Urbicande-la-Neuve

On-line web entertainment mixing synthetic and natural images

X. Marichal, C. De Vleeschouwer, J.F. Gobbers,
J.F. Delaigle, J.M. Mas Ribès, B. Macq

Laboratoire de Télécommunications et Télédétection
Université catholique de Louvain
2 place du Levant
B-1348 Louvain-la-Neuve, Belgium
Tel: +32 10 47.41.05/22.71 - Fax: +32 10 47 20 89
E-mail: marichal@tele.ucl.ac.be, macq@tele.ucl.ac.be

A. Nandi

Magic Square - Casterman S.A.
81 rue du Marteau
B-1000 Bruxelles
Tel: +32 2 285 05 25
E-mail: an@magicsquare.be

ABSTRACT

The present paper introduces a "MPEG-4 - like" system providing content on the Internet. Street cameras capture real images from which moving objects are automatically extracted. Such objects are then combined with virtual background and objects coming from the comics world. The scene is then presented on-line to the user who can interact with it.

1. INTRODUCTION

Urbicande-la-Neuve aims at exploring new narrative concepts at the crossroads of art and technology, of virtuality and reality. Stories and algorithms are mixed together in order to give birth to another space for the comics of "The Obscure Cities" (cfr. *http://members.aol.com/IKONPress/index.html*).

Basically, three cameras have been placed in the city of Louvain-la-Neuve (cfr. *http://www.ucl.ac.be/LLN/visite-en.html*). The background images of these webcams have been redrawn with the architectural style of some Obscures Cities. Then, the aim is to extract the real objects (people, cars,...) moving in front of the camera and make them exist in the virtual background, along with virtual objects directly issued from the comics imaginary world.

In order to compose all the visual objects within one scene and to allow for interactivity, MPEG-4 [1] could be used as the transmission layer since it integrates both these functionalities. Nevertheless, due to the real-time constraints, only a subset of MPEG-4 is implemented. An on-line demonstration of the system is visible from *http://urbicande.tele.ucl.ac.be.*

The overall scheme is depicted on figure 1. This scheme implies the following aspects, which are developed in the next sections:

- To achieve a real time segmentation of moving objects captured by a relatively good quality camera;
- To associate these objects extracted from the real world with predefined synthetic objects in order to compose a coherent scene;
- To perform the real-time encoding of these arbitrary shaped objects with the help of various coding techniques that have appeared within the MPEG-4 framework;
- To design an interactive browser enabling to launch events on the basis of requests from the user.

2. SEGMENTATION

In order to extract the real moving objects out of the real scene captured by the camera, the segmentation scheme of figure 2 is used.

Basically, in order to segment the image at time t, two change masks are generated by comparison with the previous and the next pictures. Since every of these masks contains not only the moving object at its location at time t but also its location in the reference frame, both masks are combined with a logical AND operator. The resulting mask ($Mask_{tmp}$) generally depicts in a very good way the object contours.

This work was supported by a FIRST contract of the "Région Wallone".

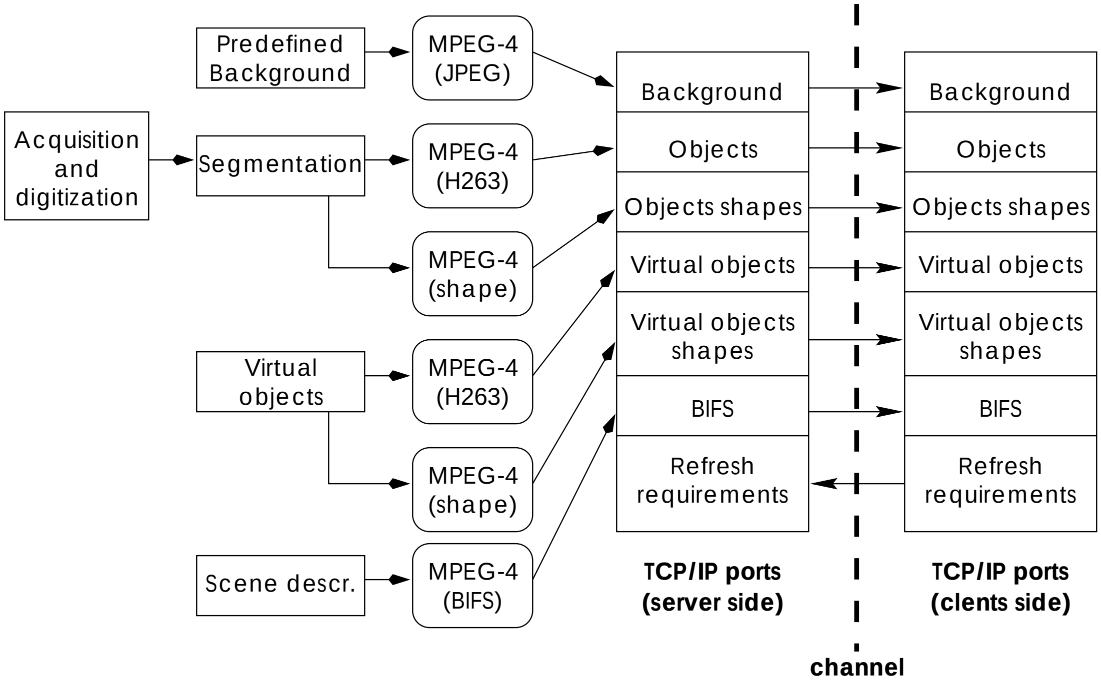

Figure 1: System overview

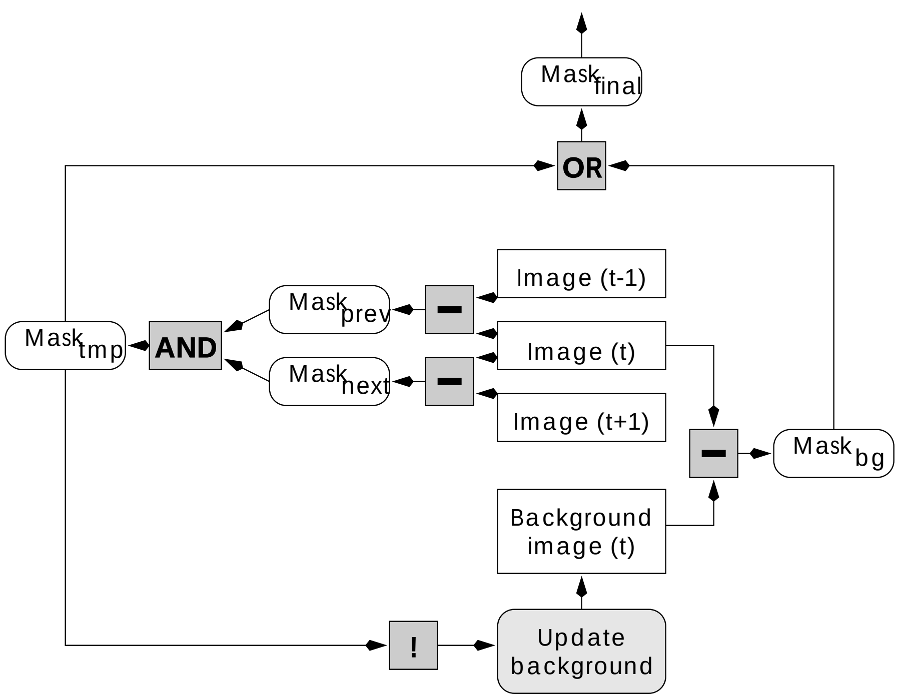

Figure 2: Real-time segmentation of moving objects

However, the inside of the objects is not always correctly detected as a part of this change mask. It is the reason why a reference image background is used. The change mask between this background image and the image to segment ($Mask_{bg}$) often allows detecting the inside of objects. A logical OR operation applied to $Mask_{tmp}$ and $Mask_{bg}$ provides the system with a reasonably good segmentation of the moving objects. Moreover, objects that stop moving are also detected since they appear in $Mask_{bg}$.

If the use of a reference background image is crucial to enable a complete and fast segmentation of objects, it does nevertheless necessitate to automatically generate this image. The problem is not trivial since the background illumination changes along time or due to weather conditions. It is therefore obvious that the background image cannot be extracted once for all but needs to be constantly updated. The solution is to use a mobile median filter. In order to further improve the quality of this background image, only pixels which do not belong to moving objects, i.e. which are not part of $Mask_{tmp}$, are injected in the filter.

3. CREATION OF VIRTUAL OBJECTS AND BACKGROUND

The real background is replaced with a synthetic background (see for instance figure 3) with the architectural style of "Obscures Cities". This has been drawn by professional artists from Casterman, specialized in comics.

In addition to the background, the artists have also drawn several (moving) virtual objects, like the ones depicted on figure 4. The appearance of these objects could be launched by events induced by the behavior of the real objects or explicitely requested through user interaction.

4. COMPRESSION AND TRANSMISSION

The various objects (MPEG-4 VOPs) are then encoded and stored into several buffers. Since MPEG-4 recognizes both H.263 [2] and JPEG [3] standards, these have been chosen as the compression algorithms because of their existing real-time implementations. However, it has to be noted that, for the purpose of random access from any new client connecting to the system server, only the intra-coding mode of H.263 is used.

Figure 4: Some virtual objects -(c) Schuiten and Casterman

Nevertheless, the use of video objects is more elaborated than a simple compression of rectangular images. Each frame is associated with a mask that allows defining the shape of the objects (alpha channel). This shape information is a binary one. It means that for each luminance pixel, a corresponding alpha pixel is set to 255 (transparency) or 0 (opacity). The masks are used during the composition by the rendering system.

In practice, macroblocks (MB) are encoded in a raster scan order. For each MB, the shape information is encoded thanks to the method described in [4, 5, 6]. Context-based arithmetic encoding (CAE) encodes the bitmap of shapes using the context of the neighboring pixels to predict the current one. Actually, a template of ten pixels is used: it defines the probability model for the CAE, that can be viewed as a Markov source model with ten states. For the macroblocks with all opaque or all transparent values, only the coding mode is sufficient without encoding the bitmap information. Once the shape information has been defined, the intra mode of H.263 is used to encode the texture (*lum, cb, cr*) information of MBs that contain some transparent pixels.

At the end, we have three kinds of objects:

1. Arbitrary shaped video objects coded with H263 and associated with an alpha channel.
2. The background (still image) coded with JPEG.
3. Additional objects encoded with JPEG or H263 depending on whether they are moving or not, with an associated alpha channel.

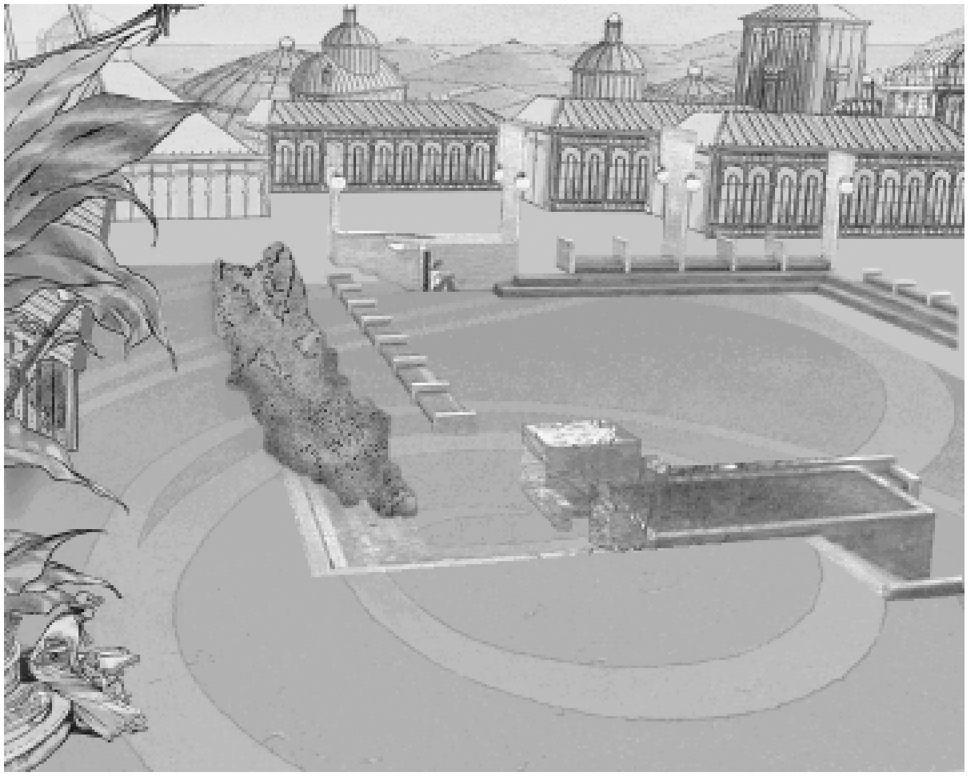

Figure 3: One virtual background -(c) Schuiten and Casterman

5. SCENE COMPOSITION AND INTERACTION

The different objects described above will be displayed together on the user's device. They need to be co-ordinated and synchronized together. So, temporal information is needed, as well as spatial information. The scene description is transmitted in a separate bit stream, compliant with the MPEG-4 BIFS description.

The visualization application is a JAVA applet to be plugged into a browser. It then emulates, in a very simplified way, the behavior of the flex-mux and DMIF parts of MPEG-4 via TCP-IP queries.

6. REFERENCES

[1] F. Pereira, K. O'Connell, R. Koenen, and M. Etoh. Special Issue on MPEG-4, Part 1: Invited Papers. *Signal Processing: Image Communication*, 9(4), May 1997.

[2] Telecommunication Standardization Sector of ITU. Draft ITU-T Recommendation H.263. *ITU Recommendations*, July 1995.

[3] G.K. Wallace. The JPEG Still Picture Compression Standard. *Communications of the ACM*, 34(4):30–44, April 1991.

[4] MPEG Video group. MPEG-4 Video Verification Model, version 11.0, ISO/IEC JTC1/SC29/WG11 (N2178). Tokyo, March 1998.

[5] N. Brady, F. Bossen, and N. Murphy. Context-Based Arithmetic Encoding of 2D Shape Sequences. volume I, pages 29–32, Santa Barbara, October 1997. Int. Conf. on Image Processing (ICIP).

[6] J. Osterman, E.S. Jang, and J.-S. Shin. Coding of arbitrarily shaped video objects in MPEG-4. volume I, pages 496–499, Santa Barbara, October 1997. Int. Conf. on Image Processing (ICIP).