Representation of Temporal Changes of Flexible 3–D Objects

Oliver Grau

Institut für Theoretische Nachrichtentechnik und Informationsverarbeitung
Universität Hannover, Appelstr. 9A, 30167 Hannover, Germany
Email: grau@tnt.uni–hannover.de

ABSTRACT

A representation for 3–D objects and their temporal changes is presented. The representation was developed in the ACTS PANORAMA project for the 3–D modelling of communication scenes. The shape of the scene objects is represented by triangular meshes. The object can be modelled hierarchically with partial rigid or flexible components. Further more, temporal changes of the scene objects caused by object motion are handled by an incremental data stream. The static models and temporal changes can be exchanged between image analysis modules using a special file format and exchange protocol. Finally a converter is described, that filters the relevant data and visualizes the 3–D model and temporal changes using a VRML 2.0 viewer.

1. INTRODUCTION

The goal of the PANORAMA [1] setup is to provide telepresence for the system user. A typical scene that is processed with the PANORAMA modelling system is depicted in figure 1. The parameters of the objects to be modeled will be estimated from a trinocluar input image sequence captured with three fixed cameras. The image analysis modules currently under development in PANORAMA should automatically create a parametric 3–D scene description. This description is transferred to the receiver and visualized on a (auto–)stereoscopic display.

The scene description is composed of model objects defined by their 3–D shape, 3–D motion and surface texture parameters [3]. The shape of the objects is represented by triangular meshes. Additionally for a subdivision into components a hierarchical object representation is needed as depicted in figure 2. The scene description sd(k) contains a complete parametric description for a time instance k. The following section explains the static representation of the scene.

In progression of time the image analysis system is able to analyse temporal changes of the scene that must be represented and stored in the scene description. The changes concern: the model geometry (shape), topology and texture (color). A change in the geometry can be caused by motion estimation or shape update. In the first case the motion is described as a translation and rotation of the hierarchical object components. In the case of a shape update a list of new coordinates is generated. For the storage and transfer of these temporal changes a stream concept was developed. The section 3 explains this concept.

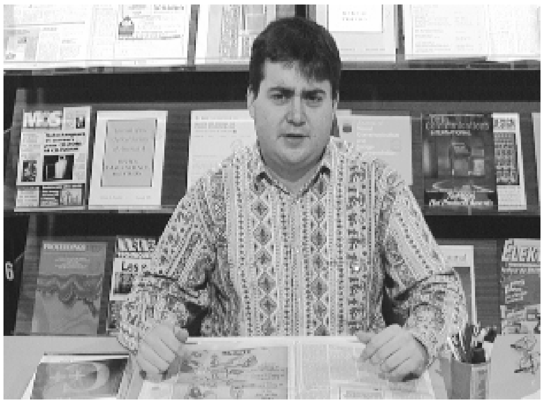


Figure 1 PANORAMA test sequence "Ludo–5"

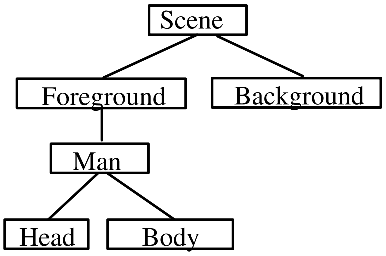


Figure 2 Object components can be grouped hierarchically. The scene description sd(k) contains the shape, texture and motion parameters at time instance k

submitted paper for: International Workshop on Synthetic–Natural Hybrid Coding and Three–Dimensional (3D) Imaging '97

2.Static Representation of Flexible 3–D Objects

For flexible or partially flexible objects like human beings it is not possible to distinguish clearly between the moving part and the part caused by a flexible shape modification. For a practical computation of both parts it is useful to assume componentwise rigid surfaces to first compute the motion [5] and then compute the shape changes.

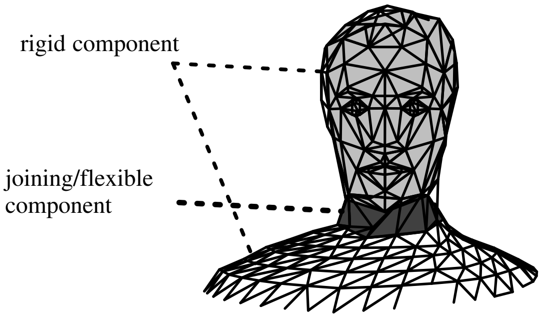


Figure 3 flexible and rigid components

As shown in figure 3 the head of a human and his torso can be assumed to be inherently rigid, that means the motion of all triangles of one of the components can be described with the same motion parameters. At the connection of two surfaces with different motion parameters, a flexible surface connects both rigid components. In the example in figure 3 the neck of the person is assumed as such a joining flexible surface.

A joining component does not move independently. For this reason it uses only vertex points that are shared with the neighboring rigid surfaces. Each triangle of a flexible surface has two points common with one of the rigid surfaces and one point with the second rigid surface. In case of motion update only the rigid surfaces are updated. The flexible surface is indirectly changed because its vertices follow the neighboring surfaces. In general this modification leads to a change of the triangle shapes but the surface is always closed.

The surface color of the 3–D models is stored in texture maps. Texture maps (and other optional parameters) are handled as soft attributes, since these parameters are optional and not necessary for all modules of the image analysis. A soft attribute can be assigned to any scene object. Further for the retrieval of soft attributes inheritance is implemented over the scene hierarchy. With this feature texture maps can be assigned trianglewise, i.e. one map per triangle or for a group of triangles (e.g. one map for node ”Head” and one for node ”Body” in figure 2). The inheritance is used for other parameters, like hierarchical motion parameters or auxiliary information.

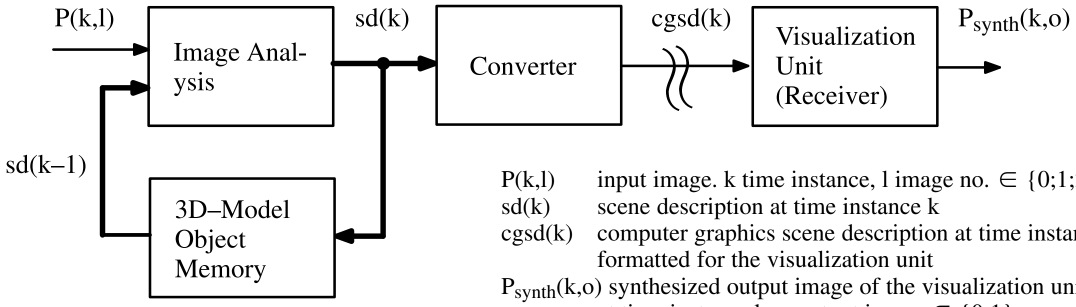
3.A Stream Concept for Handling of Temporal Changes

The 3–D scene description is used internally for image analysis. In addition the relevant parts of this description are converted into a standard computer graphic file format and transferred to the computer unit for visualization, as shown in figure 4. The scene description sd(k) used in the image analysis contains some more information about the scene, like local motion parameters that are useful for the visualization. A converter filters out the relevant data cgsd(k) which is transferred employing an information channel to the receiver. At the current state of the implementation VRML 2.0 is used as the file format [2] and the transfer is realized by regular files. Later the transfer should be realized by streams. The receiver system uses a standard VRML 2.0 viewer.

The temporal changes of the scene description can be formulated differential:

$$sd(k-1) + \Delta sd(k) \rightarrow sd(k) \qquad (1)$$

The differential changes Δsd(k) are created by the image analysis. For the transfer to a receiver only this



P(k,l) input image. k time instance, l image no. ∈ {0;1;2}
sd(k) scene description at time instance k
cgsd(k) computer graphics scene description at time instance k formatted for the visualization unit
$P_{synth}(k,o)$ synthesized output image of the visualization unit at time instance k, o output image ∈ {0;1}

Figure 4 Data Flow in PANORAMA

submitted paper for: International Workshop on Synthetic–Natural Hybrid Coding and Three–Dimensional (3D) Imaging ’97

differential changes must be considered and transferred as a data stream. The initial scene description $sd_{initial}$ can also be formulated as the transition from time instance k=–1 to k=0 with $\Delta sd(0) = sd_{initial}$.

The data stream is realized by a programming interface and implemented as a C++ class library. The data exchange uses a special file format (PANORAMA file format) and protocol.

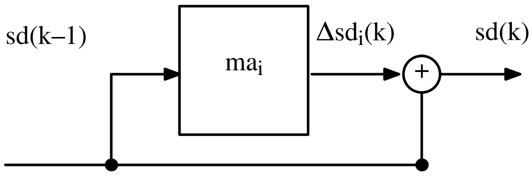


Figure 5 Module ma_i, $i \in \{1;..;N\}$ of analysis system generates contribution $\Delta sd_i(k)$

The stream concept is further used for the data exchange between the different modules of the image analysis. A module ma_i generates a contribution $\Delta sd_i(k)$ to the temporal changes (figure 5) that is propagated to a succeeding module ma_{i+1}.

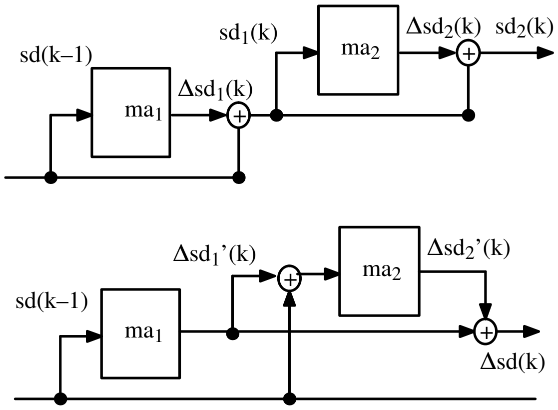


Figure 6 Cascading modules ma_i of the analysis system. Exchange of complete scene description (top). Exchange of differential contributions (bottom)

The image analysis system is composed of a number of modules ma_i, $i \in \{1;..;N\}$. Two different methods of cascading the single modules can be distinguished as depicted in figure 6: a) the complete scene description $sd_i(k)$ is transferred between the modules (top drawing in figure 6) or b) only the changes $\Delta sd_i'(k)$ of each module are propagated (bottom drawing in figure 6). The first scheme has the advantage that the modules are completely decoupled from each other. Each module creates a local copy of the current scene description in the memory and adds the differential changes $\Delta sd_i(k)$ from the input stream. This mode is used for the development of the system because the modules can be tested independently from each other. In this case the data exchange between the modules is performed using files.

The scheme b) will be used in the final version of the system when the development of the modules is finished. In this case only one global scene description is generated in the memory and the amount of data exchange between the modules is significantly increased.

3.1 Scene Description Updates

The differential scene updates $\Delta sd(k)$ concern the model geometry, topology and texture (color). On geometry and texture updates the corresponding parameters are transferred.

Changes in the topology of an object or object component are currently handled by a complete transfer of the changed objects and its dependant hierarchical components.

3.2 Addressing Objects

The differential scene update stream is accumulative. For an efficient data transfer of the scene updates $\Delta sd(k)$ the sender of the update must be able to address objects in the receiver. The object addressing must be unique during the life time of the stream and concerns the internal data representation and the representation in the visualization unit. For this reason each data object contains a unique identification number. Because the data objects can be created in different processes the allocation of the numbers is handled by the object stream. That means image analysis modules request for identification numbers from the PANORAMA data stream.

3.3 Time Stamps

Each transferred object or update contains a time stamp. The time stamps are employed for synchronization if more than one data stream is combined into a scene description or for the identification in an analysis module that considers temporal changes over a time interval.

4.Mapping the PANORAMA Scene Description to VRML 2.0

The data representation used in the analysis modules covers different and additional features than used in the visualization system. Therefore a filter selects the

submitted paper for: International Workshop on Synthetic–Natural Hybrid Coding and Three–Dimensional (3D) Imaging '97

relevant information and converts it into the VRML 2.0 file format. The VRML 2.0 specification [2] covers the definition of static 3–D models with shape and textures. Further it supports object motion and shape updates. This features are used at the current state of the implementation for the visualization of moving objects and object components and their flexible deformations.

5.Results

Figure 7 shows a VRML 2.0 model of the scene ”Ludo–5” that was captured using the PANORAMA setup and image processing modules [6],[4],[5]. The image processing modules are implemented using the data structures and according library described in this contribution. Each module is realized as a separate program and the data exchange is realized using a C++ interface to the PANORAMA file format for static and temporal updates as described in section 3.

Figure 7 VRML 2.0 model ”Ludo–5” with motion parameters. (Created using image analysis modules [6],[4],[5])

At the current state of the implementation of the VRML 2.0 interface only motion and shape updates can be visualized. For temporal changes in structure like the transfer of new triangles or the update of texture, an extension of VRML 2.0 is needed. Such an extension would contain a protocol for the transfer of temporal changes from the image analysis to the viewer and the implementation of a parser in the viewer that is capable of handling the incoming data.

6.Conclusions

A representation for 3–D objects and their temporal changes was presented. The representation was developed for the use in the ACTS PANORAMA project for the 3–D modelling of scene objects. Typically scene objects like human beings are modeled with partial rigid and flexible components using a generic description with attributed triangular meshes. Explicit parametric models, like parametric human models could be realized on top of the developed representation as specialized C++ classes and make use of the data exchange via the stream concept.

For the final visualization of the 3–D models a standard VRML 2.0 viewer is used. VRML 2.0 doesn’t support direct dynamic scene update reading information from a stream and mixing of different stream sources. For a realization of these features some specific extensions must be implemented in the viewer and a protocol for the transfer must be specified. The standardization of these topics could probably be covered by MPEG–4 SNHC.

7.References

[1] ACTS AC–092 PANORAMA, Research Project funded by the European Community, http://www.tnt.uni–hannover.de/project/eu/panorama/

[2] VRML Architecture Group, VRML – Version 2.0 Specification, http://vag.vrml.org/VRML2.0/FINAL/

[3] Grau, O., ”Definition of 3–D Model Objects”, ACTS–PANORAMA Project Deliverable AC092/UH/DS/P/004, Aug. 1996.

[4] Riegel Th., Manzotti, R., Pedersini, F., ”3–D shape approximation for objects in multiview image sequences”, Int. Workshop on Synthetic – Natural Hybrid Coding and Three Dimensional Imaging (IWSNHC3DI’97). Sept.97 Rhodes

[5] Falkenhagen, L., ”3–D Motion Estimation of 3–D Model Objects”, ACTS–PANORAMA Project Deliverable AC092/UH/DS/P/006, Nov. 1996.

[6] Falkenhagen, L., ”Block–Based Depth Estimation from Image Triples with Unrestricted Camera Setup”, IEEE MMSP workshop, June 1997, Princeton, New Jersey, USA

submitted paper for: International Workshop on Synthetic–Natural Hybrid Coding and Three–Dimensional (3D) Imaging ’97