FULL OCCLUSION MANAGEMENT FOR WAVELET-BASED VIDEO CODING

Thomas André, Marc Antonini, Michel Barlaud

I3S Laboratory, UMR 6070 of CNRS, University of Nice-Sophia Antipolis
Bât. Algorithmes/Euclide, 2000 route des Lucioles - BP 121 - 06903 Sophia Antipolis Cedex, France
Phone: 33(0)4.92.94.27.21 - Fax: 33(0)4.92.94.28.98 - {andret,am,barlaud}@i3s.unice.fr

ABSTRACT

Motion-compensated lifting schemes have become a reference for the temporal filtering of video data. However, block-based motion estimation and compensation produce annoying blocking artifacts around the moving objects and near the borders of the images. In this paper, we propose a new lifted temporal filtering method, based on joint segmentation and motion estimation. This method consists in attributing locally the motion information to content-adapted regions instead of blocks. We first present our "Puzzle filtering" algorithm and we state the conditions for its invertibility. Then, we propose a method to extract regions of occlusion from the motion and segmentation information. The obtained regions are finally exploited within the proposed Puzzle filtering. First experimental results show that the blocking artifacts are completely removed and that the occlusions are successfully managed, which results in an important subband entropy decrease.

1. INTRODUCTION

Video compression has become a critical application over the last years, due to the development of digital cinema, high-definition television and internet-related applications. Important research works led to very efficient norms and algorithms, such as the hybrid coders MPEG4 and H.264/AVC. Wavelet-based video coders [1, 2] with motion-compensated $t+2D$ lifting schemes [4] brought new useful features such as scalability [5, 6, 7], while remaining close to hybrid coders in terms of coding efficiency [8].

Nevertheless, there is still room for improvement. In particular, most of the previously mentioned video coders make use of block-based motion estimation in order to improve the efficiency of the temporal filtering. Since the subdivision into blocks does not match the positions of the moving objects, some blocks overlap regions with different motion, which creates blocking artifacts in the coded-decoded sequence. Additionally, the existing coders do not manage occlusions. An occlusion management should help to improve the coding gain and the visual quality of the occluded zones, that are usually very badly rendered and characterized by flutter around the moving objects.

The MPEG-4 norm describes an object-based video coding [9], but this feature has not been much developed, mostly because high-precision semantic segmentation methods such as [10] are still not fully automatic. However, recent works [11] show that hybrid object-based video coding is competitive.

In this paper, we present a temporal filtering method that makes use of joint segmentation-motion estimation information. We propose a simplified approach, which doesn't require the segmentation information to match the semantic segmentation into background and objects. Indeed, we suppose that there are only two regions, which we call arbitrarily the "object" and the "background"; this condition must only be verified locally, which is a reasonable assumption. With different motion-compensated lifted wavelet transforms applied onto specific regions, occlusions are successfully managed, and the flutter and blocking artifacts located at the border of the moving objects are removed.

The paper is organized as follows. First, we describe in section 2 a new lifting-based temporal filtering method, which we call "Puzzle temporal lifting", able to filter several regions with different motions, and we discuss the invertibility of the proposed scheme. Then, we define in section 3 the different regions that must be characterized in order to remove the blocking artifacts and to manage the occlusions. We also propose an algorithm to extract these regions from the motion and segmentation information. Finally, in section 4, we combine the presented algorithms and we show promising first experimental results.

2. PUZZLE TEMPORAL LIFTING

In this section, we present a temporal lifting-based filtering, able to process several regions differently. We first present the algorithm in the simple case where only two different regions are distinguished in the image.

2.1 Case of two different regions

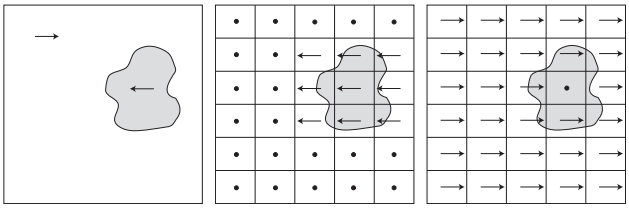
2.1.1 Problem statement and notations

We assume that we know the segmentation of the images into two regions: an arbitrary "object", and a "background". We stress that the proposed algorithm does not require this segmentation to be semantic. We also suppose that we know the respective motion of the two segmented regions. All this information can be obtained by a joint segmentation-motion estimation algorithm based on the work presented in [10], or by a manual segmentation (which is used, for example, in digital cinema post-processing) followed by a region-matching motion estimator. The objective here is to filter the "object" and the "background" independently, using their respective motion information. The motion-compensated $(2,2)$ lifting scheme, which corresponds to the popular 5/3 transversal wavelet transform, will be used to filter both regions.

The figure 1a shows an example of a simple sequence (I_i), where a rigid object translates over a moving background. The motion estimator divides each image into blocks. In each block, it first distinguishes an object and a background using the segmentation information. Then, it determines the respective motion of the object and the background, using any classical block-matching algorithm modified into "region-matching": as a result, it computes one motion vector per block and per region, as shown in Fig. 1b and 1c. Note that the proposed method is valid as long as there is only two different regions *per block*.

Let M_i^{obj} be the mask corresponding to the object in the i^{th} image. For each pixel $m=(x,y)$ in this image, $M_i^{obj}(m)=1$ if p belongs to the object, and $M_i^{obj}(m)=0$ otherwise. We also note M_i^{bg} the mask corresponding to the background in the i^{th} image; thus, we have $M_i^{bg}=\overline{M_i^{obj}}$.

Let us also denote by $v_{i\to j}^{obj}(b)$ (resp. $v_{i\to j}^{bg}(b)$) the vector representing the motion of the object (resp. the background) in the block b, from the image i to the image j. These motion vectors are used to motion-compensate images or masks; for example, let $I_{2i-1}\left(v_{2i-1\to 2i}^{bg}\right)$ be the $2i-1^{th}$ image, motion-compensated using the motion vectors $v_{2i-1\to 2i}^{bg}$.



(a) Object mask (gray) (b) Object motion (c) Background motion

Figure 1: Example of a simple sequence where a rigid object (gray) translates over a moving background. There is one motion vector per block for the object, and one per block for the background (arrows). Dots stand for null vectors.

2.1.2 Analysis

The input video sequence (I_i) can be filtered using either the object's motion information, or the background motion information. The computation of the high-frequency subbands in each case yields:

$$H_i^{bg} = I_{2i} - \frac{1}{2}\left[I_{2i-1}\left(v^{bg}_{2i-1\to 2i}\right) + I_{2i+1}\left(v^{bg}_{2i+1\to 2i}\right)\right] \quad (1)$$

$$H_i^{obj} = I_{2i} - \frac{1}{2}\left[I_{2i-1}\left(v^{obj}_{2i-1\to 2i}\right) + I_{2i+1}\left(v^{obj}_{2i+1\to 2i}\right)\right] \quad (2)$$

Since we would like to filter the object using the object's motion, and the background using the background's motion, the global high-frequency subband H_i should be computed as follows:

$$H_i = H_i^{bg} * M_{2i}^{bg} + H_i^{obj} * M_{2i}^{obj} \quad (3)$$

where the operator $*$ stands for a term-to-term multiplication. In other words, for each pixel p, $H_i(p)$ is locally either equal to $H_i^{bg}(p)$ or to $H_i^{obj}(p)$, because M_i^{bg} is the complement of M_i^{obj} in the image.

Similarly, the low-pass subband L_i is computed as follows:

$$L_i = L_i^{bg} * M_{2i-1}^{bg} + L_{2i-1}^{obj} * M_i^{obj} \quad (4)$$

where L_i^{bg} and L_i^{obj} are computed as regular $(2,2)$-lifting update steps.

2.1.3 Synthesis

Let us now discuss the invertibility of the proposed transform. By inverting the lifting scheme from the previous equations, we obtain:

$$I_{2i-1} * \left[M_{2i-1}^{bg} + M_{2i-1}^{obj}\right] = L_i + f_L(H_{i-1}, H_i) \quad (5)$$

$$I_{2i} * \left[M_{2i}^{bg} + M_{2i}^{obj}\right] = H_i + f_H(I_{2i-1}, I_{2i+1}) \quad (6)$$

where the expressions $f_L(H_{i-1}, H_i)$ and $f_H(I_{2i-1}, I_{2i+1})$ do not depend on the image being reconstructed. By definition, for each image j, we have $M_j^{bg} = \overline{M_j^{obj}}$, and the reunion of these masks covers the entire surface of the image. It follows that

$$I_j * \left[M_j^{bg} + M_j^{obj}\right] = I_j \quad (7)$$

which proves the invertibility of the proposed scheme.

2.2 Generalization to N regions

Let us suppose now that we have characterized N different regions in each image, with different motions, and that we need to apply a specific filtering to each one of them. We note M_i^n the mask corresponding to the region n of the image i, and $v^n_{i\to j}$ the motion vector of the region n from the image i to the image j.

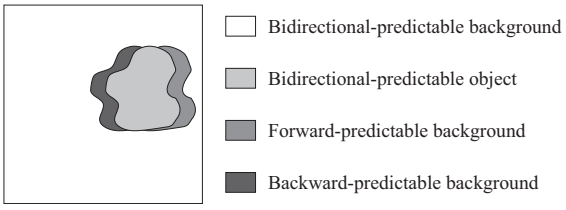


Figure 2: Regions of occlusion for the example of Fig. 1.

According to the previous discussion, the low-pass and high-pass subbands can be computed as follows:

$$H_i = \sum_{n=1..N} H_i^n * M_{2i}^n \quad \text{and} \quad L_i = \sum_{n=1..N} L_i^n * M_{2i-1}^n$$

and it can be easily shown that the proposed scheme is invertible if the reunion of the masks $(M_i^n)_{n=1..N}$ covers the entire image, and if their intersection is empty.

3. PUZZLE WITH OCCLUSIONS MANAGEMENT

In this section, we focus on the different regions of the images that should be used within the proposed framework in order to eliminate the blocking artifacts and to manage the occlusions. We first determine these regions and the corresponding sets of filters. Then, we describe an algorithm which computes these regions from the joint segmentation-motion information that we have.

3.1 Regions and filters for occlusion management

Let us suppose that we want to filter the image i using the $(2,2)$ transform, and take the occlusions into account. This operation requires the knowledge of several regions of the image. Indeed, the background must be divided into three areas: the area invisible in the image $i-1$ (thus predictable from the image $i+1$ only), the area invisible in the image $i+1$ (thus predictable from the image $i-1$ only), and the area which is not occluded (thus predictable from both directions). Similarly, the object is divided into three areas.

We obtain six different regions in the image, each one represented by its mask; for example, let $M_i^{bg,BW}$ be the background part (bg) of the image i which is predictable from the backward image only (BW). The figure 2 shows the decomposition into regions for the example of the figure 1; in this example, the object is rigid and does not disappear behind anything or outside the image, thus it is fully bidirectional-predictable.

Let us now describe the filtering of the background region. The non-occluded areas of the background can be processed using the regular bidirectional $(2,2)$ filters. For example, the high-pass subband of the bidirectional-predictable area is computed as follows:

$$H_i^{bg,bidir} = I_{2i} - \frac{1}{2}\left[I_{2i-1}\left(v^{bg}_{2i-1\to 2i}\right) + I_{2i+1}\left(v^{bg}_{2i+1\to 2i}\right)\right]$$

Since the occluded areas must be predicted from one direction only, the Haar filter is well-adapted. The high-pass subbands of the backward-predictable and (resp.) forward-predictable areas are computed as follows:

$$H_i^{bg,BW} = I_{2i} - I_{2i-1}\left(v^{bg}_{2i-1\to 2i}\right) \quad (8)$$

$$H_i^{bg,FW} = I_{2i} - I_{2i+1}\left(v^{bg}_{2i+1\to 2i}\right) \quad (9)$$

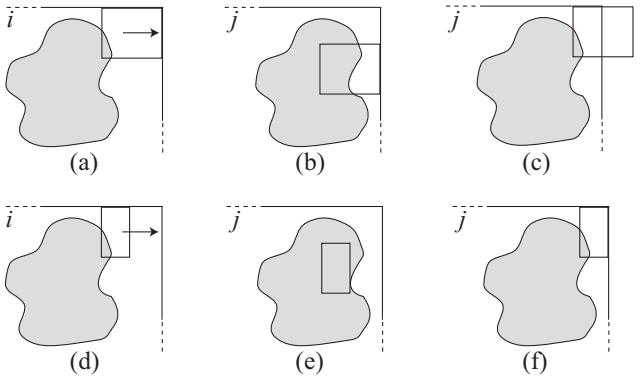


Figure 3: Algorithm for the detection of occluded borders between images i and j. A classical block-matching algorithm is unable to find the correct match for the block (a), and points to a wrong block (b), because the correct block (c) is not located inside the target image j. The proposed algorithm detects these outliers (d,e,f).

Finally, the expression of the high-pass subband of the background area is:

$$\begin{aligned} H_i^{bg} &= H_i^{bg,bidir} * M_i^{bg,bidir} \\ &\quad + H_i^{bg,BW} * M_i^{bg,BW} + H_i^{bg,FW} * M_i^{bg,FW} \end{aligned} \tag{10}$$

The processing of the object areas is performed similarly.

3.2 Region extraction

We have determined the appropriate filters for each one of the six required regions of the images. In the following, we describe an algorithm that extracts these six regions from the information we have, which is the motion information and the segmentation masks. We will denote by C the central image to be filtered, and by B (resp. F) its backward (resp. forward) neighbor.

First, let us extract the forward-predictable area of the background, represented by the mask $M_C^{bg,FW}$. This is the area of the background which is visible in C but not in B. In other words, it corresponds to the background of C, from which the motion-compensated background of B is substracted (operator '$\setminus$'):

$$M_C^{bg,FW} = M_C^{bg} \setminus M_B^{bg}\left(v_{B\to C}^{bg}\right) \tag{11}$$

Similarly, the backward-predictable area of the background is obtained as follows:

$$M_C^{bg,BW} = M_C^{bg} \setminus M_F^{bg}\left(v_{F\to C}^{bg}\right) \tag{12}$$

One can remark that there might be regions which are not predictable at all; these regions are present in both $M_C^{bg,FW}$ and $M_C^{bg,BW}$. Since it is important for the masks to be complementary, we must modify them as follows:

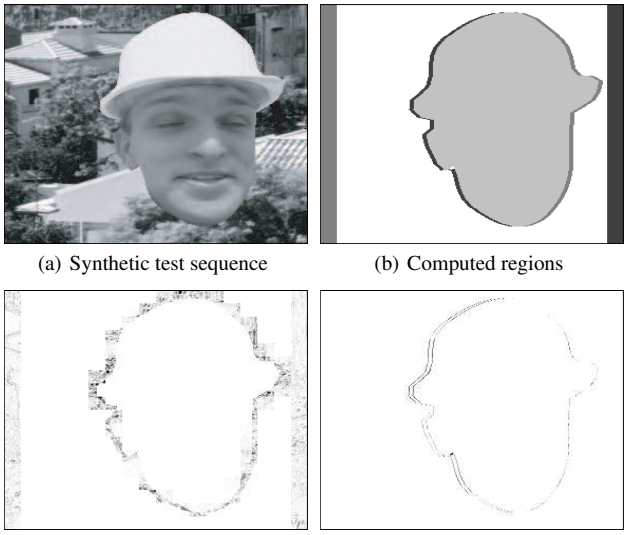
$$M_C^{bg,FW} \leftarrow M_C^{bg,FW} \setminus M_C^{bg,BW} \tag{13}$$

$$M_C^{bg,BW} \leftarrow M_C^{bg,BW} \setminus M_C^{bg,FW} \tag{14}$$

Finally, we set:

$$M_C^{bg,bidir} = M_C^{bg} \setminus \left(M_C^{bg,FW} \cup M_C^{bg,BW}\right) \tag{15}$$

which defines the region which will be filtered in both directions. This region includes the bidirectional-predictable area and the non-predictable area of the background.



(a) Synthetic test sequence (b) Computed regions

(c) Classical filtering (d) Puzzle filtering

Figure 4: Synthetic test sequence (a), and example of the regions obtained using the proposed method (b), with the same color codes as in Fig. 2. (c) and (d): first image of the high-frequency subband, obtained by the filtering of the synthetic test sequence. Higher absolute values are represented by darker pixels.

The three objects areas, namely $M_C^{bg,FW}$, $M_C^{bg,BW}$ and $M_C^{bg,bidir}$, can be obtained in a similar way. By construction, these six computed areas satisfy the conditions of invertibility of the scheme stated in the previous section.

Note that the "background" and the "object" play a symmetrical role in the whole process. They just define two different local regions that do not have to match the "real", semantic objects and background.

3.3 Borders management

The border of the images are often very badly rendered because they tend to disappear from an image to another as soon as the camera is moving (Fig. 3a,c). The motion of the corresponding blocks is thus badly estimated (Fig. 3b). In the proposed Puzzle framework, we could consider them as occluded regions, and thus completely remove the resulting blocking artifacts. However, the previous region extraction method is unable to mark those regions as occluded, because their estimated motion vector doesn't point outside the image (Fig. 3b).

To overcome this problem, we propose a simple test of motion invertibility localized at the borders of the frames. For each border block, we compare the prediction error obtained using the estimated motion vector, with the one obtained using the inverse motion vector. More precisely, we propose the following algorithm. For each border block B:

1. check if $-v_{j\to i}(B)$ points outside the image (Fig. 3c); if not, no occlusion is detected.
2. determine B', the part of B that would still be located inside the image if translated by $-v_{j\to i}(B)$ (Fig. 3d).
3. compare $B'_{ij} = v_{i\to j}(B')$ (Fig. 3e) and $B'_{\_ji} = -v_{j\to i}(B')$ (Fig. 3f): if $MSE(B', B'_{\_ji}) < MSE(B', B'_{ij})$, then mark B as occluded.

4. EXPERIMENTAL RESULTS

In this section, we present first experimental results for a synthetic test sequence similar to the one presented in Fig. 1. The object is a rigid textured head, and it moves over a textured background

(a) Classical filtering (b) Puzzle filtering

Figure 5: Filtering of the test sequence "Erik" using both methods. Higher absolute values are represented by darker pixels.

(Fig. 4a). The motion was estimated using a classical 16×16 block-matching algorithm. The segmentation was determined by hand, and it presented an error of one pixel all around the object, which is a common error due to the precision of segmentation techniques.

We compared the subbands computed by a classical lifting scheme and by the proposed Puzzle filtering, for two sets of filters: the $(2,2)$ filters, and the $(2,0)$ filters which correspond to the truncated 5/3 wavelet transform [12]. An example of the regions obtained by the algorithm proposed in 3.2 is shown in Fig. 4; examples of the obtained subbands are also shown. In spite of the subsisting prediction errors, due to the pixel-accuracy of the masks, it appears that the proposed method produces much cleaner subbands than the classical temporal filtering, with much lower entropy and energy, as shown in table 1. The localization and the regularity of the remaining prediction errors should allow a much more efficient subband encoding than the classical method. The masks, whose entropy does obviously not exceed 1 bpp, should also be efficiently encoded using context-based or Zero Run-Length coders.

Set of filters		$(2,2)$		$(2,0)$	
Algorithm		BBF	PF	BBF	PF
Energy	HF $(.10^{-3})$	16.0	**1.0**	16.0	**1.0**
	LF $(.10^{4})$	1.63	1.53	1.53	
Entropy (bpp)	HF	2.13	**0.17**	2.13	**0.17**
	LF	8,92	7.75	7.60	

Table 1: Comparison of the classical block-based filtering (BBF) method and the proposed Puzzle filtering (PF), on the synthetic test sequence, for two sets of filters: average normalized energy and entropy of the high-frequency (HF) and low-frequency (LF) subbands.

We also tested the proposed filtering method on the real sequence "Erik". The Puzzle filtering was performed using 32×32 macroblocks. The automatic segmentation into regions and the motion estimation was obtained using [13]. The classical filtering method was improved by dividing each overlapping 32×32 macroblocks into 4 smaller blocks, in order to obtain a fair comparison.

We noted that the regularity of the segmentation over time had a significant impact on the performances of the occlusion detection algorithm, and thus on the prediction error. However, despite a few local imprecisions of the segmentation, the Puzzle filtering with occlusion management successfully removed most of the blocking artifacts around the moving character. Finally, although the resulting HF subbands have similar energy and entropy to those obtained using a classical filtering, the Puzzle subbands are much cleaner, as shown in Fig. 5.

5. CONCLUSION

We have presented a new lifting-based temporal filtering method, called Puzzle filtering, that makes use of joint segmentation-motion estimation algorithms in order to remove blocking artifacts. First results show an important entropy decrease in the subbands, and confirm the removal of blocking artifacts. Future works will concern the inclusion of the proposed technique into the scalable wavelet-based video coder presented in [8].

REFERENCES

[1] J.-R. Ohm, "Three Dimensional Subband Coding with Motion Compensation," *IEEE Trans. on Image Processing*, vol. 3(5), pp. 559–571, Sep. 1994.

[2] S.J. Choi and J.W. Woods, "Motion-compensated 3-D Subband Coding of Video," *IEEE Trans. on Image Processing*, vol. 8(2), pp. 155–167, Feb. 1999.

[3] A. Secker and D. Taubman, "Motion-compensated Highly Scalable Video Compression Using an Adaptive 3D Wavelet Transform Based on Lifting," in *Proc. of IEEE Intern. Conf. on Image Processing*, Thessaloniki, Greece, Oct. 2001, pp. 1029–1032.

[4] J. Viéron, C. Guillemot and S. Pateux, "Motion compensated 2D+t wavelet analysis for low rate FGS video compression," in *Proc. of Tyrrhenian Intern. Workshop on Digital Comm.*, Capri, Italy, Sept. 2002.

[5] A. Secker and D. Taubman, "Lifting-Based invertible motion adaptive transform (LIMAT) framework for highly scalable video compression," in *IEEE Transaction on Image Processing*, vol. 12(12), pp. 1530-1542, Dec. 2003.

[6] N. Cammas and S. Pateux, "Fine grain scalable video coding using 3D wavelets and active meshes," in *SPIE Visual Communications and Image Processing, VCIP 2003*, Jan. 2003.

[7] G. Pau, C. Tillier, B. Pesquet-Popescu and H. Heijmans, "Motion Compensation and Scalability in Lifting-Based Video Coding," *EURASIP Signal Processing: Image Communication, special issue on Wavelet Video Coding*, pp. 577–600, Aug. 2004.

[8] M. Cagnazzo, T. André, M. Antonini and M. Barlaud, "A model-based motion compensated video coder with JPEG2000 compatibility," in *Proc. of IEEE Intern. Conf. on Image Processing*, Singapore, Oct. 2004, pp. 2255–2258.

[9] MPEG4 Video Group, *Coding of audio-visual objects: Video*, ISO/IEC JTC1/SC29/WG11 N2202, Mar. 1998.

[10] S. Jehan-Besson, M. Barlaud and G. Aubert, "DREAM2S: Deformable Regions driven by an Eulerian Accurate Minimization Method for image and video segmentation," *International Journal of Computer Vision*, vol. 53(1), pp. 45–70, 2003.

[11] M. Chaumont, S. Pateux and H. Nicolas, "Object-based video coding using a dynamic coding approach," in *Proc. of IEEE Intern. Conf. on Image Processing*, Singapore, Oct. 2004, pp. 377–380.

[12] T. André, M. Cagnazzo, M. Antonini, M. Barlaud, N. Božinović and J. Konrad, "(N,0) motion-compensated lifting-based wavelet transform," in *Proc. of IEEE Intern. Conf. on Acoustics, Speech and Signal Processing*, Montreal, Canada, May 2004, pp. 121–124.

[13] S. Boltz, E. Debreuve and M. Barlaud, "A joint motion segmentation algorithm for video coding," in *Proceedings of EUSIPCO*, Antalya, Turkey, Sept. 2005 (to appear).