

# Scalable and Efficient Video Coding Using 3-D Modeling

Raphaële Balter, Patrick Gioia, and Luce Morin

**Abstract**—In this paper, we present a three-dimensional (3-D) model-based video coding scheme for streaming static scene video in a compact way but also enabling time and spatial scalability according to network or terminal capability and providing 3-D functionalities. The proposed format is based on encoding the sequence of reconstructed models using second-generation wavelets, and efficiently multiplexing the resulting geometric, topological, texture, and camera motion binary representations. The wavelets decomposition can be adaptive in order to fit to images and scene contents. To ensure time scalability, this representation is based on a common connectivity for all 3-D models, which also allows straightforward morphing between successive models ensuring visual continuity at no additional cost. The method proves to be better than previous methods for video encoding of static scenes, even better than state-of-the-art video coders such as H264 (also known as MPEG AVC). Another applications of our approach are smoothing camera path for suppression of jitter from hand-held acquisition and the fast transmission and real-time visualization of virtual environments obtained by video capture, for virtual or augmented reality and interactive walk-through in photo-realistic 3-D environments around the original camera path.

**Index Terms**—3-D model-based Coding, 3-D reconstruction, second-generation wavelets.

## I. INTRODUCTION

WITH the development of video applications over networks and wireless devices such as cell phones and personal digital assistants (PDAs), low bit-rate video compression is still a key issue. More precisely, distant visualization on heterogeneous terminals requires video coding schemes providing a scalable bitstream adaptable to multiple and variable terminal resources.

State-of-the art video coders rely on pixel-based prediction/correction paradigms and they provide very efficient compression algorithms for generic contents video sequences in a wide range of bitrates. Indeed, this type of compression scheme has been optimized to reach its best performances in the latest standard video coder H264 ITU/MPEG-AVC [1].

Manuscript received December 7, 2004; revised November 17, 2005. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yoshihisa Shinagawa.

R. Balter is with France Telecom R&D, 35512 Cesson-Sevigne, France, and also with the IRISA-INRIA, Campus de Beaulieu, 35042 Rennes, France (e-mail: raphael.balter@orange-ft.com).

P. Gioia is with France Telecom R&D, 35512 Cesson-Sevigne, France (e-mail: patrick.gioia@orange-ft.com).

L. Morin is with IRISA-INRIA, Campus de Beaulieu, 35042 Rennes, France (e-mail: luce.morin@irisa.fr).

Color versions of Figs. 1–6, 8–12, and 14–16 are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TMM.2006.879873

Exploiting particularities of the encoded content can dramatically improve compression efficiency by using specific coding schemes. Typically, 3-D model-based video coding exploits the fact that the scene contains known objects for which a specific or generic 3-D model is available and can be transmitted together with texture and animation parameters. This method produces very efficient compression and it is particularly well adapted to the video-conference field where a 3-D model of the human face is used to represent the video sequence of the speaker [2]. However, it is necessary for the scene content to be known and that an *a priori* known 3-D model is available both at the coder and the decoder [3]. For a video with unknown contents, 3-D model-based coding may still be used if the scene is static, i.e., with no moving object inside, and acquired by a moving camera, by automatically reconstructing the captured 3-D environment from the video and transmit it as a 3-D model, a texture and camera parameters.

Automatic 3-D modeling of static scenes from uncalibrated images and video sequences has been studied for a long time, using computer vision structure-from-motion and self-calibration techniques [4]–[7]. Most previous work focused on off-line video analysis for obtaining an accurate 3-D model of the scene in order to replace manual modelling or to provide a precise reference frame for augmented reality [8], but few works have considered the issues of compressing and streaming the resulting 3-D representation.

Such considerations have been mainly addressed for transmitting synthetic 3-D models. Several methods have been proposed for the efficient and scalable coding of the 3-D models geometry and connectivity providing a progressive and scalable bitstream [9]–[11]. It is assumed that texture will be transmitted as an image using standard fixed image coders or as a few parameters in the case of parametric texture. However, in the case of 3-D models extracted from videos, texture is the most expensive information to be transmitted, and it is also a major factor in the final visual quality. With such input data, an effective 3-D model coding and streaming scheme should take into account the geometry, connectivity and texture. In the context of multi view-point acquisition, as studied in the MPEG-3-DAV consortium [12], real-time streaming of 3-D point-based representation has been proposed, assuming fixed and calibrated cameras [13]. Other approaches seek to compress image-based rendering view-sets of virtual environment [14]. 3-D information can be used to compensate disparity between between images [15]. However all those representations are often limited to small objects or require a particular capture system.

In order to benefit from the compactness and from the functionalities of 3-D model-based coding in the case of unknown scene contents and uncontrolled acquisition procedure, we pro-

pose a video coding scheme based on a set of successive 3-D models extracted from subsections of the video, instead of a unique one containing all the information viewed in the entire video sequence, as in previous automatic shape-from-motion schemes [7]. This choice has several advantages:

- Global consistency of extracted 3-D information is not required. This allows us to simplify estimation and use inaccurate camera parameters.
- The set of 3-D models directly provides a streaming format.
- Global illumination changes along time are modeled and reconstructed.
- Sequences of arbitrary size can be processed with on-the-fly estimation and streaming of the 3-D models.
- Camera motion is unconstrained.

Our first experiments validated this approach for low bitrate coding [16]. This scheme still allows 3-D functionalities usually provided by classical 3-D model-based video coding, such as illumination changes, object insertion, stereoscopic visualization or generation of virtual viewpoints close to actual ones [17].

However, the previously proposed scheme does not provide full scalability, which is a key point for targeted applications such as distant and interactive visualization. In particular, mesh geometry has fixed resolution. It involves, furthermore, a complex and computationally expensive morphing and re-meshing process at the decoder side to ensure smooth visual transition between successive 3-D models [16], [18].

In both image coding and synthetic 3-D models coding, wavelets [19] have been effectively used to achieve scalability in an elegant and efficient way. Second-generation wavelets [20] provide hierarchical representations for arbitrary sampled data and they are the current most effective tool for scalable representation of 3-D models [21].

Therefore we propose a scalable video coding scheme based on wavelet decomposition of an *evolving model* represented by a consistent 3-D models stream. It provides low bit-rate coding, as well as time and spatial scalability and 3-D functionalities. Targeted applications include impact simulation, and help for geo-positioning or virtual tourism.

In the following study, we first give an overview of the proposed method, then briefly describe the extraction of 3-D models in Section III and explain the proposed hierarchical representation more thoroughly in Section IV. We then present inter-relations between media flows and we explain the coding and decoding schemes in Section V. Finally the results on real video sequences are finally shown and discussed.

## II. OVERVIEW OF THE METHOD

The proposed representation is based on 3-D information extracted with the Galpin reconstruction algorithm. For each subsection of the sequence called a *group of frames (GOF)*, it provides a dense depth map and camera positions for each frame in the GOF. The behavior of the coder is then as follows.

The first step is to transform each depth map into a hierarchical 3-D triangular mesh. We define the *base mesh*, denoted  $BM_k$  as the mesh related to GOF  $k$  at coarser level and the *fine mesh*, denoted  $FM_k$  as the dense mesh related to GOF  $k$  at finer level. The refinement from coarser to finer level is

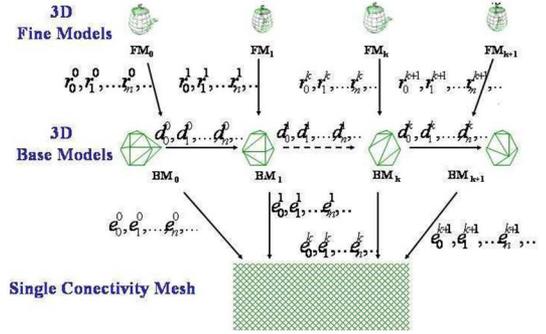


Fig. 1. Proposed representation based on a 3-D model stream and second-generation wavelet.

then expressed as wavelets coefficients ( $r_i^j$  on Fig. 1) using a second-generation wavelet transform. Scale coefficients ( $e_i^j$  on Fig. 1) represent the geometry of the base mesh  $BM_j$ . Successive 3-D models in the stream are encoded differentially with coefficients  $d_i^j$ .

To ensure time consistency of the wavelets coefficients for successive models, wavelet decomposition is applied based on a single connectivity mesh (SCM), possibly evolving in time, and gathering the connectivities of each base mesh model.

This representation induces several media streams, such as topology (the connectivity of the single base mesh), geometry (wavelet coefficients and incremental model representation), and texture. These streams are closely interrelated and they are multiplexed in order to produce a single streamable format.

In Sections III–VII, we describe the main components and the stream types they generate further.

## III. 3-D MODELS GENERATION

The 3-D models stream is automatically extracted from the input video sequence using shape from motion methods developed in computer vision [5]. Unlike state of the art multiview stereo reconstruction algorithms such as [22] which aim at reconstructing an accurate 3-D model from calibrated cameras and controlled view-points, our reconstruction method is focused on retrieving a 3-D model suitable for video restitution, and compatible with uncontrolled video acquisition. Each 3-D model is extracted and used for a restricted portion of the video sequence called a GOF. Two successive GOFs share one image (cf. Fig. 2). These border images are usually called *keyframes*. Keyframes are automatically selected according to video contents, based on several criteria. These criteria mainly depend on motion, percentage of outgoing points in images, and 3-D reconstruction feasibility and stability [17]. On average, a GOF contains 30 frames.

Disparity estimation is performed by a dense mesh-based affine motion estimator using multigrid and multiresolution approaches [23]. This robust algorithm minimizes the EQM and allows to estimate large disparities with lighting variations, thanks to mesh deformation. The motion field is then readjusted under epipolar geometry constraint. The camera intrinsic parameters are set to approximate values in order to get a quasi euclidean reconstruction. Extrinsic parameters define camera 3-D motion during the acquisition. They are computed using

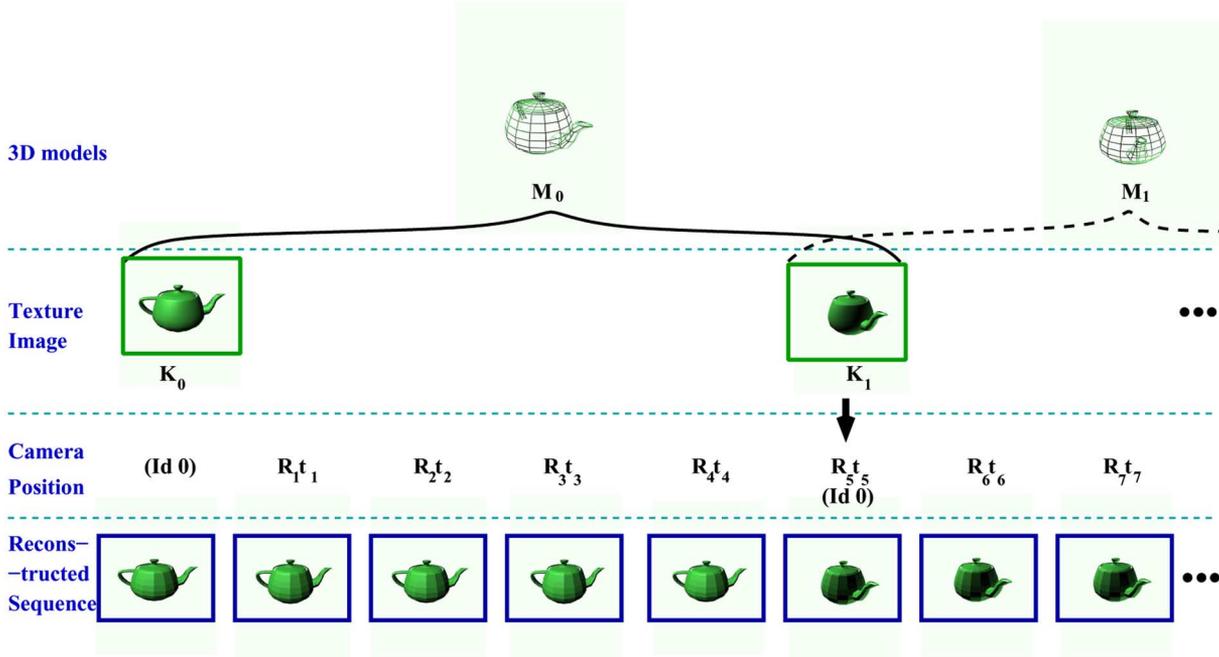


Fig. 2. Reconstruction of the original sequence.

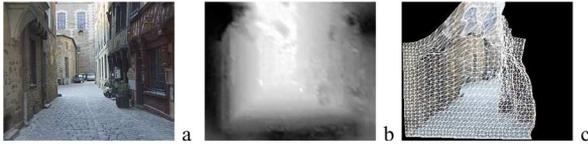


Fig. 3. Example of a depth map (b) extracted from the *Street* video sequence, the associated vrm model (c), and the corresponding image in the sequence (a).

classical self-calibration methods and an adapted bundle adjustment algorithm [17] allowing readjustment between models that is necessary for virtual reality applications. The dense motion field from the first to the last image of the GOF and camera parameters for these two images allow the reconstruction of a dense depth map for the first image of the GOF. Fig. 3 shows an example of such a depth map extracted from the *Street* video sequence. Camera-extrinsic parameters are then retrieved for each image in the video sequence using a pose estimation algorithm.

The 3-D reconstruction step thus provides for each GOF:

- the 3-D model geometry: a dense depth map of the scene viewed from the first image in the GOF;
- the 3-D model texture: the first image in the GOF;
- camera parameters for each frame in the GOF.

#### IV. HIERARCHICAL 3-D MODELS

We now explain how the hierarchical 3-D triangular mesh is constructed from the dense depth map.

##### Notations

The following notations will be used in the rest of the paper.

- $M_k^i$  is the 3-D model related to GOF  $k$  at resolution  $i$ .

- $K_k$  is the keyframe for GOF  $k$  (i.e., the first image in GOF  $k$ , also used as texture image  $T_k$  for  $M_k^i$ ).
- $C_k$  is the camera position related to keyframe  $K_k$ .  $C_k$  is defined by a translation  $t_{t_k}$  and a rotation  $R_{t_k}$ .
- $CM_k$  is the corresponding mesh; it denotes the 3-D model associated with GOF  $k$  whose vertices match vertices in the precedent model  $M_{k-1}^i$  that are still visible from the next point of view  $C_k$  and the related faces.
- We denote as  $Pr(M, T, C)$  the image issued from perspective projection of 3-D model  $M$  textured with image  $T$  onto the viewpoint related to camera  $C$ .

##### A. Single-Connectivity Mesh and Global Indexing

Using a stream of 3-D models instead of a unique one for the whole sequence provides several benefits that were mentioned previously. However, it also has the drawback of independently and arbitrarily sampling each 3-D model. As a consequence, the vertices of two successive models are not matching points, whereas the models usually represent largely overlapping parts of the scene (see Fig. 5). Applying hierarchical wavelet decomposition on such independently sampled models, as illustrated in Fig. 4, leads to high residual information and suboptimal coding efficiency. Moreover, such independent sampling prevents smooth swapping between 3-D models at visualization stage. Therefore, we propose to build a consistent sampling for all 3-D models with vertices corresponding to identical physical points. This is done by separating connectivity and geometry; a planar graph, denoted as *single connectivity mesh (SCM)*, gathers the connectivity information of every base mesh in the sequence, regardless of their geometry. This mesh evolves during time in order to take into account outgoing and incoming points. The SCM is computed starting from the connectivity information of the

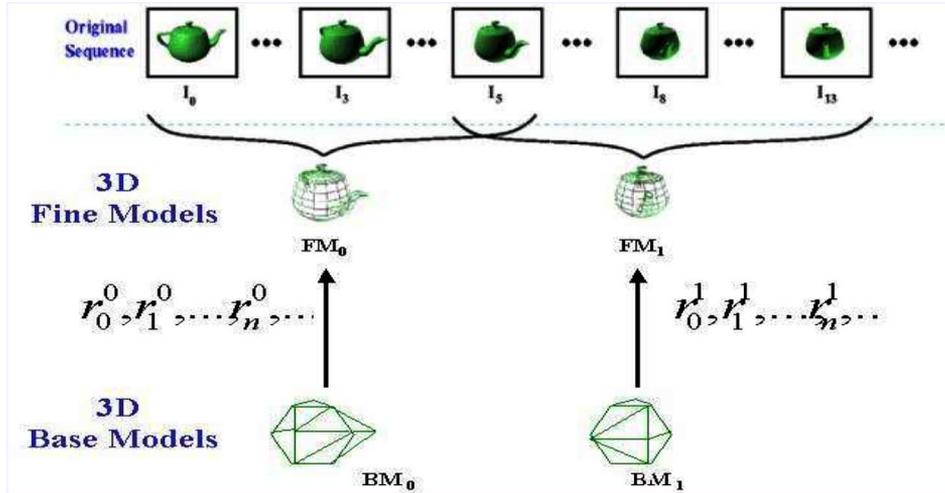


Fig. 4. Principle of the wavelet decomposition for the 3-D model stream.

first base mesh, and updated with the connectivity information associated with new points appearing from one base mesh to another. The SCM computation and update is based on the base meshes construction described in Section IV-C. A global indexing system provides a unique index for each vertex in the SCM, thus implicitly defining matching between base meshes vertices. The SCM is described as a list of triangles expressed in the new global indexing system. The SCM also provide a unique index for each face in the SCM.

Wavelet decomposition based on the SCM is consistent for all models and leads to compact coding. Moreover, smooth swapping at visualization can then be achieved by direct morphing between vertices without ghost effect due to a fading [17] nor morphing additional computing cost [16], [18]. This can be done at each level of subdivision thanks to the consistent connectivity of all base meshes and the global index system. Indeed, thanks to base mesh faces global index, a unique index can be computed for all vertices at each level using his barycentric coordinates in base face. At each level  $i$ , smooth transition between models  $M_k$  and  $M_{k+1}$  can be achieved by linear interpolation between corresponding vertices

$$M_c = \alpha * M_k^i + (1 - \alpha) * M_{k+1}^i \text{ with } \alpha = \frac{\|t_{t_{k+1}} - t_{t_c}\|}{\|t_{t_{k+1}} - t_{t_k}\|}$$

where  $M_c$  denotes the interpolated model for current time  $t_c$  and  $t_{t_c}, t_{t_k}$  and  $t_{t_{k+1}}$  denote translation vectors defining camera position for the current frame, keyframe  $K_k$  and  $K_{k+1}$ , respectively.

### B. Base Meshes Construction

Base meshes use nonuniform triangulation in order to ensure global connectivity consistency and smooth transition between models. Furthermore, to better represent the video content, the base mesh must also to fit features of the scene.

For the first GOF, the adaptive triangular mesh is based on feature points computed on the first frame in the GOF (keyframe), using a block-based Harris corner detector [24].

The size of block used for Harris detection fixes the number of the base mesh vertices (e.g., 200 vertices for our experiments). A two-dimensional (2-D) Delaunay triangulation of these points under the constraint of image borders provides the base mesh connectivity. In order to avoid texture stretching near image borders, we add vertices on image borders. The 3-D base mesh  $BM_0$  is then derived by elevation of this 2-D mesh using 3-D information provided by the depth map.

In order to build the SCM, each base mesh is forced to contain the correspondents of the previous base mesh vertices, if they are still visible in the GOF. To meet the SCM constraint, triangles of these correspondents, whose set is denoted by  $CM_k$ , are included in the base mesh:  $CM_k \subseteq BM_k$ .

When adding vertices on the border of the model, the new triangulation has to preserve the connectivity derived from the preceding GOF without edge crossing. This is achieved by 2-D Delaunay triangulation constrained by image borders and correspondent mesh  $CM_k$  borders. These new triangles are added to the SCM.

In Section IV-D, we describe the wavelet analysis scheme applied on the base meshes in order to provide a multiresolution scalable representation for each 3-D model.

### C. Wavelet Decomposition

The goal of wavelet decomposition is to decorrelate geometric information so as to proceed to the first step towards compression. In addition, the multiresolution aspect of this transform allows very efficient reconstruction and transmission, possibly in real time [25].

Since we describe geometric deformations, first-generation wavelets do not apply. Indeed, these parameterizations are defined over topological spaces (typically base meshes  $BM_k$  of Fig. 1) which are not linear spaces. Thus, wavelets themselves have to be defined according to the base domain, its subdivisions and geometric irregularities.

In the context of subdivision surfaces [20], wavelets can be defined starting from a low-pass reconstruction filter  $P^j$ . This filter operates over a global topological subdivision consisting in facets quadrisections, similarly as interval dichotomies in the

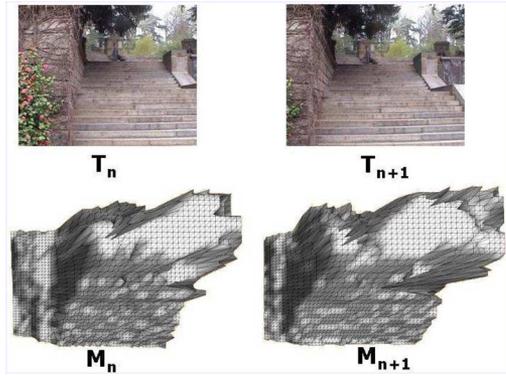


Fig. 5. Successive models for the *Thabor* sequence (c)  $M_n$ , (d)  $M_{n+1}$  and associated texture image (a)  $T_n$ , (b)  $T_{n+1}$  (lateral translation of the camera).

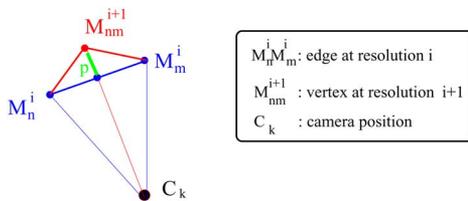


Fig. 6. Estimation of the depth to encode with wavelets.

classical wavelet setting. Filter  $P^j$  transforms coefficients at level  $j - 1$  into a prediction at level  $j$

$$c^{j+1} = P^j c^j. \quad (1)$$

The resulting coefficients are an approximation, without adding any information, which coincides with the refinement operator in the case of subdivision surfaces. The wavelet setting can be seen as “completing” the representation by adding details through a high-pass reconstruction filter  $Q^j$ . This filter has to satisfy an exact reconstruction criterion, which implies that matrix  $(P^j Q^j)$  is invertible.

Scaling functions  $(\phi_i^j)$  and wavelets  $(\psi_i^j)$  are directly defined by these filters, so that the parameterization to transform can be expressed as

$$S = \sum_{j \geq 0} \sum_i d_i \psi_i^j + \sum_i c_i^0 \phi_i^0. \quad (2)$$

In our case, we use continuous piecewise linear wavelets, which implies that matrix  $P^j$  has the form  $P = (I P')^t$  where  $I$  denotes identity and  $P'$  a canonical averaging matrix. As for matrix  $Q$ , it is chosen so that the resulting wavelets are stable and provide good compression. This is achieved by the requirement of vanishing moments through the lifting scheme [26].

For encoding depth maps, we start by defining a geometrical deformation as illustrated on Fig. 6. The transform is expressed as a scalar in terms of distance to the observer: the scalar function  $\rho : MB_k \rightarrow \mathbf{R}$  maps a point  $x$  on the base mesh to the offset  $p$  between  $x$  and  $M_{nm}^{i+1} \cap (C_k x)$ .

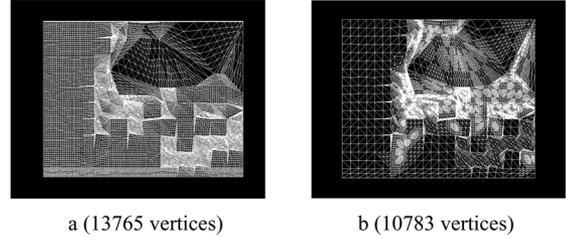


Fig. 7. Example for sequence *Thabor*: progressive meshes reconstructed with (a) classical and (b) adaptive wavelet decomposition.

We then process the wavelet transform by applying  $(P^j Q^j)^{-1}$  for every  $j$

$$\begin{pmatrix} c^j \\ d^j \end{pmatrix} = (P^j Q^j)^{-1} c^{j+1}. \quad (3)$$

The representation to encode is shown on Fig. 1. Fine models  $FM_i$  are represented by base meshes  $BM_k$  and wavelet coefficients  $r_i^j$ . Scale coefficients  $e_i^j$  expressing the geometry of base meshes are gathered and indexed by the SCM.

In classical decomposition, all faces are subdivided at the same fixed level at the encoding stage. However all the faces of the mesh do not need to be subdivided at the same level depending on the size on the face and on the part of the scene they represent. Therefore we introduce an adaptive wavelet decomposition. The level a face is decomposed at is given by the size of the 3-D face in order to gather two criteria; the depth gradient and the area of the 2-D face of the image. Fig. 7 gives examples of meshes given by classical and adaptive decomposition for the *Thabor* sequence.

At this stage, we have obtained a set of multiresolution meshes based on non-uniform triangulation, with corresponding vertices. This representation has several advantages, among which are:

- vertex positions can be adapted to scene contents;
- vertex to vertex correspondence between successive models is implicitly provided by the mesh structure and therefore does not need to be transmitted or estimated at the decoder side. It allows to smooth transitions between 3-D models through implicit morphing using a simple linear interpolation between vertices.

This 3-D representation for videos induces several media streams, such as topology (the connectivity of the single base mesh), geometry (wavelet coefficients and incremental model representation), and texture as well as camera parameters for each frame. Efficient coding of these streams is described in Section V.

## V. COMPRESSION OF THE REPRESENTATION

### A. Inter-Relations Between Different Media

A key observation is that the information streams to be transmitted are not independent and an efficient coding algorithm should take into account this redundancy for both compression rate and quality of the reconstructed sequence.

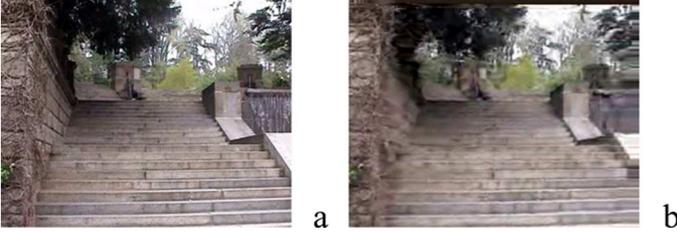


Fig. 8. *Thabor* sequence: predicted images. (a) Image 107 from original sequence. (b) Associated predicted image.

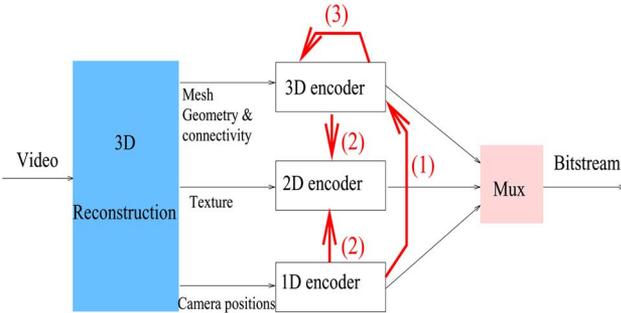


Fig. 9. Inter-relationships between the media in 3-D model-based coding.

Here is the description of inter-relationships occurring within 3-D model-based coding that we use in our coder.

First of all camera positions can help the transmission of 3-D models. Indeed, for each vertex of the nonuniform meshed model, five coordinates have to be transmitted: three coordinates for vertex location (3-D coordinates) and 2 for texture (2-D coordinates). If the camera positions are known on the decoder side only three parameters instead of five are required. Indeed 2-D texture coordinates can be retrieved by reprojecting 3-D vertices  $M_j$  on camera position of the GOF key frame viewpoint. As the key frame is also the texture image, the coordinates of the resulting projection  $m_j$  are the texture coordinates for vertex  $M_j$ . These parameters can be the exact positions of 3-D vertices  $M_j$  or texture coordinates  $m_j$  and the associated depth  $d_j$  if the 3-D model is an elevation map. This is represented on Fig. 9 by the arrow (1).

Furthermore, since 3-D models represent overlapping parts of the scene, the related textures include redundant information. To exploit this redundant information, compressing texture images using a classical scheme IPP, where the first image is in Intra mode and the others are in Predicted mode, is useful. Using one-dimensional (1-D) and 3-D information predicted images can be estimated thanks to the reprojecting of the precedent textured model onto associated camera (cf. Fig. 8). This is represented by the arrows (2) on Fig. 9.

In the same way, 3-D models geometry share common information. This redundancy can be reduced by using an IPP scheme for 3-D information. Predicted models are given by the common part of the precedent model. This is represented on Fig. 9 by the arrow (3). These inter-relationships are summarized on Fig. 9. To taking into account those interrelations allows to dramatically reduce the bit rate [27].

Depending on the envisioned applications, texture or geometry has to be favored. For instance, texture is very important in

video broadcasting applications, because human vision is very sensitive to texture. In virtual reality applications (virtual viewpoint generation or addition of objects for example), 3-D geometry has to be more accurate. With a unique stream instead of several ones we can update on the fly the rate associated to each medium whose particular encoders are presented in Section V-B.

## B. Camera Encoding

Key frame camera positions are differentially encoded and intermediate camera positions are retrieved by linear interpolation between key positions

$$C_c = \alpha * C_k + (1 - \alpha) * C_{k+1} \text{ with } \alpha = \frac{\|t_{k+1} - t_c\|}{\|t_{k+1} - t_k\|}.$$

## C. Texture Encoding

An IPP scheme is used where predicted image  $P(I_{k+1})$  is obtained by the reprojecting of the precedent textured model on the current key position, as described in Section V-A. Padding is used in areas where prediction does not apply (cf. Fig. 8)

$$P(K_{k+1}) = Pr(M_k^i, T_k, C_{k+1}).$$

Fine granularity scalability for the texture images is allowed by EBCOT coder. The use of the IPP scheme hinder decoding scalability. Therefore we add a low bandwidth for texture transmission. At the coding stage, the image used in order to get the prediction is the precedent image, but decoded at a very low bit rate. Refinements are transmitted in the error image.

## D. Connectivity and 3-D Geometry Encoding

Three-dimensional information encoding is based on the base mesh  $BM_k$  and a set of wavelet coefficients for refinements.

Two-dimensional texture coordinates are not encoded, as explained in Section V-A since they can be retrieved by reprojecting the 3-D model on the related key position.

The base meshes are encoded using a topological surgery (TS) encoder [10] for geometry and connectivity. We can rapidly identify vertices having a correspondent in the next model by re-projecting vertices of the current model on the key image of the next GOF. In this way, we retrieve the common information between two models at the decoding stage without transmitting additional information. The global indexing system introduced in Section IV helps to implicitly encode correspondences between successive base meshes. In order to avoid numerical errors, a stage of robust selection of base mesh vertices is added to the Harris corner selection.

After the wavelet transform, we get some sets of wavelet coefficients  $(r_i^j)_i^j$  with low first-order entropy. This representation is then binarized using a zero-tree algorithm suited to the geometric setting [21], [28]. To this end, a special hierarchy is setup on the mesh, ordering vertices instead of facets. The SPIHT algorithm can be applied directly onto this hierarchy, similarly to the 2-D case. The use of this adaptation of the SPIHT zero-tree encoder adds bitplane scalability.



Fig. 10. *Thabor* sequence: (a) texture image and (b) reconstructed image. While its PSNR is much lower, the visual quality of the reconstructed image is similar to the texture image visual quality.

Note that the SPIHT algorithm does not contain any entropy coding stage. It is possible to take advantage of such a coding in a post-process, but this may not be desirable in the case of adaptive decoding or bitstream degradation, since it makes real-time decoding slower.

## VI. RESULTS AND DISCUSSION

We show results on two sequences, illustrating the compression rates reached by comparison to Galpins and H264 encoders at low and very low bitrates on both constrained and free navigation.

For the wavelet decomposition we use the classical midpoint bi-orthogonal analysis performing a subsampling [26].

### A. Visual Quality and PSNR

While PSNR is appropriate for measuring block based errors, it has however, little meaning when it comes to geometric distortion. Global distortion on reconstructed images is produced both by texture (texture image compression artefacts) and geometric distortions (from 3-D model estimation errors and depth compression artefacts). Geometric distortion greatly decreases the peak signal-to-noise ratio (PSNR) when it may have little impact on visual quality. A demonstrative example is the geometric distortion defined by a one-pixel translation.

This is shown by comparing visual quality and PSNR of texture images and reconstructed images, as in Fig. 10. The texture image PSNR is the PSNR obtained with texture distortion alone, and without geometric distortion. It is dramatically much larger than the eventual PSNR value on reconstructed image, but visual quality is equivalent for both images.

Thus, low PSNR values of reconstructed images are essentially due to geometric distortion, but they do not reflect visual quality, which is more related to texture accuracy.

We thus show PSNR values in an informative way and rather rely on visual assessment of the reconstructed images, in particular in the case of free viewpoint generation for which PSNR has little meaning.

### B. Compression Results

We show compression results for a sequence of 110 frames of the *Thabor* sequence for low and very low bit rates on Figs. 11 and 12. No comparison can be made with H264 if such a low rate cannot be reached at 25 Hz.

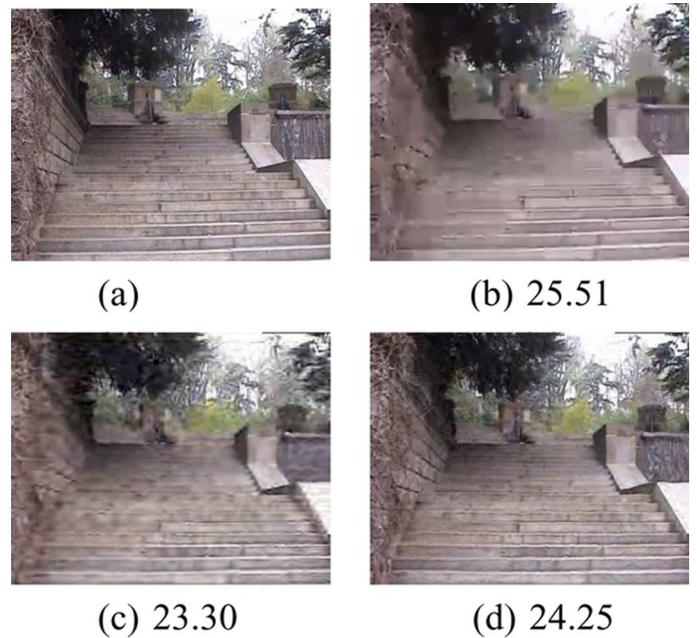


Fig. 11. *Thabor* sequence: Image 122 from: (a) original sequence (CIF, 25 Hz) and (b) reconstructed images at 125 kb/s with H264 coder, (c) with Galpin coder, and (d) with our coder.



Fig. 12. *Thabor* sequence: Reconstructed images 71 at 30 kb/s with (a) Galpin coder and with (b) our coder.

In Galpin’s method, depth maps were encoded as an image with EBCOT. The number of vertices in the uniform mesh is then reduced to be competitive with the rate achieved by our progressive coder (15 kb for 2400 vertices against 22 kb for 1600 vertices for Galpin’s coding for the stairs sequence). This profit allows to allocate more bitrate for texture information in order to better preserve texture details (as shown on Fig. 12 on the wall on the right or in the background on the image). Since texture information is prominent over geometry for low bitrates, this profit is particularly useful in order to achieve a very low bitrate.

### C. Scalability Results

Here we show some results of the scalability obtained with our coder. We show PSNR values even if it does not allow the evaluation of the quality of the reconstructed sequence, because of geometric distortion.

The table of the Fig. 13 shows the number of vertices and the associated rate depending on the level of the wavelet decomposition. Fig. 14 shows reconstructed images associated with these levels of decomposition. The size of the binary representation increases with the wavelet decomposition level, and so does the

Level	Number of Vertices	size (bits)
0	148	7744
1	565	10869
2	2185	17155
3	8736	29806

Fig. 13. Number of vertices and associated rate function of the level of the wavelet decomposition.

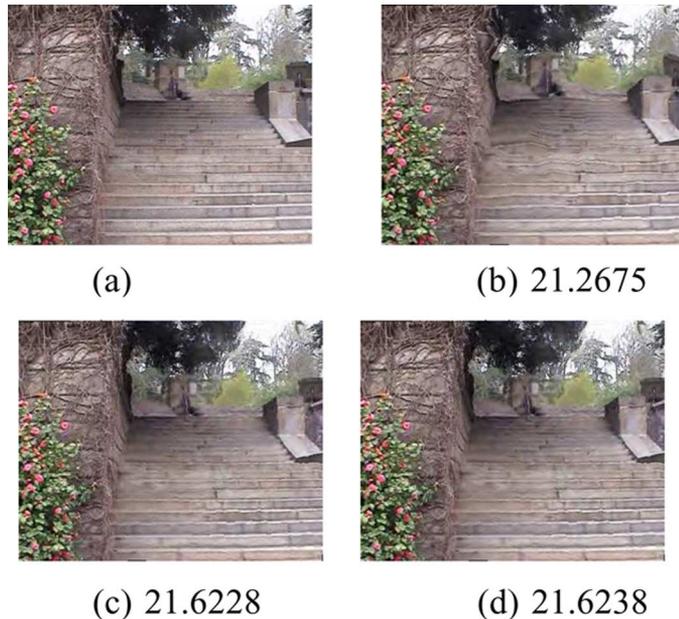


Fig. 14. *Thabor* sequence: spatial scalability Image 71 from the (a) original sequence (CIF, 25 Hz) and reconstructed images at 125 kb/s at different level of wavelet decomposition: (b) 0, (c) 2, and (d) 3.

quality of the reconstructed images. This is particularly visible on the steps of the stairs. For this sequence, the choice of level 2 seems to be a good rate/distortion tradeoff.

#### D. Virtual Navigation Results

Including inter-relations into the coder not only dramatically decreases compression rates, but it also increases the visual quality of the reconstructed sequence by linking up different models together.

Fig. 15 shows successive images around a transition between two GOFs, the last of the preceding GOF and the first of the following. One can see the discontinuity between two successive frames of the video on these images with the right-hand edge blank due to missing information for appearing areas. Fig. 15 shows also the same images reconstructed with our method. The artefacts are greatly reduced by the morphing enabled by model vertex matching.

Galpin 3-D fading [17] allows to reduce artefacts near model transitions and also produces ghost effects on the images of the middle of the GOF and on images associated to free viewpoints. The implicit morphing strongly contributes to the visual quality of the scene, avoiding these ghost effects while smoothing transitions between models.

Fig. 16 shows reconstruction results during free navigation, i.e., when the viewer is not restricted to the camera path defined

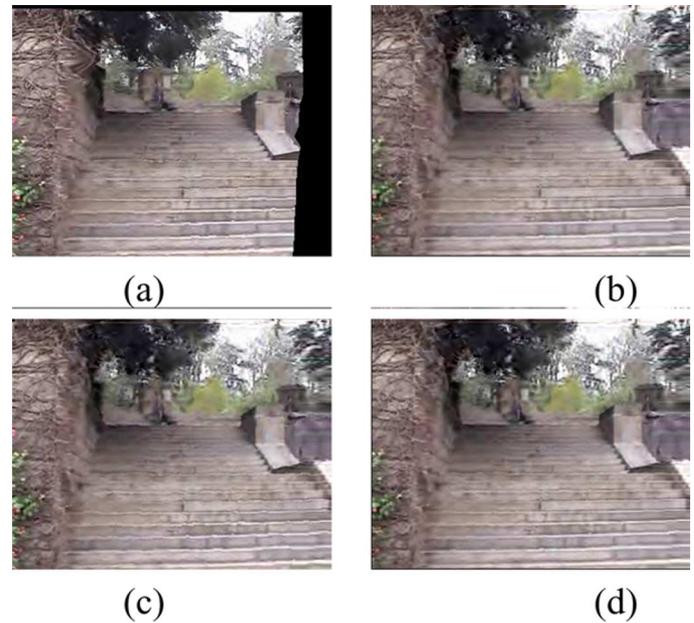


Fig. 15. *Thabor* sequence: Reconstruction of original path around a transition between two successive models. Successive reconstructed images without post-treatment (no morphing nos fading) (a) and (b). Successive reconstructed images with our coder (c) and (d).

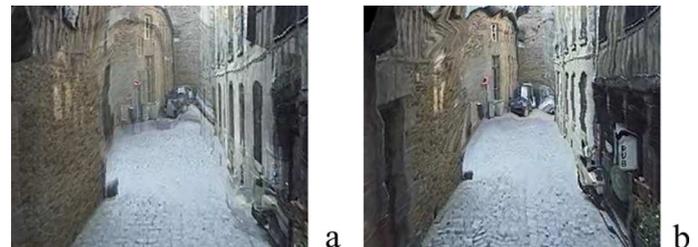


Fig. 16. Street sequence: Reconstruction on virtual path with (a) Galpin and (b) proposed method.

during capture. In a similar way, results on the original path visual quality of reconstructed images is increased by eliminating artefacts of ghost effects even though some geometric distortions are visible near the upper image border, due to nonuniform triangulation.

Our algorithm is, however, limited by occlusions and by 3-D informations and camera parameters precision. Our approach does not require an accurate geometry, and therefore, the models cannot be used to generate a free viewpoint far from the original camera.

## VII. CONCLUSIONS AND FUTURE WORK

We have presented a new model-based coding scheme for static video with fine-grain scalability, allowing content adaptation over a very wide spectrum of terminals and networks. This scheme takes advantage of specific video content, i.e., a fixed scene acquired with a moving camera, to build a 3-D representation which allows better performances and advanced functionalities. In particular, 3-D can be streamed adaptively in applications of free navigation over networks. The coder, showing better compression results and finer scalability than previous schemes, exploits all the power of second-generation wavelets

and implicit morphing thanks to the design of a connectivity mesh gathering each GOF connectivity information.

To further improve this scheme, it would be interesting to express the temporal increments in a wavelet basis themselves. Furthermore, reconstructed images have shown the need for a suitable error metric for reconstructed images taking into account the geometric distortion to meet visual quality measure. Finally, whereas encoding/decoding the representation can be done on-line, nonlinear optimizations for disparity estimation or bundle adjustment harm complexity of the 3-D extraction algorithm. It could be interesting to try new graphics processors to accelerate treatments to reach real-time applications such as video-conferencing.

#### REFERENCES

- [1] H. Schwarz and T. Wiegand, "The emerging jvt/h.261 video coding standard," in *Proc. IBC*, Amsterdam, The Netherlands, 2002.
- [2] F. Preteux and M. Malciu, "Model-based head tracking and 3D pose estimation," in *Proc. Visual Conf. Image Processing*, 1998, pp. 94–110.
- [3] B. Girod, "3D image models and compression—Synthetic hybrid or natural fit?," in *Proc. ICIP*, Kobe, Japan, Oct. 1999.
- [4] R. Koch, M. Pollefeys, and L. V. Gool, "Realistic surface reconstruction of 3D scenes from uncalibrated image sequence," *J. Visualiz. Comput. Anim.*, vol. 11, pp. 115–127, 2000.
- [5] M. Pollefeys, M. Vergauwen, F. Verbiest, K. Cornelis, and L. V. Gool, "From image sequences to 3D models," in *Proc. 3rd Int. Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space Images*, Ascona, Switzerland, Jun. 2001.
- [6] A. Zisserman, A. Fitzgibbon, and G. Cross, "Vhs to vrml: 3D graphical models from video sequences," in *Proc. IEEE Int. Conf. Multimedia Computing and Systems*, Jun. 1999, vol. 1, pp. 51–57.
- [7] D. Nister, "Reconstruction from uncalibrated sequences with a hierarchy of trifocal tensors," in *Proc. 5th Eur. Conf. Computer Vision ECCV'2000*, Dublin, Ireland, 2000.
- [8] K. Cornelis, M. Pollefeys, M. Vergauwen, and L. V. Gool, "Augmented reality using uncalibrated video sequences," in *Proc. 2nd Workshop on Structure from Multiple Images of Large Scale Environments*, Dublin, Ireland, 2000.
- [9] P. Alliez and C. Gotsman, "Recent advances in compression of 3D meshes," in *Proc. Symp. Multiresolution in Geometric Modeling*, Cambridge, U.K., 2003.
- [10] G. Taubin and J. Rossignac, "Geometric compression through topological surgery," *ACM Trans. Graph.*, vol. 17, no. 2, pp. 84–115, 1998.
- [11] J. Rossignac, "Edgebreaker: Connectivity compression for triangle meshes," *IEEE Trans. Vis. Comput. Graph.*, vol. 5, no. 1, pp. 47–61, Jan.–Mar. 1999.
- [12] Applications and Requirements for 3dav 2003, I. JTC1/SC29/WG11.
- [13] E. Lamboray, S. Würmlin, and M. Gross, "Real-time streaming of point-based 3D video," in *Proc. IEEE Virtual Reality Conf.*, 2004, pp. 91–98.
- [14] M. Magnor, P. Ramanathan, and B. Girod, "Multi-view coding for image based rendering using 3-D scene geometry," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 11, pp. 1092–1106, Nov. 2003.
- [15] P. Ramanathan and B. Girod, "Random access for compressed light fields using multiple representations," in *Proc. Conf. Computer Vision and Pattern Recognition*, Baltimore, MD, 2004.
- [16] F. Galpin, R. Balter, L. Morin, and K. Deguchi, "3D models coding and morphing for efficient video compression," in *Proc. Conf. Computer Vision and Pattern Recognition*, Baltimore, MD, 2004.
- [17] F. Galpin and L. Morin, "Sliding adjustment for 3D video representation," *EURASIP J. Appl. Signal Process., Special Issue on Signal Processing for 3D Imaging and Virtual Reality*, no. 10, pp. 1088–1101, 2002.
- [18] B. L. Guen, R. Balter, L. Morin, and P. Alliez, "Morphing de modes 3D estimes," *Journées D'études et D'changes CORESA'2004*, May 2004.
- [19] I. Daubechies, "Ten lectures on wavelets, society for industrial and applied mathematics ed," in *CBMS-NSF Regional Conf. Series in Applied Mathematics*, Philadelphia, PA, 1992, vol. 61.
- [20] M. Lounsbery, T. D. DeRose, and J. Warren, "Multiresolution analysis for surfaces of arbitrary topological type," *ACM Trans. Graph.*, vol. 16, no. 1, pp. 34–73, Jan. 1997.
- [21] A. Khodakovsky, P. Schroder, and W. Sweldens, "Progressive geometry compression," in *SIGGRAPH 2000 Conf. Proc.*, New Orleans, LA, Jul. 2000.
- [22] S. Paris, F. Sillion, and L. Quan, "A surface reconstruction method using graph cut optimization," *Int. J. Comput. Vis.*, vol. 66, no. 2, pp. 141–161, 2005.
- [23] S. Pateux, G. Marquant, and D. Chavira-Martinez, "Object mosaicking via meshes and crack-lines technique. Application to low bit-rate video coding," in *Proc. Picture Coding Symp.*, 2001.
- [24] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proc. 4th Alvey Vision Conf.*, Manchester, U.K., 1988.
- [25] P. Gioia, O. Aubault, and C. Bouville, "Real-time reconstruction of wavelet encoded meshes for view-dependent transmission and visualization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 7, pp. 1009–1020, Jul. 2004.
- [26] P. Schroder and W. Sweldens, "Spherical wavelets: Efficiently representing functions on the sphere," in *Proc. Siggraph 95*, 1995, pp. 161–172.
- [27] R. Balter, P. Gioia, L. Morin, and F. Galpin, "Scalable and efficient coding of 3D model extracted from a video," in *Proc. 2nd Int. Symp. 3DPVT (3D Data Processing, Visualization, and Transmission)*, Thessaloniki, Greece, Sep. 2004.
- [28] A. Saïd and W. A. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 3, pp. 243–250, Jun. 1996.



coding, computer vision, and 3-D television.



transmission of multirepresentation collaborative environments. Since 2000, he has actively participated in the MPEG-4 standardization of the SNHC group.



**Raphaële Balter** was born in Nancy, France. She received the Dipl. Ing. degree in electronics and computer science from Ecole Polytechnique de l'université de Nantes, Nantes, France, and the Ph.D. degree in signal processing from University of Rennes, Rennes, France, in 2005. Her Ph.D. thesis was prepared at IRISA/INRIA Rennes and France Telecom R&D Rennes.

She is currently a Research Engineer in the TECH/IRIS Laboratory, France Telecom R&D, Cesson-Sevigne, France. Her research interests include video

**Patrick Gioia** received the M.S. degree in mathematics in 1997 and the Ph.D. degree in computer science in 2001, both from the University of Rennes, Rennes, France.

He has been a Research Engineer with France Telecom R&D, Cesson-Sevigne, France, since 1997. His main research interests include computational geometry, compression of 3-D surfaces, multiresolution representation of 3-D models, and visualization of large databases over network. His current research activities focus on compression methods for efficient

**Luce Morin** was born in Grenoble, France. She received the engineer degree in physics and digital images from ENSPS, Strasbourg, France, in 1989 and the Ph.D. degree from INPG, Grenoble, in 1993.

Since then, she has been an Assistant Professor at the University of Rennes, Rennes, France, where she teaches computer science, image processing, and computer vision. Her research activities, as a member of the Temics Project in the IRISA/INRIA Rennes Laboratory, deal with 3-D modelization for video sequence communication.