SEGMENTATION-BASED MORPHOLOGICAL INTERPOLATION OF PARTITION SEQUENCES *

R. BRÉMOND and F. MARQUÉS
Dept. of Signal Theory and Communications
Universitat Politècnica de Catalunya
Campus Nord - Módulo D5
C/ Gran Capità, 08034 Barcelona, Spain
Tel: (343) 401 64 50, Fax: (343) 401 64 47
E-mail: ferran@gps.tsc.upc.es

Abstract. A new algorithm for partition sequence interpolation is proposed. In a coding context, such a tool is necessary to reach high compression rates. Our scheme relies on a region-by-region approach. We propose a region ordering, based on an error criterion. Before ordering, some regions are merged according to a motion criterion. The shape of each region changes continuously, and the intermediate images are built from the new shaped regions with a dead leave model.

Key words: segmentation-based image coding, image interpolation, Hausdorff distance.

1. Introduction

Image interpolation turns out to be a point of increasing interest in the field of image sequence coding. Interpolation techniques allow to sample, in the encoder side, the image sequence and to transmit only the selected frames. In the decoder side, non-selected images are reconstructed relying on the decoded version of the transmitted images. Interpolation is also a tool of value for old movies enhancement. It allows to get sequences with 25 images/second out of old movies with lower frequency rates.

The key step of all image sequence interpolation techniques is the motion estimation. The underlying idea is to follow the motion of basic entities such as pixels, blocks, or regions, between frames t and $t+p$ of a sequence. Relying on this motion model, intermediate steps of the chosen basic entity can be computed. Regardless of the motion model, interpolation techniques may rise some conflicts. Depending on the relative motions, two entities may cover the same pixel in an intermediate frame. The inverse problem may occur when some pixels are not valuated at all.

The most common motion estimation algorithm is the so-called block-matching [1]. Unfortunately, this technique yields noisy interpolated boundaries (block effect). The overlapping problem, is usually solved by choosing the value which gives the smallest error (in the cost-function sense). For the hole problem, the only solution with these methods consists in an approximation with a continuous 2D physical model, such as bilinear interpolation [1]. This approximation relies on a physical model: the image between times t and $t+p$ is transformed by a planar continuous

* This work has been supported by the Human Capital and Mobility Project ERBCHBICT 941329 (*Image Interpolation based on mathematical morphology. Applications to motion restoration and to sequence coding*) of the European Union

deformation, and this continuity allows to interpolate an uncertainty area with a smooth continuous function. Unfortunately, real sequences are not the result of such continuous transforms.

The development of region-based methods in many fields of image processing makes possible to develop such an approach for sequence interpolation. The region-based approach to interpolation uses a segmentation description of image sequences. As result of the segmentation step, images are split into a set of regions forming a partition. Such regions are characterized by their texture and contours. The segmentation should put in correspondence regions in successive images by tracking them through the time domain [4]. Then, each region is interpolated (its shape and texture separately), and the individual interpolations are combined in order to build an interpolated image. If the segmentation is good enough, each region moves coherently so that the description of the motion of each region is more accurate than with the block-matching.

2. General scheme

I_t and I_{t+p} are the partitions (label images) corresponding to the segmentation of frames t and $t+p$ of a sequence. We propose a general scheme for building the intermediate partitions $I_{t+1} \dots I_{t+p-1}$ with no other input data than the two initial partitions I_t and I_{t+p}. Therefore, this interpolation technique is independent of the type of segmentation process that has been used and, thus, it can be applied to any segmentation-based coding scheme. Let $R_t(i)$ be the region with label i in I_t and $R_{t+p}(i)$ the region with the same label in I_{t+p}.

The proposed scheme copes separately with each region or group of regions (concept of meta-region). The scheme can be split into four steps [3]: Region parametrization, Region ordering, Region interpolation and Partition creation.

- Region parametrization: the evolution of a region from $R_t(i)$ to $R_{t+p}(i)$ is divided into regular motion and shape deformation. These types of evolution are separately modeled. If neighbor regions present similar regular motion, they are merged into a meta-region. Region parametrization is necessary in order to have a representation of the region that can be easily interpolated.
- Region ordering: depth parameters $d(i)$ are computed for each region. They should correspond to the semantic idea of depth: the deepest region should be in the background, the less deep, in the foreground. By means of the region ordering, possible conflicts due to overlapping regions in the interpolated partition can be solved.
- Region interpolation: the parameters that characterize the evolution of the region are interpolated, so that a new set of interpolated regions is obtained.
- Partition creation: interpolated regions are introduced in the interpolated partition following a dead leave model. Some pixels in the interpolated partition may not be assigned to any region and, therefore, holes may appear. Hole conflicts are solved with the help of a propagation model.

This global scheme is shown in Figure 1. Note that it does not assume any particular implementation for each step. In the sequel, an effective solution for each part of the algorithm is presented.

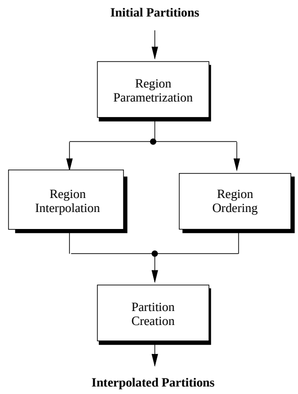


Fig. 1. Global object-based interpolation scheme

3. Region parametrization

3.1. Regular motion estimation

Several models for regular motion can be assumed. The two main types of motion in this application are translation and zooming. The translation of the center of mass $G(i)$ gives a first order approximation of the region motion. However, it may lead to wrong approximations in case of fusion or split of regions. The zoom factor $Z(i)$, due to the camera or to the physical motion, can be computed using the surface ratio between $R_t(i)$ and $R_{t+p}(i)$. However, this zoom parameter raises an uncertainty: It is not clear whether an apparent zoom (computed with this ratio) is due to a real zoom, or to a mask effect (a region hiding another). To solve this uncertainty, we propose to classify the possible situations of a region. There are three main possibilities:

1. The region belongs to the foreground. The real zoom corresponds to the surface ratio, and the translation, to the translation of $G(i)$. The corresponding motion strategy stands on the translation of $G(i)$, and on the zoom factor $Z(i)$.
2. The region belongs to the background. The apparent zoom is wrong, because of a mask effect. The corresponding motion strategy stands on the translation of $G(i)$, without zoom.
3. The region is merged (or split) between frames t and $t + p$. The apparent translation is due to a fusion of several regions into one, or to the splitting of one into several. The motion of $G(i)$ does not convey reliable information. The corresponding *careful* motion strategy is to stay motionless.

For each region $R(i)$, we compute two motion parameters: the translation $\vec{T}(i)$ of $G(i)$, and the zoom factor $Z(i)$. Then, a motion error is computed for each of the three hypotheses listed above. Depending on the hypothesis which results in the smallest error, we assign a motion *type* to the region.

The interpolation is sensitive to the choice of the cost-function. As we work on partitions, we cannot use texture information. Let $C_t(i)$ be the inner contour of $R_t(i)$ and $D_t(i)$ the distance function to the contour of $R_t(i)$:

$$\forall p \in I, [D_t(i)](p) = D(p, C_t(i)) \tag{1}$$

Then, if $R_t(i)$ is transformed into $R'_{t+p}(i)$, we use as cost-function the mean distance between the contour points of the two sets. This is computed as:

$$Cost(R_t \mapsto R'_{t+p}) = \frac{1}{L} \int_{C'_{t+p}} D_t dL \tag{2}$$

where L is the length of $C'_{t+p}(i)$. The key advantage of this function is that it focuses on the object contour. This property avoids the influence of a *surface effect*. Fig. 2 gives an example of this effect with the gate on the left. Since the cost function only relyes on contour information, the interpolation manages to follow the gate motion. This is due to the fact that the cost function is minimum when the bars of I_t and I_{t+p} fit well. Other cost-functions, such as the intersection distance, or the Hausdorff distance [2], would pay too much attention to the plain portion of the region and a wrong motion may be detected: the gate surface would be neglectable compared with any motion of another part of the object. In addition, recent progresses in mathematical morphology allow to compute $D_t(i)$ quickly [7].

Fig. 2. Importance of the contour

3.2. Merging on meta-regions

The merging procedure leads to a high level semantic segmentation. The idea is to recognize the regions which belong to the same physical object. In this work a physical object is defined as a set of neighbor regions sharing a similar motion. Therefore, we compare the motion descriptions of each pair of connected regions, and merge them if they are similar enough and of the same *type*. This merging step allows the relaxation of the motion description of each macro-region.

The merging depends on the motion description. A simple motion model is used, in order to define a *motion solidarity*. The motion model holds a translation vector $\vec{T}(i)$, as well as a zoom qualitative descriptor $z(i)$: for each region $R(i)$, $z(i) = True$ if there is a true zoom, or $z(i) = False$ if there is no zoom, or a mask effect.

We define, then, a parametric merging algorithm. A first *negative* criterion is imposed: regions which have different *types* of motion cannot be merged. Then we define the positive condition of merging: If $R(i)$ and $R(j)$ have a common edge, and are of the same type, they belong to the same macro-region if their motion parameters are similar: $\|\vec{T}(i) - \vec{T}(j)\| < \lambda$, where λ is a parameter of the model.

3.3. Shape deformation

As the previous motion model cannot take into account the real motion of complex objects, a second order model is used. If we compensate the motion of $R_t(i)$ into a region $R'_{t+p}(i)$ in frame $t+k$, and $R_{t+p}(i)$ into $R''_{t+k}(i)$, these two regions do not fit exactly. In order to take into account the continuous deformation from the first one to the second one, we use a morphological algorithm developed by F. Meyer [5]. The idea is to compute the geodesic distance from the first set inside the second one, then from the second set to the first one, and to threshold the difference. The resulting set corresponds to the shape deformation at the intermediary time $t+k$. Two deformation strategies are available. The first one is to use the algorithm for every region. The other consists in modifying Meyer's algorithm in order to cope with several labels at the same time and apply it for each macro-region. Both techniques have been tested. The main problem with the last strategy is that it may propagate errors. When the segmentation step creates regions that belong both to the background and to the foreground (see Fig. 3), labels of different semantic objects propagate more easily into each others. We suggest to use this technique only if it can be ensured that the merging step separates the semantic objects.

Fig. 3. Ambiguous regions in terms of depth

4. Region ordering: Dead leave model

The semantic approach leads to the notion of depths. We have classified each region with a depth value. The interpolated image is built by introducing each region with the following policy: first the deepest regions (the background), then the upper ones, up to the toppest region. The final value of a pixel is the label value of the last region which has covered it. It corresponds to the physical model of *opaque* objects moving and hiding to each other. The importance of the error estimation is that it gives qualitative information about the physical depth. That is, the deeper the region, the more important the motion estimation error.

The example of a synthesis image (Fig. 4) gives some evidences of this property: the ball is in the foreground. Since it is not hidden, the motion estimation is correct. On the contrary, the motion of the squares behind the ball cannot be estimated the same way: even complex motion models would yield bad results, because the apparent motion does not result from a physical motion but from a masking effect.

The depth value which should be assigned to a macro-region is the highest depth value from all its regions. The macro-region consists in one physical object, whose regions have common motion. Then, all the regions of the macro-region should have

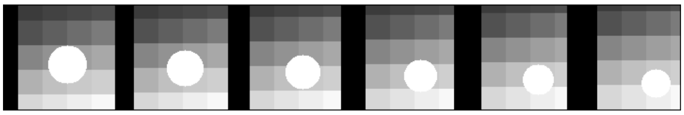


Fig. 4. Synthesis sequence

the same depth. Let us consider a macro-region belonging to the background. It is made of several regions. Some of these regions are partially hidden by objects of the foreground, which results in a high value of the computed depth. Some other regions are well estimated because they are not hidden at all. Their computed depth has a low value. The expected value for the macro-region is the highest one, because the motion error estimation does not indicate the real depth, but an apparent depth. If one part of the macro-region is behind another region, all the macro-regions should be considered deeper than this region.

5. Region interpolation: Using geodesic distances

A morphological approach to partition deformation based on Hausdorff distances has been presented in [5]. In order to compute intermediary steps between two sets, the idea is to compute the geodesic distance from the first set to the second one, then from the second set to the first one, and to threshold the difference at any intermediary time: the resulting set would correspond to the shape deformation at the chosen intermediate time.

6. Partition creation: Propagation of labels

Interpolated regions form the interpolated partition following the ordering obtained in the Region ordering step. After the dead leave modeling of the region positions, some pixels may not belong to any region. Thus, a propagation model is necessary in order to fill such holes with their surrounding labels. In the example of Fig. 4, after the dead leave step, there is the following situation:

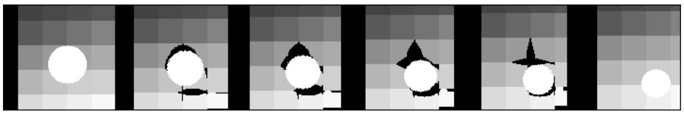
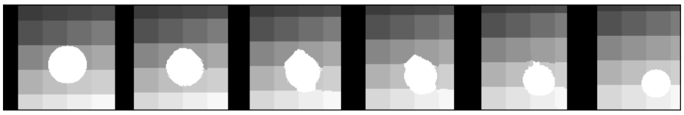


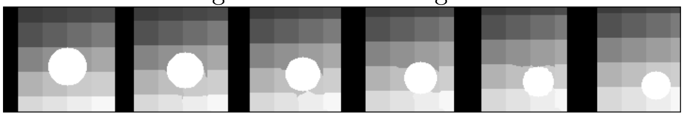
Fig. 5. Example of interpolation before the propagation step.

Fig. 6 shows that the *Skiz* algorithm gives bad results, since all labels propagate with the same priority. As the ball is well interpolated, its label should not propagate through the holes. On the contrary, the squares which are wrongly interpolated should all contribute to filling the holes.

As for the depth estimation, we rely on an error estimation to propagate each label in the interpolated images. On Fig. 7, in order to fill the area marked in black, we would like to compute $Err(A) = L_1$ and $Err(B) = L_2$, so that a label



Filling the holes: Skiz algorithm.



Filling the holes: Hierarchical propagation.

Fig. 6. Interpolation of a synthesis sequence, skiz algorithm

propagation beginning at time $-L_1$ for A and at time $-L_2$ for B would cover the hole exactly with the expected geometry.

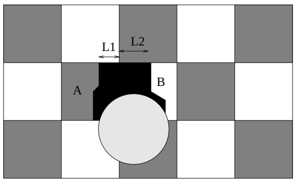


Fig. 7. Propagation of labels

We propose to use as propagation error $p(i)$ the Hausdorff distance between the compensated region $R'_{t+k}(i)$ and $R''_{t+k}(i)$. Even if we do not know the exact shape of A' and B', the Hausdorff distance between the compensated regions at time $t+k$ approximates the propagation error. This relies on the nature of the Hausdorff distance, which may be defined in terms of propagation distance.

7. Results

Fig. 8 gives four examples of interpolated sequences. The label sequences are computed with morphological segmentation tools [6]. To present the results, each region is filled with its mean value in the original image I_t for the display. In all sequences, we display one original image (on the left) and four interpolated images.

The quality of these results is good, in the sense that it is difficult to distinguish between interpolated and segmented partitions. In addition, the motion continuity of these partitions is well interpolated (see for instance the soft way the man in the sequence *Foreman* turns his head, in the first row of Fig. 8).

We have confirmed, in this paper, the interest of a region-based approach for image sequences interpolation. We have introduced a new algorithm, which performs the sequence interpolation with no other information than the region partition. It is meant to separate as much as possible the different types of data, and to be as robust as possible. This separation is a consequence of the region-based approach.

Fig. 8. Sequence interpolations

References

1. M. Bierling. Displacement estimation by hierarchical block matching. In *Visual Communications and Image Processing '88*, volume 1001, pages 942–951, 1988.
2. D. Huttenlocher, G. Klanderman, and W. Rucklidge. Comparing images using the Hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):850–863, September 1993.
3. F. Marqués, B. Llorens, and A. Gasull. Interpolation and extrapolation of image partitions using Fourier descriptors: Application to segmentation based coding schemes. In *IEEE International Conference on Image Processing*, volume III, pages 584–587, Washington, DC, October 1995.
4. F. Marqués, M. Pardàs, and P. Salembier. Coding-oriented segmentation of video sequences. In L. Torres and M. Kunt, editors, *Video Coding: The second generation approach*, pages 79–124. Kluwer Academic Publishers, 1996.
5. F. Meyer. Morphological interpolation method for mosaic images. In *Proceedings of International Symposium on Mathematical Morphology*, May 1996.
6. P. Salembier and M. Pardàs. Hierarchical morphological segmentation for image sequence coding. *IEEE Transactions on Image Processing*, 3(5):639–651, Sept. 1994.
7. P. Soille. Generalized geodesic distances applied to interpolation and shape description. In J. Serra and P. Soille, editors, *Mathematical morphology and its applications to image processing*, pages 193–200. Kluwer Academic Publishers, 1994.