VIDEO ERROR CONCEALMENT USING SPARSE RECOVERY AND LOCAL DICTIONARIES

Dzung Nguyen, Minh Dao, Trac Tran

Johns Hopkins University, Electrical and Computer Engineering
{dzungnguyen, minh.dao, trac}@jhu.edu

ABSTRACT

Video error concealment is a post-processing technique that conceals the errors in a decoded video sequence based on data available only at the decoder. Most of the current techniques adopt the approach that recovers the Motion Vector (MV) of a lost image block, uses that MV to look for data to fill in the blank then performs some refinements. We propose a method that does not rely on MV recovery, but essentially bases on sparse representation of image patches on local temporal dictionaries. Experiment results show a large improvement over Boundary Matching Algoritm (BMA), the standard method used in reference software for H.264 video codec.

Index Terms— Error concealment, Sparse recovery, Local dictionary, l_1 minimization, BMA

1. INTRODUCTION

Video communication is prone to errors in transmision media. The source of errors can be packet loss in packet switching network,losing of synchronization in streaming application, or simply damages on storage media which cause parts of a video frame undecodable. Video coding methods themselves are not robust to transmission error, since the goal is to elinimate the redundancy as much as possible to achieve the best coding efficiency. Addition mechanisms are therefore needed to cope with this transmission errors.

Various approaches had been proposed [1]. Techniques like Forward Error Correction (FEC), Robust Entropy Coding (REC), Layer Coding (LC), Multiple Description Coding (FEC), packet retransmission... impose certain requirements either on the design of the source coder, structure of the network, or technologies used in the video encoder. However, it is a common situation that the receiver has to correct errors independently of the coder, the transmitter and the network. Examples are digital broadcasting, video streaming over low bitrate network, or video offline distribution where the sender is not able to help individual receiver, or simply no longer available to help. In the context of this paper, video error concealment therefore implies receiver-based post processing techniques.

Error concealment can be done in spatial-domain, temporal-domain, transform-domain, or a combination of those [1]. In fact, methods which combine multiple techniques give pleasant performance and are used in practice [2, 3].

In the paper, we propose a error concealment method which is essentially temporal-domain recovery. However, unlike most of the temporal-domain techniques, our method does not base on motion vectors. Our method bases on the sparse representation of image block over a local dictionary built from blocks of adjacent video frames in a close neighbourhood. Experiment results show a significant improvement over BMA in terms of PSNR of reconstructed frames. Although BMA [3], the standard concealment method used in H.264 reference software, is implemented to for benchmarking purpose, it is important to mention that our method is not only applicable to H.264. Instead, as a post-processing technique working directly on image domain, it can be used with any video codec and any video transmission system over packet networks.

The paper is structured as follows: Section 2 gives brief summaries on BMA and sparse recovery, Section 3 describes the method in details, Section 4 shows some experimental results and comparisons between the proposed method and BMA, Section 5 concludes and discusses some aspects for further improvement.

2. BACKGROUNDS

2.1. Boundary Matching Algoritm

BMA served as a referenced method to benchmark many concealment methods [4, 5]. Therefore, it is worthwhile to have it shortly described.

The tranportation environment is supposed to be packet network, so losing data packets will result in missing macroblock (MB) of group of blocks in decoded frames. BMA is therefore a MB-based method. In each decoded frame, it is also assumed that the map of missing MB is known. BMA try to recover the missing area block-by-block in a specific order: column-by-column from the boundaries inwards. The concealed MBs can then be used as referenced MB to recover neighbour lost MBs. To simplify the scenario, we focus on

BMA motion compensated frames only and assumed the key frames, if exist, are well protected.

A lost MB may imply the loss of the residue, the MV or both. In case the MV is lost, BMA try to predict it from the set of zero motion vector and neigbour MVs of non-missing MBs. The one selected is the one that gives minimum total variation between the boundary pixels of the compenstated MB and its outside pixels.

2.2. Sparse recovery

Sparse recovery refers to methods which solve a linear system for sparse solution, which means a solution with smallest number of non-zero entries. The problem can be depicted as

$$\text{Solve} \quad y = Ax \quad \text{for sparse } x \tag{P1}$$

It is a well-known result [6] that in order to solve this system, or its version with the presence of noise, $y = Ax + n$, we can instead solve a convex optimization problem

$$\text{Solve} \quad \min \|x\|_1 \text{ w.r.t } \|y - Ax\| \leq \epsilon \tag{P2}$$

There are numerous algorithmic approaches for solving (P2), and a popular choice is to solve a convex quadratic minimization (with some specific choice of Lagrange multiplier τ)

$$x^* = \text{argmin}\{\tfrac{1}{2}\|y - Ax\|^2 + \tau\|x\|_1\} \tag{P3}$$

3. VIDEO CONCEALMENT USING SPARSE REPRESENTATION ON LOCAL DICTIONARIES

In this section, we describe the main ideas of representing image patches on local dictionaries, recovery of sparse solution based on partial observation and how to apply those to recover mising blocks in video sequences.

3.1. Local dictionary of an image patch

In motion compensated video coding, a MB assumed to have a best match in an adjacent reference frame, it then can be encoded by a MV that points to the referenced match and a residue. Generalizing the idea of this representation, we assume MBs have a sparse representation on a dictionary built from blocks of the same size in the same area used for motion vector search. In other words, this says the MB can be represented closely by a linear combination of a few number of image blocks in the referenced area (or columns in the local dictionary).

$$p = D\alpha + \epsilon \tag{1}$$

In the above equation, p is the vectorized image patch (or MB) of interest, D is a local dictionary whose columns are vectorized patches collected from referenced frames, in a close spatial neighbourhood of p, α is the vector of coefficients and supposed to be sparse, ϵ is the residue.

The representation of MB with MV then can be seen as just a specific case in this new framework where α has only one non-zero coefficient which equals 1, the corresponding active columns in D is the best matched block found by block matching algorithm, and ϵ is the difference between the MB and its best match. In case a MB has a perfect match with zero residue, e.g. the MB contains an object in a translational motion, solving (1) should give us back the MV representation if sparse recovery solver is set up right. In other cases, linear combination of several blocks generally give a better approximation of the MB than using only one reference block. Figure 1 visualizes the idea. As one can see, in this approach,

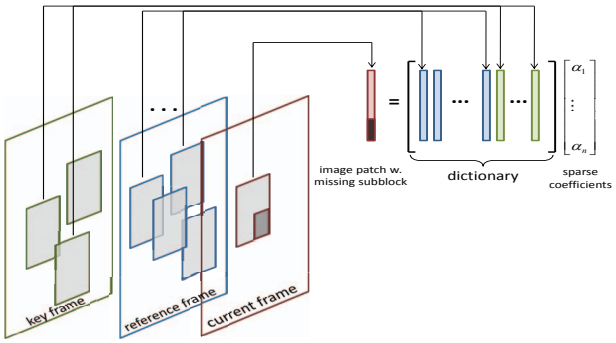


Fig. 1. Image patch represented using local dictionary

forming the image patch and building the corresponding local dictionary don't require any knowledge of MV or confine us to the grid of MB defined by the encoder. In fact, any image patch of any arbitrary size (that does not invalidate the assumption of motion compensation) can have such representation. This gives us complete freedom in deciding the grid and the block size in a frame at the decoder side and makes our proposed method completely independent of the technologies used in the encoder.

3.2. Sparse recovery of a partially corrupted block

One important assumption for the method to work is the locations of the lost blocks are known. We then slice the lost areas into sub-blocks of smaller size. Concealment is done for each sub-block one-by-one. For each sub-block, an image patch is formed using that sub-block and its surrounding non-missing (or already concealed) sub-blocks. The surrounding sub-blocks can be selected in such a way that the amount of clean image data is maximized, see Figure 2. Once the missing locations are known, without loss of generality, we can write $p = [p_1,\ p_2]^T$ where p_1 is the clean (or concealed) data, and p_2 is the missing part. Equation (1) can now be rewritten as

$$\begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} D_1 \\ D_2 \end{bmatrix} \alpha + \epsilon \tag{2}$$

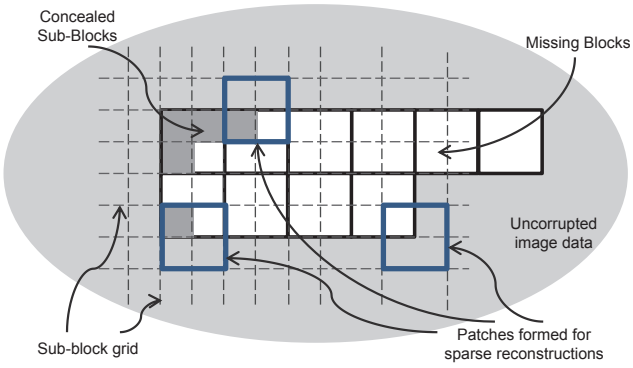


Fig. 2. Grid of sub-blocks and forming of partially corrupted image patches

Where D_1, D_2 are sub-directories associated with p_1 and p_2 respectively. p_1 can also be seen as the partial observation of the model (1). Based on this information, we can recover the α as in (P3)

$$\alpha^* = \text{argmin}\{\tfrac{1}{2}\|p_1 - D_1\alpha\|^2 + \tau\|\alpha\|_1\} \qquad \text{(P4)}$$

The missing sub-block p_2 can then be recovered using α^* and D_2

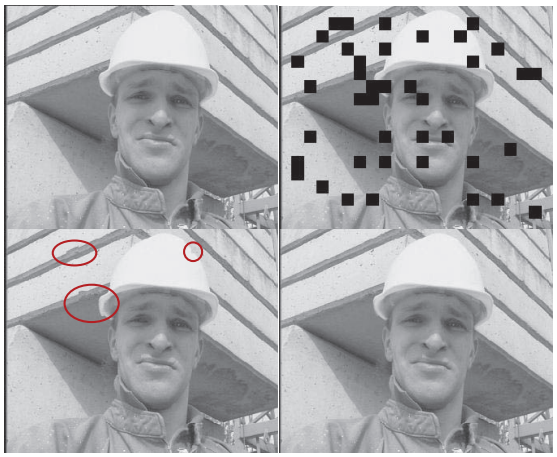
$$p_2 = D_2\alpha^* \qquad (3)$$

4. EXPERIMENTAL RESULTS

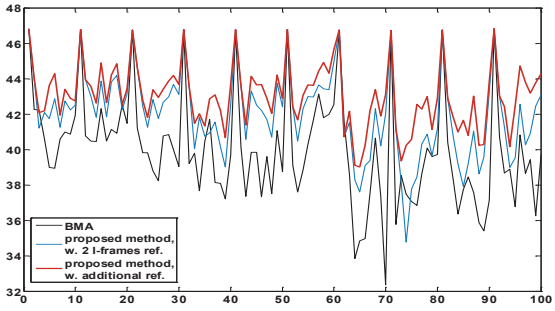
Both the proposed method and BMA are implemented in Matlab for experiments. To have fair comparisons, we based on very similar assumption BMA made: MB based, motion compensated coding. Though BMA has a simple mechanism for Intra-frame concealment in the key frames (I-frames), we only focus on Inter-frame concealment. Both methods are applied on motion compensated frames. I-frames, if exist, are assumed well protected (having no lost MB), but still suffer from quantization error. The quantization effect at the decoding side is simulated by transforming the original sequence to DCT domain, quantizing with certain step size, then transforming back to the image domain. We set up test both methods on two common schemes: with and without clean key frames. The first 100 frames of 'foreman' sequence are used in experiments. To solve (P4), we use a quadratic optimization algorithm based on Linear Complementary Programming, which is extremly simple and suitable for small scale local dictionaries. However, any sparse reconstruction package can be used as well.

In the first scenario (Figure 3) we assume the key frames exist at every 10 frames in the sequence. The quatization step size in DCT domain is 4, that explains why all key frames at the decoder have PSNRs around 46.7 dB. The concealments therefore done only in motion compensated frames (B-frame) between keyframes. MV search is done at the decoder, and a MV in a B-frame can refer to either closest I-frames of the current index. MB size is 16x16. Each B-frame suffers from 10% of MB loss which happens at random places.

Since the encoder is allowed to search for MV in both nearest I-frames, the BMA method is considered using 2 reference frames for each reconstruction. The first set up for our method is to use the same 2 I-frames as reference frames to build local dictionaries. But while BMA can not use more reference frame since this is already decided by the encoder, our method has no such limitation. In another experiment, we add the previous frame (which is already concealed earlier) as another reference frame. The resulted improvement is 3.1 dB on average over the whole test sequence (Figure 3), which shows the potential of the method.



(a) frame #46; first row: original frame and missing pattern; second row: concealed frames by BMA and proposed method

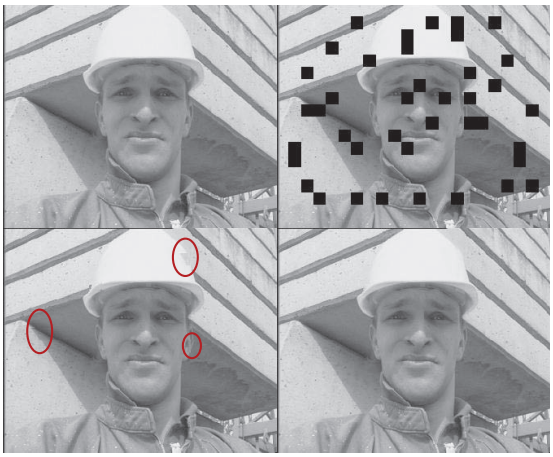


(b) PSNRs of 100 reconstructed frames of Foreman sequence

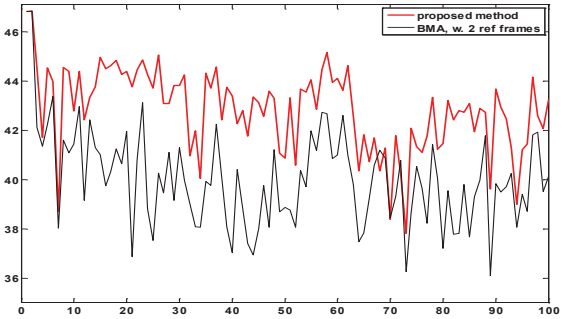
Fig. 3. Experiments on a sequence with key frames.

It is sometimes difficult to justify the existence of clean I-frames. For example, in a network like the Internet where every packet basically has the same protection, it is hard to perfectly protect the I-frames. Therefore, the second scenario deals with the case without I-frames. At each frame, both

BMA and proposed method used two previous frames as reference. For the sake of this experiment, the first two frames of the video are kept uncorrupted to avoid any block error at the beginning propagates through the whole sequence. Missing patterns of MBs are regenerated at every frame (no longer excluding I-frames). Figure 4 displays the results. On average, the proposed algorithm gains 2.8 dB over BMA concealment.



(a) results on frame #54. The ovals mark some mismatches of MBs concealed by BMA



(b) PSNRs of 100 reconstructed Foreman's frames

Fig. 4. Experiments on a sequence without key frames.

5. CONCLUSION AND DISCUSSIONS

In this paper we proposed a simple but powerful approach for video error concealment. The method work as a post-processing technique at the receiver side in the video communication system, and is completely independent from the sender/encoder. Unlike most of the methods for the same video concealment setup, our method does not try on recovering the lost MV, but rather bases on the sparse representation of image patches on local dictionaries.

There is still plenty of room for further improvements. For example, using various sub-block size, shifting the sub-block grid and averaging results would reduce the blocking effect if any. It is also possible to incorporate the refinement step as in some competivive methods to improve the visual quality of a reconstruced frame based on some spatial smoothness conditions. Our dictionaries so far contains only temporal neighbouring patches. Including patches in the current frame (spatial neighbourhood) would improve the recovery of blocks in texture area. Building more selective dictionaries which contains less redundancy would also help on the robustness of the recovery process as well as reduce execution time.

This method is developed in the video concealment setup, but in future research, we are going to apply the approach to problems in image/video inpainting or completion [7]: image/video restoration, video removal to name a few.

6. REFERENCES

[1] B.W. Wah, X. Su, and D. Lin, "A survey of error-concealment schemes for real-time audio and video transmissions over the internet," in *Proc. Mult. Soft. Eng., Int. Symp. on*. IEEE, Dec. 2000, pp. 17–24.

[2] S. Aign and K. Fazel, "Temporal and spatial error concealment techniques for hierarchical mpeg-2 video codec," in *ICC '95 Seattle*. IEEE, Jun. 1995, vol. 3, pp. 1778 – 1783.

[3] Ye-Kui Wang, M. M. Hannuksela, V. Varsa, A. Hourunranta, and M. Gabbouj, "The error concealment feature in the h.26l test model," in *Proc. Image Processing. Int. Conf. on*. IEEE, Dec. 2002, vol. II, pp. 729 – 732.

[4] Y. Chen, Y. Hu, O.C. Au, H. Li, and C. W. Chen, "Video error concealment using spatio-temporal boundary matching and partial differential equation," *Multimedia, IEEE Trans. on*, vol. 10, no. 1, pp. 2 – 15, Jan. 2008.

[5] W. Lie and Z. Gao, "Video error concealment by integrating greedy suboptimization and kalman filtering techniques," *Cir. and Sys. for Video Tech., IEEE Trans. on*, vol. 16, no. 8, pp. 982 – 992, Aug. 2006.

[6] E.J. Candes and M.B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21 – 30, Mar. 2008.

[7] Y. Wexler, E. Shechtman, and M. Irani, "Video error concealment by integrating greedy suboptimization and kalman filtering techniques," *Pattern Anal. Mach. Intell., IEEE Trans. on*, vol. 29, no. 3, pp. 463 – 476, Mar. 2007.