

SPATIO-TEMPORAL SCENE LEVEL ERROR CONCEALMENT FOR SHAPE AND TEXTURE DATA IN SEGMENTED VIDEO CONTENT

Luis Ducla Soares¹, Fernando Pereira²

¹Instituto Superior de Ciências do Trabalho e da Empresa - Instituto de Telecomunicações

²Instituto Superior Técnico - Instituto de Telecomunicações

¹lds@lx.it.pt, ²fp@lx.it.pt

ABSTRACT

In this paper, a novel shape and texture error concealment technique for segmented object-based video scenes is proposed. This technique is different from existing concealment techniques because it considers, not only the corrupted video objects to be concealed, but also the context/scene in which they are inserted. In the proposed technique, concealment is done by using information from the current time instant as well as from the past. The obtained results suggest that the use of this technique significantly improves the subjective visual impact of scenes on the end-user, when compared to independent concealment of video objects.

1. INTRODUCTION

The emergence of the MPEG-4 object-based audiovisual coding standard [1] opened up the way for new video services, where scenes consist of a composition of objects. In order to make these object-based services available in error-prone environments, such as mobile networks or the Internet, appropriate error concealment techniques are necessary.

Several such techniques have already been proposed in the literature, e.g., [2][3][4][5][6][7] for shape and [8] for texture. These techniques, however, have a serious limitation in common, which is the fact that each video object is independently considered, without ever taking into account how the objects fit in the scene. After all, the fact that a concealed video object has a pleasing subjective impact on the user, when it is considered on its own, does not necessarily mean that the subjective impact of the whole scene, when the objects are put together, will be acceptable; this represents the difference between object level and scene level concealment. An example of this situation is given in Figure 1.

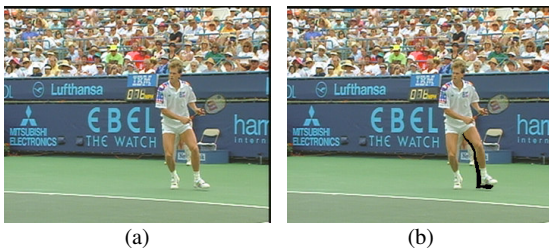


Figure 1 – Illustration of a typical scene concealment problem: (a) Original video scene; (b) Composition of two independently error concealed video objects

When concealing a complete video scene, the way the scene was created has to be considered since this will imply different problems and solutions in terms of error concealment. As shown in Figure 2, the video objects in a scene can be defined either by segmentation of an existing video sequence (segmented scene), in which case all shapes have to fit perfectly together, or by composition of pre-existing video objects whose shapes do not necessarily have to perfectly fit together (composed scene).

Additionally, it is also possible to use a combination of both approaches.

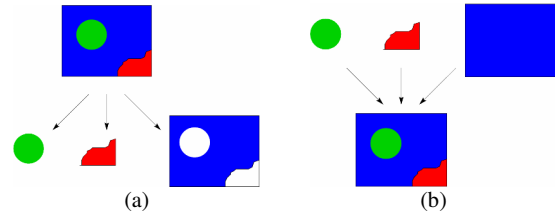


Figure 2 – Two different scene types: (a) Segmented scene; (b) Composed scene

In this paper, segmented video scenes (or the segmented parts of hybrid scenes) are considered since the concealment of composed scenes can typically be limited to object level concealment. In addition, the proposed technique, which targets the concealment of both shape and texture data, relies not only on available information from the current time instant, but also on information from the past — it is a spatio-temporal technique. In fact, it is the only known spatio-temporal technique that targets both shape and texture data, and works at the scene level.

2. SCENE LEVEL ERROR CONCEALMENT IN SEGMENTED VIDEO

In order to better understand the proposed scene level error concealment solution, the type of problems that may appear in segmented video scenes when channel errors occur should be briefly considered, as well as what can be done to solve them.

Segmented video scenes are obtained from rectangular video scenes by segmentation. This means that, at every time instant, the arbitrarily shaped video object planes (VOPs) of the various video objects in the scene will fit perfectly together like the pieces in a jigsaw puzzle. These arbitrarily shaped VOPs are transmitted in the form of rectangular bounding boxes, using shape and texture data. The shape data corresponds to a binary alpha plane which is used to indicate the parts of the bounding box that belong to the object and, therefore, need to have texture associated to it. For the remainder of the paper, it will be considered that some kind of block-based coding, such as (but not necessarily) defined in the MPEG-4 Visual standard [1], was used and that channel errors manifest themselves as bursts of consecutive corrupted blocks for which both shape and texture data will have to be concealed, both at object and scene levels.

2.1 Shape error concealment in segmented video

Since, in segmented scenes, the various VOPs in a time instant have to fit together like the pieces in a jigsaw puzzle, if there is any distortion in their shape data, holes or object overlap will appear, leading to a subjective negative impact. However, the fact that the existing VOPs have to perfectly fit in together can also be used when it comes to the concealment of shape data errors. In many cases, it will be possible to conceal at least some parts of the corrupted shape in a given corrupted VOP by considering

uncorrupted complementary shape data from surrounding VOPs. For those parts of the corrupted shape for which complementary data is not available because it is corrupted, concealment will be much harder. Thus, depending on the part of the corrupted shape that is being concealed in a VOP, two distinct cases are possible:

- **Correctly decoded complementary shape data** – The shape data from the surrounding VOPs can be used to conceal the considered part of the corrupted shape since it is uncorrupted.
- **Corrupted complementary shape data** – The shape data from the surrounding VOPs cannot be used to conceal the part of the corrupted shape at hand since it is also corrupted.

These two cases, which are illustrated in Figure 3, correspond to different concealment situations and, therefore, will have to be treated separately in the proposed technique.

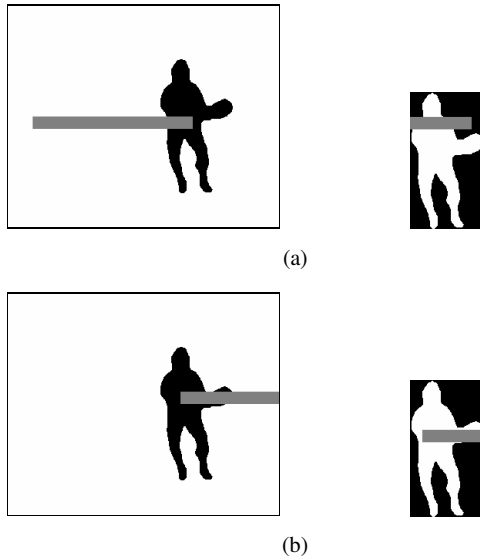


Figure 3 – Illustration of the two possible concealment situations for the Stefan video objects (Background and Player); (a) Correctly decoded complementary shape data exists; (b) Complementary shape data is corrupted in both objects

2.2 Texture error concealment in segmented video

When concealing the corrupted texture of a given VOP in a video scene, the available texture from surrounding VOPs appears to be of little or no use since different objects typically have uncorrelated textures. However, in segmented scenes, the correctly decoded shape data from surrounding VOPs can be indirectly used to conceal the corrupted texture data. This is possible because the shape data can be used to determine the motion associated with a given video object, which can then be used to conceal its corrupted texture, as was previously done in [6]. Therefore, by concealing parts of the corrupted shape data of a given VOP with the correctly decoded complementary shape data it will then be possible to estimate the object motion and then conceal the corrupted texture.

3. PROPOSED SCENE LEVEL ERROR CONCEALMENT ALGORITHM

By considering what was said above for the concealment of shape and texture data in segmented video scenes, a complete and novel scene level shape and texture error concealment solution is proposed in this paper. The proposed concealment algorithm includes two main consecutive phases, which are described with detail in the following two sections.

3.1 Shape and texture concealment based on available complementary shape data

In this phase, all the parts of the corrupted shape for which correctly decoded complementary shape data is available are concealed first. To do this for a given corrupted VOP, two steps are needed:

1. **Creation of complementary alpha plane** – To begin, a complementary alpha plane, which corresponds to the union of all the video objects in the scene except for the one currently being concealed, is created.
2. **Determination of shapel transparency values** – Afterwards, each corrupted shapel of the VOP being concealed is set to the opposite transparency value of the corresponding shapel in the complementary alpha plane. Since the complementary alpha plane can also have corrupted parts, this is only done if the needed data is uncorrupted.

This whole procedure is repeated for all video objects with corrupted shape. It should be noticed that, for those parts of the corrupted shape for which complementary data is available, this type of concealment recovers the corrupted shape without any distortion with respect to the original shape, which does not happen in the second phase described in Section 3.2.

In order to recover the texture data associated with the opaque parts of the shape data that has just been concealed, a combination of global and local motion (first proposed in [6]) is used. To do this for a given VOP, four steps are needed:

1. **Global motion parameters computation** – To begin, the correctly decoded shape and texture data, as well as the shape data that was just concealed, are considered in order to locally compute global motion parameters for the VOP being concealed.
2. **Global motion compensation** – Then, the computed global motion parameters can be used to motion compensate the VOP of the previous time instant.
3. **Concealment of corrupted data** – This way, the texture data associated with the opaque parts of the shape data that has just been concealed is obtained by copying the co-located texture in the motion compensated previous VOP.
4. **Local motion refinement** – Since the global motion model cannot always accurately describe the object motion due to the existence of local motion in some areas of the object, a local motion refinement scheme is applied. In this scheme, the available data surrounding the corrupted data being concealed is used to determine if any local motion exists and, if so, refine the concealment.

3.2 Shape and texture concealment for which no complementary shape data is available

In this phase, the remaining corrupted shape data, which could not be concealed in the previous phase because no complementary shape data was available in surrounding objects, will be concealed. The texture data associated with the opaque parts of the concealed shape will also be recovered. This phase is divided in two steps:

1. **Individual concealment of video objects** – Since the remaining corrupted shape of the various video objects in the scene has no complementary data available that can be used for concealment, the remaining corrupted shape and texture data will be concealed independently of the surrounding objects. This can be done by using any of the available techniques in the literature. Here, however, to take advantage of the high temporal redundancy of the video data, individual concealment of video objects will be done by using a combination of global and local motion compensation concealment, as proposed in [6]. This technique is applied to conceal both the shape and the texture data of the corrupted video object at hand.

2. **Elimination of scene artifacts by refinement of the object concealment results** – As a result of the previous step, holes or object overlap may appear in the scene since objects have been processed independently. The regions that correspond to holes are considered undefined, in the sense that they do not belong to any object yet (i.e., shape and texture are undefined). As for the regions where objects overlap, they will also be considered undefined and treated the same way as holes because a better method to deal with them (i.e., one that would work consistently for most situations) could not be found. In this last step, these undefined regions will be divided among the video objects around it. To do this, a morphological filter based on the dilation operation [9] is cyclically applied to the N objects in the scene, A_1, A_2, \dots, A_N , until all undefined regions disappear. The morphological operation to be applied to object A_j is the following:

$$A_j \oplus B - \left[A_j \oplus B \right] \cap \bigcup_{i=1, i \neq j}^N A_i, \quad (1)$$

where the 3×3 structuring element B that is used for the dilation operation \oplus is shown in Figure 4. By cyclically applying this filter, the undefined regions will be progressively absorbed by the objects around them until they finally disappear, as illustrated in Figure 5. To estimate the texture values of the pixels in these new regions, an averaging procedure is used. This way, in each iteration of the above mentioned morphological operation, the texture of the pixels that correspond to the shapes that have been absorbed is estimated by computing the mean of the adjacent 4-connected neighbors that were already included in the object. Since these regions over which texture concealment is necessary are typically very small, this procedure is adequate.

0	1	0
1	1	1
0	1	0

Figure 4 – Structuring element used for the dilation operation in the refinement of individual concealment results

4. PERFORMANCE EVALUATION

In order to illustrate the performance of the proposed shape concealment process, Figure 6 should be considered. In this example, the four video objects of the News video scene (in Figure 6 (a)) have been corrupted, as shown in Figure 6 (b). In the remainder of Figure 6, the various steps of the concealment process are shown, leading to the final concealed video objects in Figure 6 (f). To compare these video objects with the original ones in Figure 6 (a), the D_n and $PSNR$ metrics used by MPEG [6] may be used for shape and texture, respectively. The D_n metric is defined as:

$$D_n = \frac{\text{Different shapes in concealed and original shapes}}{\text{Opaque shapes in original shape}}, \quad (2)$$

which can also be expressed as a percentage, $D_n [\%] = 100 \times D_n$. As for the $PSNR$ metric, since arbitrarily shaped video objects are used, it is only computed over the pixels that belong to both the decoded VOP being evaluated and the original VOP. The obtained D_n values are 0.01%, 0.15%, 0.12% and 0.59%, respectively for the Background, Dancers, Speakers and Logo video objects shown in Figure 6 (f). The corresponding $PSNR$ values are 37.58 dB, 26.20 dB, 30.27 dB and 27.62 dB; the uncorrupted $PSNR$ values are 38.25 dB, 33.51 dB, 34.18 dB and 29.27 dB, respectively. As can be seen, although the shapes and textures of these video objects have been severely corrupted, the results are quite impressive, especially when compared to what is typically achieved by independent concealment alone. The main reason for such a

great improvement is the usage of the complementary shape data from surrounding objects during the concealment process, which does not happen when only independent concealment is performed.

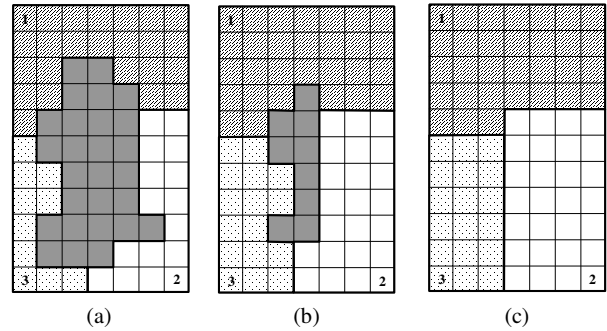


Figure 5 – Elimination of undefined region by morphological filtering: (a) Initial undefined region; (b) Undefined region is shrinking; (c) Undefined region has been eliminated

5. FINAL REMARKS

In this paper, a shape and texture concealment technique for segmented object-based video scenes, such as those based on the MPEG-4 standard, was proposed. Results have been presented showing the ability of this technique to recover lost data in segmented video scenes with rather small distortion. Therefore, with this technique, it should be possible for object-based video applications (with more than one object) to be actually deployed in error-prone environments with an acceptable visual quality.

6. ACKNOWLEDGMENT

The work presented was developed within VISNET, a European Network of Excellence (<http://www.visnet-noe.org>).

7. REFERENCES

- [1] ISO/IEC 14496-2, “Information Technology – Coding of Audio-Visual Objects, Part 2: Visual,” December 1999.
- [2] S. Shirani, B. Erol, F. Kossentini, “A Concealment Method for Shape Information in MPEG-4 Coded Video Sequences,” *IEEE Transactions on Multimedia*, Vol. 2, No. 3, pp. 185-190, September 2000.
- [3] L. D. Soares, F. Pereira, “Spatial Shape Error Concealment for Object-based Image and Video Coding,” *IEEE Transactions on Image Processing*, Vol. 13, No. 4, pp. 586-599, April 2004.
- [4] G. M. Schuster, X. Li, A. K. Katsaggelos, “Shape Error Concealment Using Hermite Splines,” *IEEE Transactions on Image Processing*, Vol. 13, No. 6, pp. 808-820, June 2004.
- [5] P. Salama, C. Huang, “Error Concealment for Shape Coding,” *Proc. of the IEEE International Conference on Image Processing*, Rochester, NY, USA, Vol. 2, pp. 701-704, September 2002.
- [6] L. D. Soares, F. Pereira, “Motion-based Shape Error Concealment for Object-based Video,” *Proc. of the IEEE International Conference on Image Processing*, Singapore, October 2004.
- [7] L. D. Soares, F. Pereira, “Combining Space and Time Processing for Shape Error Concealment,” *Proc. of the Picture Coding Symposium*, San Francisco, CA, USA, December 2004.
- [8] L. D. Soares, F. Pereira, “Spatial Texture Error Concealment for Object-based Image and Video Coding,” *Proc. of the EURASIP Conference on Signal and Image Processing, Multimedia Communications and Services*, Smolenice, Slovakia, June 2005.
- [9] R. C. Gonzalez, R. E. Woods, *Digital Image Processing*, 2nd Ed., Prentice-Hall, 2002.

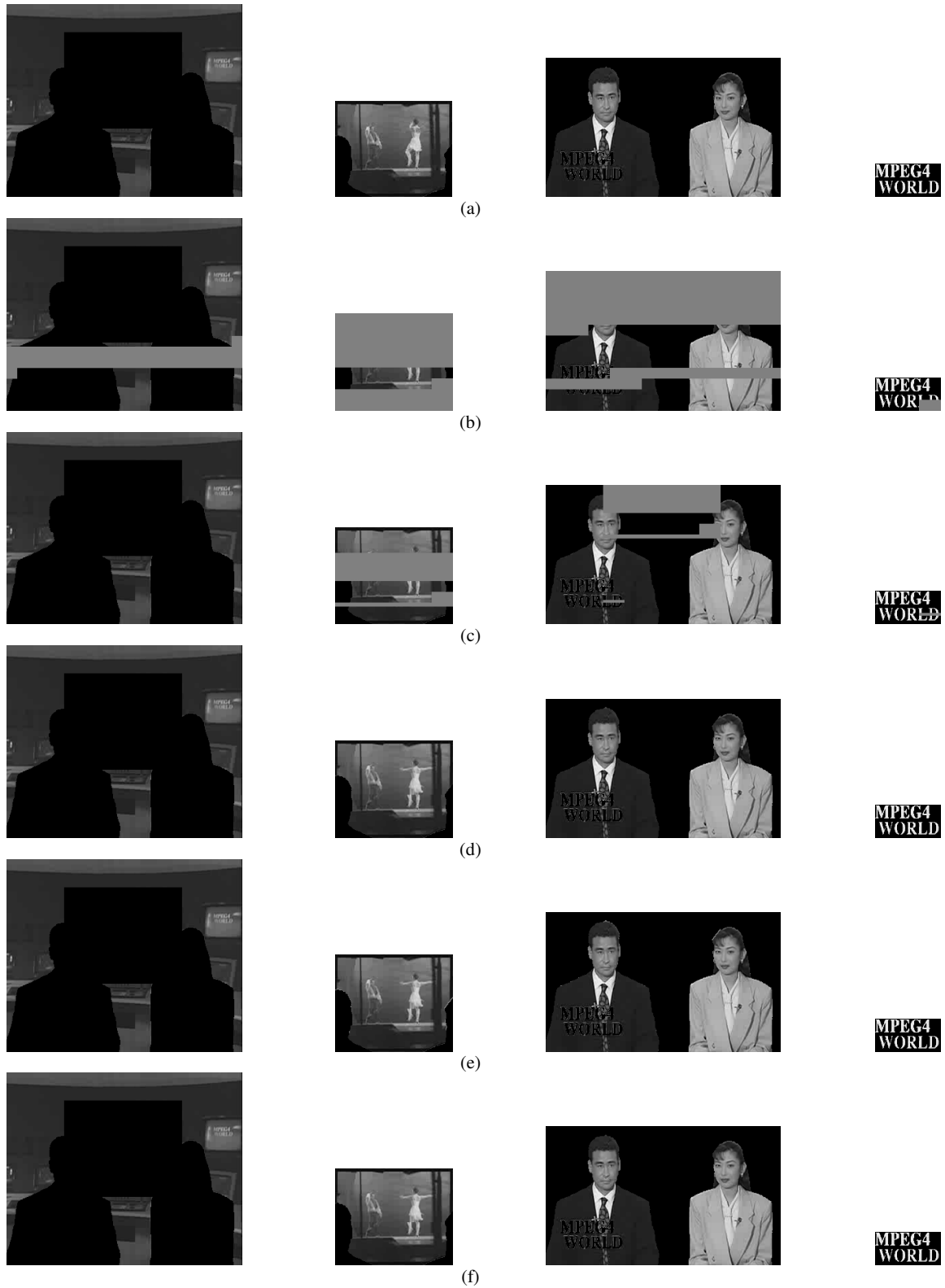


Figure 6 – The concealment process for the News video scene: (a) Original uncorrupted video objects (Background, Dancers, Speakers and Logo); (b) Corrupted video objects; (c) Video objects after the corrupted data for which complementary data exists has been concealed; (d) Video objects after individual concealment; (e) Undefined regions that appear after individual concealment (shown in grey); (f) Final concealed video objects