IEEE Trans. on CSVT, Vol.3, No.6, pp.421-432, Dec.1993.
[5] R.A.Gopinath, M.Lang, H.Guo and J.E.Odegard: "Wavelet-based Post-Processing of Low Bit Rate Transform Coded Images," IEEE ICIP'94, Nov.1994.
[6] R.R.Coifman and D.L.Donoho: "Translation-Invariant De-noising," Wavelets and Statistics, Springer Lecture Notes, 1994.
[7] A.Nosratinia: "Denoising JPEG Images by Re-Application of JPEG", IEEE Multimedia Signal Proc., Dec.1998.
[8] R.Samadani, A.Sundararajan and A.Said: "Deringing and Deblocking DCT Compression Artifacts with Efficient Shifted Transforms," IEEE ICIP 2004, Oct.2004.
[9] E.Dubois and M.S.Sabri: "Noise Reduction in Image Sequences using Motion-Compensated Temporal Filtering," IEEE Trans. on Commun., Vol.32, No.7, pp.826-831, Jul.1984.
[10] M.K.Ozkan, M.I.Sezan and A.M.Tekalp: "Adaptive Motion-Compensated Filtering of Noisy Image Sequences," IEEE Trans. on CSVT, Vol.3, No.4, pp.277-290, Aug.1993.
[11] J.R.Ohm: "Three-Dimensional Subband Coding with Motion Compensation," IEEE Trans. on Image Proc., Vol.3, No.5, pp.559-571, Sep.1994.
[12] T.Wiegand, G.Sullivan, G.Bjontegaard and A.Luthra: "Overview of the H.264/AVC Video Coding Standard," IEEE Trans. on CSVT., Vol.13, No.7 pp.560-576, July 2003.
[13] N.S.Jayant and P.Noll: "Digital Coding of Waveforms," Prentice Hall, 1984.
[14] D.T.Vo and T.Q.Nguyen: "Quality Enhancement for Motion JPEG using Temporal Redundancies," IEEE ICIP 2007, Sep.2007.