MATCHING PURSUITS MULTIPLE DESCRIPTION CODING FOR WIRELESS VIDEO

Xiaoyi Tang and Avideh Zakhor

Department of Electrical Engineering and Computer Science
University of California at Berkeley
{xiaoyi, avz}@eecs.berkeley.edu

ABSTRACT

Multiple description coding (MDC) is an error resilient source coding scheme that creates multiple bitstreams of approximately equal importance. We develop a 2 description video coding scheme based on the 3 loop structure originally studied in [1]. We modify the discrete cosine transform structure to the matching pursuits framework and evaluate performance gain using maximum likelihood (ML) enhancement when both descriptions are available. We find that ML enhancement works best for low motion sequences. Performance comparison is made between our MDC scheme and single description coding (SDC) schemes over two-state Markov channels and Rayleigh fading channels. We find that MDC outperforms SDC in bursty slowly varying environments. In the case of Rayleigh fading channels, interleaving helps SDC close the gap and even outperform MDC depending on the amount of interleaving performed, at the expense of additional delay.

1. INTRODUCTION

Multiple description coding (MDC) generates multiple encoded bitstreams of a source and sends them through several independent channels. The source can be reconstructed at a lower yet acceptable quality from any single bitstream, and at a higher quality from more bitstreams. Unlike layered coding which requires correct reception of the base layer for enhancement layers to be useful, MDC can reconstruct the source from any subset of bitstreams. MDC provides a robust source coding scheme for communication over unreliable channels without quality-of-service guarantee.

There has been a great deal of work on designing practical MDC schemes, especially for video [1, 2]. The challenge in video MDC design arises due to the predictive nature of video. The decoder could receive any subset of all descriptions and the encoder must track state of the decoder to prevent drift. In this paper, we propose a new video MDC scheme based on matching pursuits (MP). The outline of the paper is as follows. In Section 2, we migrate the H.263 based 3 prediction loop MDC scheme in [1] to the MP video coding framework [3]. In Section 3, we improve the rate-distortion performance of the scheme by applying enhancement techniques to all three reconstructions when both descriptions are available. Finally, in Section 4, we compare the performance of our MDC scheme with traditional single description coding schemes over two-state Markov channels and Rayleigh fading channels through simulations.

2. MATCHING PURSUITS VIDEO MDC

Our video MDC scheme is based on the three prediction loop structure proposed by Reibman *et al.* [1] and the MP framework

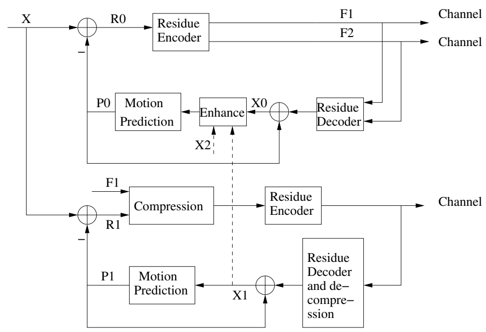

Fig. 1. *Diagram of multiple description video coding.*

[3]. Almost all existing video coding standards use the hybrid motion compensated DCT. However, block-based DCT coding introduces noticeable distortion and block artifacts at lower bit rates. In video coding, the residual from motion compensation is expanded onto a larger basis set than the complete set provided by DCT. At each stage of coding the residual, a search for the best basis function, also called an atom, is performed by computing the inner product between the residual and a basis function. The residual is subtracted from the best basis function and the iteration repeats. This ensures that most important features are coded first.

The diagram of the three loop MD video coding is shown in Figure 1. Prediction is essential to achieve video coding efficiency. In two description coding, the decoder could receive either one or two descriptions. Thus there are three prediction loops in the encoder so that the decoder could still track the encoder state when a description is lost. In the central prediction loop, a new frame is first motion compensated from its prediction based on both descriptions. The residue is then coded into two correlated bit streams that are sent to two separate channels. In [1], a DCT transform is first used to transform the residue to the frequency domain, followed by a correlation matrix. In our matching pursuits scheme, the residue is coded into two sets of atoms, F1 and F2, to be sent over two channels. The first L atoms found during MP iterations are shared by both sets and subsequent atoms are alternatively put into the two sets. The correlation between these two sets of atoms is controlled by the number of shared atoms L. Motion vectors, frame headers, and intra-coded (I) frames are duplicated and sent through both channels. In the two symmetric side loops, of which, only one is drawn in the diagram, we also use MP coding. There is no motion estimation in the side loops as the motion vectors are taken from the central loop. The energy of the residue R1(R2) from compensation based on one description is first reduced by exploiting its correlation with the coded residue from the central loop F1(F2) through simple pixelwise subtraction.

Tradeoff between the quality of reconstruction from two descriptions (PSNR0) and quality of reconstruction from one de-

This work was supported by NSF grant 9979442.

scription (PSNR1) can be controlled in two ways, namely by rate allocation between the central and side loops, as well as the number of shared atoms L between two atom sets F1 and F2; increasing L reduces PSNR0 and increases PSNR1. The tradeoff is studied in Section 3, after we examine one improvement to the MDC scheme.

3. ENHANCEMENT

In [1], the side loops are considered to be pure redundancy and so, when both descriptions are received, the reconstructed video is taken from the central loop. Since each of three prediction loops has its own reconstruction of the source frame, we can combine all three reconstructions to yield a better reconstruction than the one in the central loop. This is a special case of the multi-channel restoration problem [4].

3.1. Model

The reconstructed video in each loop can be modeled as

$$y_i(l,m) = x(l,m) + n_i(l,m) \qquad i = 1,2,3 \tag{1}$$

where i is the loop index, $y_i(l,m)$ is the reconstructed pixel value at position (l,m) in loop i, $x(l,m)$ is the pixel at (l,m) in a source frame, and $n_i(l,m)$ is the additive noise. In vector notation, let $\overrightarrow{X}$, a $M \times 1$ vector, denote the lexicographical ordering of source image, i.e. $\overrightarrow{X} = [x(1,1), x(1,2), x(1,3), \ldots]^T$. Let

$\overrightarrow{Y} = [y_1(1,1), y_2(1,1), y_3(1,1), y_1(1,2), y_2(1,2), y_3(1,2), \ldots]^T$

and

$\overrightarrow{N} = [n_1(1,1), n_2(1,1), n_3(1,1), n_1(1,2), n_2(1,2), n_3(1,2), \ldots]^T$

Rewriting Eq (1) in vector form, we get

$$\overrightarrow{Y} = A\overrightarrow{X} + \overrightarrow{N} \tag{2}$$

where A is a $3M \times M$ matrix defined by

$$A = \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & a & 0 & 0 \\ 0 & 0 & .. & 0 \\ 0 & 0 & 0 & a \end{bmatrix} \tag{3}$$

with $a = [1,1,1]^T$.

3.2. Performance of ML Enhancement

There are several methods to estimate $\overrightarrow{X}$ given $\overrightarrow{Y}$ in Eq (2), namely least squares (LS), regularized least squares (RLS), Maximum Likelihood (ML), Maximum A-Posteriori (MAP), and Linear Minimum Mean Square Error (LMMSE). We use ML estimation because it outperforms LS and RLS, and is simpler than MAP and LMMSE. Assuming the noise is zero-mean Gaussian[1], ML estimation is given by

$$\hat{\overrightarrow{X}}_{ml} = (A^T R^{-1} A)^{-1} A^T R^{-1} \overrightarrow{Y} \tag{4}$$

where $R = E\{\overrightarrow{N}\overrightarrow{N}^T\}$ is the covariance matrix of $\overrightarrow{N}$. We use pixelwise estimation to keep overhead of coding estimated coefficients low. Specifically, we approximate R as a block diagonal matrix

$$R = \begin{bmatrix} R_n & 0 & 0 & 0 \\ 0 & R_n & 0 & 0 \\ 0 & 0 & .. & 0 \\ 0 & 0 & 0 & R_n \end{bmatrix} \tag{5}$$

where R_n is a 3×3 correlation matrix of noise values in the same pixel location of three reconstructions. Therefore, Eq (4) reduces to

$$\hat{\overrightarrow{X}}_{ml}(l,m) = (a^T R_n^{-1} a)^{-1} a^T R_n^{-1} [y_1(l,m), y_2(l,m), y_3(l,m)]^T \tag{6}$$

Since the decoder does not have information about noise, the encoder must encode the 3 ML coefficients given by the elements of the 1×3 vector $(a^T R_n^{-1} a)^{-1} a^T R_n^{-1}$. In practice, R_n is computed first in the encoder by averaging samples over an entire frame. Since the sum of three ML coefficients in Eq (6) is 1, only two coefficients are quantized and encoded at 6 bits. It creates a nominal 12 bits overhead for each frame, even with fixed length coding.

To evaluate the performance of ML enhancement, we code a number of QCIF sequences as shown in Table 1. In the central loop, 50 atoms/frame/description are coded, with the first 15 atoms shared between the two descriptions and remaining 70 atoms alternatively split between the two descriptions. In the side loops, additional 30 atoms/frame/description are coded. All sequences, except Larry, are 10 seconds in duration and 10 fps. The first frame of each sequence is intra-coded with quantization step size equal to 8. For all sequences except Larry, all other frames are inter-coded unless a scene change is detected, in which case the frame is intra-coded. For the Larry sequence, we have 1689 frames at 15 fps. To avoid excessive error propagation, an I frame is encoded every 100 frames for Larry, in addition to I frames due to scene change. Total bit rates of two descriptions and average PSNR without ML enhancement of the sequences are listed in the upper half of Table 1. For comparison, average PSNR of a single description MP video codec at the same bit rate is listed in the SDC column, and the difference in PSNR0 between SDC and MDC without ML estimation is listed in the Δ_e column. MDC has a lower PSNR0 than SDC due to the redundancy.

Sequence	bitrate (kbps)	MDC w/o ML est. PSNR0 (dB)	MDC w/o ML est. PSNR1 (dB)	SDC PSNR0 (dB)	Δ_e (dB)
Hall	45.2	35.70	34.66	37.01	1.31
Akiyo	42.6	38.27	37.01	40.52	2.25
Foreman	85.6	30.92	30.28	34.32	3.40
Larry	74.4	34.19	33.24	36.18	1.99

Sequence	ML w/o fb. Δ_a (dB)	ML est. with fb. Δ_b (dB)	ML est. with fb. Δ_c (dB)	Δ_d (dB)
Hall	0.52	0.66	0.47	0.65
Akiyo	0.40	0.65	0.57	1.60
Foreman	0.16	0.15	0.19	3.25
Larry	0.29	0.44	0.41	1.55

Table 1. *MDC/SDC bit rate and average PSNR. Δ_a: the difference between MDC PSNR0 using ML estimation without feedback (ML w/o fb.) and MDC PSNR0 without ML estimation. Δ_b: the difference between MDC PSNR0 using ML estimation with feedback and MDC PSNR0 without ML estimation. Δ_c: the difference between MDC PSNR1 using ML estimation with feedback and MDC PSNR1 without ML estimation. Δ_d: the difference between SDC PSNR0 and MDC PSNR0 using ML estimation with feedback. Δ_e: the difference between SDC PSNR0 and MDC PSNR0 without ML estimation.*

We first examine the performance gain by using ML enhancement without feedback. By this we mean the reconstruction from the central loop, not the enhanced reconstruction, is used in the encoder central prediction loop and the enhanced reconstruction is used in the decoder. The baseline scheme for comparison has no ML enhancement and uses the reconstruction in the central loop when both descriptions are received. The average gain in PSNR0 from ML enhancement without feedback is listed in Δ_a column in Table 1. It can be seen that the gain achieved by ML enhancement varies from sequence to sequence. ML enhancement works better for sequences with low motion, such as Hall and Akiyo, than high motion sequences.

Additional gain can also be achieved by feeding back the enhanced reconstruction to the central loop. More specifically, the

[1] Strictly speaking, the noise is not Gaussian, though it has a bell-shaped distribution. However, Gaussian model is still used to facilitate analysis.

enhanced reconstruction of the previous frame is used for motion prediction and compensation of the current frame in the central loop. Table 1 lists the average gain in PSNR0 from the feedback in column Δ_b, as well as the gain in PSNR1 in column Δ_c, over MDC without ML estimation. As seen, there are gains in PSNR1. This is because the use of enhanced reconstruction from two descriptions in the central prediction loop creates better motion vectors and two sets of atoms, namely F1 and F2, that reduce the residue energies in the side loops. The variation in PSNR1 gain from feedback among different sequences is related to the variation in PSNR0 gain from ML enhancement, as shown in column Δ_a. There is little additional gain in PSNR0 due to feedback; the difference between Δ_b and Δ_a is small, except for a few low motion sequences. The combined average gains in both PSNR0 and PSNR1 are quite significant for low motion sequences. Column Δ_d in Table 1 lists the difference in PSNR0 between SDC and MDC using ML estimation with feedback. The difference is smaller for low motion sequences due to better gain from ML enhancement and less redundancy in MDC from motion vector duplications. Comparing Δ_d and Δ_e columns, we see that the gap between PSNR0 of MDC and SDC has been reduced after ML estimation.

We have studied the performance of ML enhancement with the number of atoms per description per frame in the central loop (C=50) and side loops (S=30), as well as number of shared of atoms in the central loop (L=15) fixed. We can keep the bit rate of both descriptions constant and study the tradeoff between PSNR0 and PSNR1 by varying C, S, and L. To obtain the PSNR0 and PSNR1 tradeoff curve at approximately constant bit rate for each description, we fix total number of atoms per frame per description, $\mathrm{C} + \mathrm{S} = 80$, examine a number of combinations of L and S, and take the concave hull of the measured data points (PSNR0, PSNR1). More specifically, L and S take values from sets $\{1\ 5\ 15\ 25\}$ and $\{5\ 10\ 15\ 20\ 30\ 40\}$, respectively. We find that changing S is a more efficient way of allocating redundancy than changing L since most data points on the concave hull correspond to small values of L. The tradeoff with and without ML enhancement is shown in Figure 2. The bitrates for Foreman and Larry are 85.6 kbps and 74.4 kbps, respectively. As can be seen, ML enhancement improves the performance compared to the baseline case without ML enhancement. Gains are smaller at large PSNR0 since fewer bits are spent in the side loops.

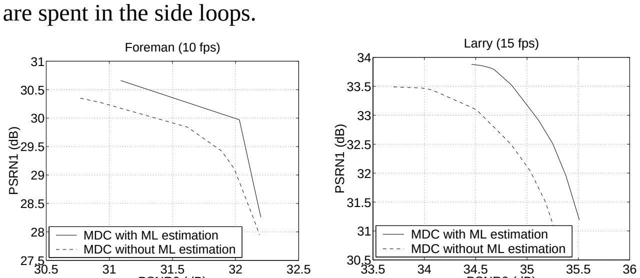

Fig. 2. *Tradeoff between PSNR0 and PSNR1 with and without ML enhancement.*

We also compare the Redundancy Rate Distortion (RRD) performance of our MP based MDC (MP-MDC) scheme with the H.263 based MD coder (MDTC) in [1]. Redundancy rate is defined as the additional bit rate of a MDC scheme achieving the same two description distortion of a SDC scheme. In this test, we use Foreman at a frame rate of 7.5 fps and reference bit rate of 49.4 kbps with PSNR0 fixed at 31.38 dB for both schemes. One out of every 15 frames is intra coded with quantization step size of 12. Figure 3 compares the RRD performance of the two coders by plotting PSNR1 versus bit rate. Redundancy rate is expressed as a percentage of the reference rate. Due to frequent intra coding and duplication of I frames in MP-MDC, the redundancy starts at a larger value than MDTC, which uses MD transform coding for I frames as well. As seen, MP-MDC significantly outperforms MDTC in PSNR1.

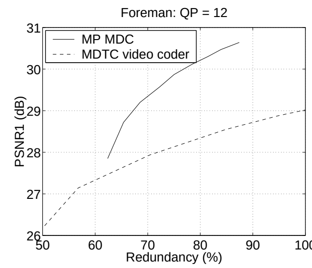

Fig. 3. *RRD performance of MDTC coder and MP-MDC.*

4. PERFORMANCE OVER LOSSY CHANNELS

In this section, we compare the performance of our video MDC scheme and single description coding (SDC) schemes over lossy channels through simulation. There are two independent channels with each description generated by MDC sent over one channel. For SDC, the bitstream is split and sent through two channels. Intuitively, we would expect MDC to outperform SDC if one channel introduces large bit error rate (BER) beyond correction capability of forward error correction (FEC) schemes employed, i.e. one channel is off, while the other channel has low BER. Under these conditions, MDC for video can reconstruct the source from only one description without losing synchronization with the encoder while SDC is likely to lose synchronization and cause error propagation. The more slowly varying the two channels are, the longer the error propagation for the SDC case, and hence the more advantageous MDC becomes over SDC. Since in video communication variation of video quality over time generally needs to be minimized, MDC could potentially offer advantages over SDC in slowly varying channels. In what follows, we will demonstrate this more quantitatively through simulations with two state Markov and Rayleigh fading channels.

We choose two MP based SDC schemes for comparison. **SDC1**: We code at the same source rate as two descriptions in MDC and use the same FEC for atoms so that total rate is the same. **SDC2**: We code at such source rate that PSNR is same as two description MDC with no channel error, and choose FEC to match the overall rate with MDC. Comparisons in the literature are typically between MDC and SDC1 over lossy channels, showing that MDC outperforms SDC1 at large loss rates and vice versa at small loss rates due to redundancy in MDC. We believe a more interesting comparison would be between MDC and SDC2, both having same error free performance. SDC2 allocates all available redundancy to FEC while MDC uses weaker FEC and allocates remaining redundancy to source coding. Comparison between MDC and SDC2 would test the efficiency of a MDC design.

For fair comparison, we assume the FEC protection methods for I frames, motion vectors and headers are the same in both MDC and SDC schemes. Due to the importance of I frames, we send two copies of I frames, one to each channel with BCH(127,64,10) protection for both MDC and SDC schemes. If there are remaining errors in an I frame, reconstructed GOB(s) that are affected by the errors are replaced by the corresponding GOB(s) in the previous frame. Motion vectors are also duplicated, protected by

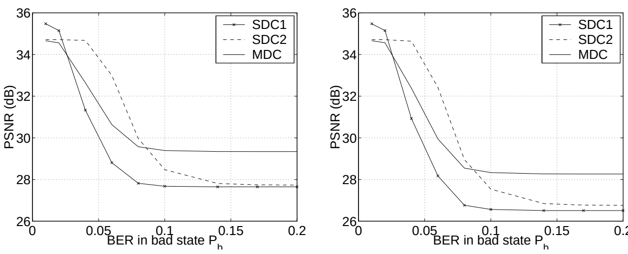

Fig. 4. *Performance of SDC and MDC schemes in two-state Markov channels.* $P_g = 0.001$ *(a) slowly varying channels* $p_1 = 0.05, p_2 = 0.05$ *(b) fast varying channels* $p_1 = 0.8, p_2 = 0.8$

BCH(127,64,10) code and sent over two channels. If there are errors in motion vectors, default value of zero is assumed.

Atoms in SDC are equally divided among two channels per frame by allocating atoms in the even GOBs to one channel and atoms in the odd GOBs to the other channel. The FEC used for atoms in MDC and SDC1 is BCH(127,78,7). For atoms in SDC2, the FEC used is BCH(127,50,13). The overall bit rate after FEC is 140 kbps for all schemes. When atoms are received incorrectly, all remaining atoms in the same GOB are lost due to loss of synchronization in entropy coding. For MDC, we apply a simple error concealment method: There is a large mismatch between encoder and decoder when one channel is off causing the loss of many atoms. The mismatch can be reduced, if the other channel has small errors and atoms in the description received through the good channel are copied to the other description because of the redundancy in the two sets of atoms of two descriptions. The decision when to apply this method is heuristic; in the following simulations, it is applied when one channel receives 30 or more atoms than the other.

4.1. Two State Markov Channels

In this model, a channel can be in either "good" state with bit error rate (BER) P_g, or "bad" state with BER P_b. The state transition probabilities are p_1 (from "good" state to "bad" state") and p_2 (from "bad" state to "good" state"). In the simulations, we assume channels can change state only at the start of a frame transmission to achieve relatively slowly varying channels.

Figure 4 plots the average PSNR of MDC and two SDC schemes versus BER in the bad state P_b for the Larry sequence with different channel transition probabilities. At low BER, SDC1 outperforms MDC and SDC2 because all errors can be corrected by the FEC schemes used and SDC1 has the least redundancy. As error rate increases (between approximately 3% and 9%), MDC outperforms SDC1 because of redundancy in MDC. However, SDC2 is the best because it has the strongest FEC protection. As BER increases further, MDC outperforms SDC2 since most errors in the bad state can no longer be corrected. This verifies our intuition that MDC outperforms SDC in slowly varying bursty channels. As the channel transition probabilities increase from 0.05 in Fig. 4(a) to 0.8 in Fig. 4(b), both MDC and SDC perform worse with a decrease of nearly 1 dB in high loss situation, and the MDC gain over SDC at $P_b = 20\%$ drops slightly from 1.6 dB to 1.5 dB.

4.2. Rayleigh Fading Channels

We measure the performance of MDC and SDC schemes over Rayleigh fading channels simulated using Jakes' method. Two channels are assumed to correspond to two carrier frequencies at 900 MHz and 1.8 GHz, respectively, the motivation being cell phones which operate at both frequencies depending on whether they are used in USA or Europe. A 200 kHz channel bandwidth and binary phase shift keying modulation are assumed. We run tests in both slowly varying channels and fast varying channels. A block interleaver is used to study the effect of interleaving delay.

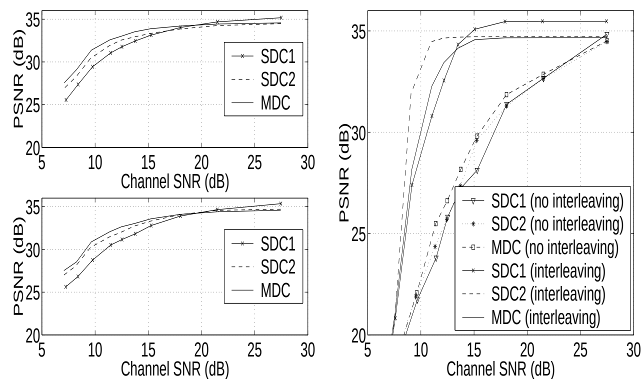

Fig. 5. *Performance of SDC and MDC schemes in Rayleigh fading channels. (a) slowly varying channels with* $f_D T$ *of* 5×10^{-6} *and* 10^{-5} *for the two MDC channels (b) fast varying channels with* $f_D T$ *of* 3×10^{-4} *and* 6×10^{-4} *for the two MDC channels*

Figures 5(a) and 5(b) show the performance of SDC and MDC schemes in slowly and fast varying Rayleigh channels, respectively. The Doppler frequency f_D equals vf_c/c, where v is the mobile speed, f_c is the carrier frequency, and c is the light speed. The product $f_D T$, where T is the bit interval, corresponds to the varying speed of a Rayleigh channel. Since the mobile moves at a given speed, and one carrier frequency is twice the other one, this results in $f_{D1}T$ for channel 1 to be twice as large as $f_{D2}T$ for channel 2. As shown in the top plot of Fig. 5(a), in slowly varying channels, without interleaving, MDC outperforms SDC1 and SDC2 at channel SNRs below 18 dB. As shown in the bottom plot of Fig. 5(a), bit interleaving is not effective at combating slowly varying fading channels even though the performance gap between MDC and SDC2 becomes smaller with interleaving delay of 100 milliseconds. In faster varying channels as shown in Figure 5(b), without interleaving all three schemes have similar performance, and with interleaving SDC2 outperforms MDC at channel SNRs between 7 and 12 dB. Thus, bit interleaving is quite effective for fast varying fading channels. As channels become more uniform in bit error occurrence with increasing interleaving delay, FEC protection ensures no performance degradation above certain thresholds in channel SNR.

5. CONCLUSION

We study a video MDC scheme based on a 3 loop structure and MP. We examine performance gain through ML enhancement from both descriptions. Performance comparison is made between MDC and SDC schemes over lossy channels. We find that MDC with large redundancy outperforms SDC in slowly varying environment with bursty nature. In the Rayleigh fading channels, bit interleaving helps SDC close the gap and even outperform MDC depending on the amount of interleaving and speed of channel variation.

6. REFERENCES

[1] A. Reibman, Y. Wang, M. Orchard, and R. Puri, "Multiple description coding for video using motion compensated prediction," in *Proc. ICIP 99*, pp. 837–841, October 1999.

[2] V. Vaishampayan and S. John, "Interframe balanced multiple description video compression," in *Proc. Packet Video 1999*, April 1999.

[3] R. Neff and A. Zakhor, "Very low bit-rate video coding based on matching pursuits," *IEEE Trans. CSVT*, vol. 7, pp. 158–171, February 1997.

[4] A. Katsaggelos, "A multiple input image restoration approach," *Journal of Visual Communication and Image Representation*, vol. 1, pp. 93–103, September 1990.