Signal Processing: *Image Communication* 27 (2012) 153–171

Contents lists available at SciVerse ScienceDirect

Signal Processing: *Image Communication*

journal homepage: www.elsevier.com/locate/image

Motion compensation based on tangent distance prediction for video compression

Jonathan Fabrizio [a,*], Séverine Dubuisson [b], Dominique Béréziat [b]

[a] LRDE-EPITA, 14-16, rue Voltaire, F-94276 Le Kremlin Bicêtre Cedex, France
[b] Université Pierre et Marie Curie, Laboratoire d'Informatique de Paris 6, 4 place Jussieu, Paris 75252 Cedex, France

ARTICLE INFO

Article history:
Received 17 December 2010
Accepted 3 December 2011
Available online 16 December 2011

Keywords:
Video compression
Motion compensation
Tangent distance
Theora

ABSTRACT

We present a new algorithm for motion compensation that uses a motion estimation method based on tangent distance. The method is compared with a Block-Matching based approach in various common situations. Whereas Block-Matching algorithms usually only predict positions of blocks over time, our method also predicts the evolution of pixels into these blocks. The prediction error is then drastically decreased. The method is implemented into the Theora codec proving that this algorithm improves the video codec performances.

1. Introduction

Video compression refers to reducing the quantity of data used to represent video images. A video is a sequence of frames (or images) that are related along the temporal and spatial dimensions: two consecutive frames might be similar, and the only observed changes are supposed to be due either to the displacements of objects or the camera, to the changes of illumination, or to the noise. In order to reduce the amount of data in image sequences to be transmitted, it is then necessary to determine the spatio-temporal redundancies and to exploit it by defining predictable properties. Considering the data to encode, these properties are used to make predictions, and only the errors between original and predicted data are sent. This technique by itself does not reduce the amount of data (for video compression, we transmit an image of errors, that contains as many pixels as the original image), but, combined with a statistical entropy coding, reduces the data size. In fact, these errors have smaller dynamics than the original pixel values, giving a smaller entropy, resulting in a diminution of the number of bits necessary to encode data.

We distinguish two main prediction types: the temporal prediction and the spatial one.

Spatial, or intraframe prediction, only uses the current frame information: pixels of the frame buffer, sorted into their raster order, are supposed to be similar. By only considering pixels previously examined (and thus already coded) in a specific neighborhood, the coder predicts the value of the current pixel. The main difficulty in such approaches is the choice of weighting coefficients for the pixels in the neighborhood. Usually, spatial prediction is adapted to the image content (edges, area, etc.). Among the large number of existing spatial predictors, a well known one is the median adaptive predictor (MAP) [24]. MAP selectively uses three linear predictors based on a simple function of surrounding values and gives a good prediction even in the presence of edge features. This predictor has been embedded into the LOCO-I algorithm [37]. Spatial prediction is used in numerous codecs in the spatial domain (H.264/AVC [38,30]) or in the frequency domain (Theora Codec [8]).

Temporal prediction, also called interframe prediction, uses earlier or/and later frames in the sequence to predict

[*] Corresponding author. Tel.: +33 1 53 14 59 40; fax: +33 1 53 14 59 13.
E-mail address: jonathan.fabrizio@lrde.epita.fr (J. Fabrizio).

0923-5965/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.image.2011.12.001

the current frame. Considering a pixel in the current frame, its neighbor pixels in the next and previous frames are assumed to be similar. Motion estimation can be used to eliminate temporal redundancies between frames in a video sequence [9]. Two kinds of motion estimation approaches are used to perform the prediction: (i) in an unidirectional way (front or back), without taking into account occlusion, appearance or disappearance problems; or (ii) in a bidirectional way, by using both backward and forward information in the sequence. This paper focuses on the issue of temporal prediction.

Block-Matching (BM) algorithms are commonly used for motion estimation in most of MPEG [19] implementations as well as in many other encoders. The idea consists, for a given frame, in partitioning the image domain into non-overlapping blocks (generally square blocks), and then, for each block, in searching in a reference frame the most similar region over an area near the position of the block. The more similar the regions, the lower the prediction error. A similarity criterion is usually defined as either mean square difference, or mean absolute difference. There are lots of BM algorithms in the literature: surveys can be found in [13,5,39], and an interesting empirical comparative study in [20]. Recent works mainly concern the diminution of BM complexity. This is often done by using a specific search technique, such as the most powerful one, called the diamond search [40], or by using an adaptive modeling of blocks. We can also adopt a coarse-to-fine strategy, such as the multi-level approach proposed in [11], where outliers areas are progressively eliminated. In [27], the authors propose to use patterns for motion vector estimation whose size is adapted depending on the context. In [6] a geometry-adaptive block partitioning is used, and a very recent improved version is proposed in [12] that only seeks for a limited number of partitions.

The mesh-based motion model, also called grid wrapping, provides improved interpolation accuracy compared with block-based motion models when the motion field varies smoothly in the spatial domain. The variation of the mesh topology as well as the strategy for coding the synthesis error are defined by an optimization technique following the rate-distortion criterion. The motion is generally modeled by the displacements of the mesh nodes, so that the amount of motion information to be transmitted remains small [2,35]. In [18] the use of a mesh-based motion compensated interpolation is shown to give better results than a simple BM. Active meshes [26] can also efficiently represent and code the various regions of the scene and the motion information are also used in temporal prediction. However, they often fail at solving the problem of motion discontinuities (in particular the cases of small objects or occlusions). Recent works try to overcome this problem by refining, as a post-processing step, the mesh node positions when their surrounding patches contain motion edges [15]. In [28], the authors design specific interpolation kernels for mesh-based motion estimation that permit to integrate BM motion vectors into a mesh. In [4] node-point motion vectors of a triangular mesh-based model are estimated using a hierarchical hexagonal refinement algorithm.

Other kinds of approaches used for motion estimation are gradient-based optical flow methods [14]. As they provide a dense vector field which is inefficient for the video compression issue, it is then necessary to reduce the velocity field size and to consider a parametric velocity model, generally chosen constant [21] or affine [3]. The optical flow estimation is then reduced to an over-constrained linear problem that can easily be computed using least square method for instance. Parametric optical flow has been successfully applied to video compression [16,1]. Optical flow as well as BM methods make the hypothesis of a luminance constancy between two consecutive frames, that is, in general, not true in video sequences. To be robust to luminance changes, an adequate luminance model, i.e. a model describing the luminance evolution between two consecutive frames, must then be defined. For example, supposing a constant variation, or an affine one, significantly improves the prediction accuracy and then the compression bit rate. The luminance model can be global [17] over the image domain or locally defined for each block [36,31]. If robustness to luminance changes has been earlier introduced for the determination of the optical flow [29], it was coupled to BM methods only for the issue of video compression [36,17,31].

We have previously proposed in [7] a new motion estimation technique. The purpose of this article is to prove that this contribution improves the motion compensation step if included into a compression scheme: we offer a new approach for motion compensation using a temporal predictor. We revisit the BM algorithm and change the way blocks are matched: we substitute the classic mean square or absolute difference for the tangent distance. Classical BM algorithms only predict block positions over time. By the use of the tangent distance, we not only estimate block positions over time, but also the affine evolution of pixels into these blocks. We demonstrate that tangent distance is equivalent to an affine parametric optical flow method: our method then takes advantages of both approaches (BM algorithm and optical flow). It is also the opportunity to theoretically introduce a luminance model in tangent distance that is robust to local and constant luminance changes.

The organization of the article is as follows. In Section 2, we present video compression by motion compensation using the tangent distance approach. After introducing the tangent distance concept, we show how it can be embedded into a motion compensation framework, justify our choices and show links with other approaches. We then expose the encoding and decoding schemes. Qualitative and quantitative results are given in Section 3. First, Sections 3.1 and 3.2 specifically focus on the motion compensation step. The prediction robustness is also evaluated in Section 3.3. Then, a complete compression scheme is presented in Section 3.4: we have embedded our method and BM into Theora codec, and compared the coding performances of these two approaches on eight standard video sequences. Concluding remarks and perspectives are finally given in Section 4.

2. Video compression by motion compensation using tangent distance

Motion compensation is split into two main parts: an encoding and a decoding step. During the encoding step,