VIRTUAL AND REAL HUMANS INTERACTING IN THE VIRTUAL WORLD

Daniel Thalmann
Ronan Boulic
Zhyong Huang
Hansrudi Noser

Computer Graphics Lab
Swiss Federal Institute of Technology
CH 1015 Lausanne, Switzerland
E-mail: thalmann@lig.di.epfl.ch

Abstract
In this paper, we discuss the role of digital actors in Virtual Environments. We describe the integration of motion control techniques with autonomy based on synthetic sensors. In particular, we emphasize the synthetic vision, audition and tactile system. We also discuss how to introduce the sensors of real humans in the Virtual Space in order to have a communication between digital actors and real humans.

1. Human Communication in the Virtual World

For Virtual Reality and interaction in Virtual Worlds, there is a need for autonomous virtual humans, reacting to environments and making decisions based on perception systems, memory and reasoning. This need to have autonomous virtual humans arises from two considerations:

- in Virtual Worlds, the more autonomous behaviour that is built into the virtual humans, the less extra work there is to be done by the operator to create the simulation

- in interactive games, autonomous human-like behaviour is necessary in order to maintain the illusion in the user that the virtual humans are real ones.

Two different kinds of interaction could be addressed: interaction between virtual humans, and interaction between virtual humans and real ones, a typical VR situation. These two kinds of interaction are sufficiently different to require different technical solutions. A pair of virtual humans (Fig.1) interacting is a closed system which can be developed by equipping the virtual humans with complementary behaviours. In order to support interaction and communication, virtual humans should be equipped with the ability to `recognize' other virtual humans and `perceive' their facial expressions, gestures and postures. There is no need for real recognition or perception, of course, because information from the data structures that define these behaviours in one virtual human can be passed directly to a second virtual human.

For an interaction between a virtual human and a real one, there is no possibility of transferring data structures, and image understanding methods are required to provide the virtual human with a perception of the real human's behaviour. True interaction between the virtual and the real humans requires a two-way communication between them at the geometric level, at the physical level, and at the behavioral level [4]. At the geometric level, 3D devices like a DataGlove allow the real human to communicate any geometric information to the virtual one. At the physical level, using a force transducer, a force or a torque may be communicated to a virtual human who can apply a force that may be felt by the real human using a force feedback device [2]. It is for example possible to simulate the VR scene where the animator and the virtual human tug on the two ends of a rope. At the behavioral level, we consider emotional communication between the virtual human and the real one. Such an experience is described by Pandzic et al. [8].

Fig.1. Two virtual humans in the Virtual World

It is also important to investigate and prototype human body models as part of the visual feedback to the human participant. This is the Self-representation in the Virtual World. Even with limited sensor information, a virtual human frame can be constructed in the Virtual World, that reflects the activities of the real body. Slater [14] indicates that such a body, even if crude, heightens the sense of presence. This virtual human should reflect the activities of the real body. Moreover, we may expect in the future very realistic human figures, allowing a perfect representation of self. Although, it could be represented as very abstract, it may also look like a human body depending on the application.

2. Autonomous Virtual Humans

2.1 Behavioral Animation and Artificial Life

For VR and multimedia applications, digital actors should be able to react to their environment and take action based on perception. In our approach, perception is based on synthetic or virtual sensors as described in the next section. With such an approach, we should be able to create simulations of situations such as digital actors moving in a complex environment they may know and recognize, or playing ball games based on their visual and touching perception.

This kind of research is strongly related to the research efforts in behavioral animation as introduced by Reynolds [10] to study in his distributed behavioral model to simulate flocks of birds, herds of land animals and fish schools. For birds, the simulated flock is an elaboration of a particle system with the simulated birds being the particles. A flock is assumed to be the result of the interaction between the behaviors of individual birds. Working independently, the birds try both to stick together and avoid collisions with one another and with other objects in their environment. In a module of behavioral animation, positions, velocities and orientations of the actors are known from the system at any time. The animator may control several global parameters: e.g. weight of the obstacle avoidance component, weight of the convergence to the goal, weight of the centering of the group, maximum velocity, maximum acceleration, minimum distance between actors. The animator provides data about the leader trajectory and the behavior of other birds relatively to the leader. A computer-generated film has been produced using this distributed behavioral model: *Stanley and Stella.* Haumann and Parent [2]

describe behavioral simulation as a means to obtain global motion by simulating simple rules of behavior between locally related actors. Wilhelms [16] proposes a system based on a network of sensors and effectors. Ridsdale [12] proposes a method that guides lower-level motor skills from a connectionist model of skill memory, implemented as collections of trained neural networks. We should also mention the huge litterature about autonomous agents [3] which represents a background theory for behavioral animation and the new artifical life concepts. More recently, genetic algorithms were also proposed by Sims [13] to automatically generate morphologies for artificial creatures and the neural systems for controlling their muscle forces. Tu and Terzopoulos [15] described a world inhabited by artificial fishes.

4.2. Synthetic sensors

In our approach, digital actors are equipped with visual, tactile and auditory sensors. These sensors are used as a basis for implementing everyday human behaviour such as visually directed locomotion, handling objects, and responding to sounds and utterances..

Synthetic vision

The most important perceptual subsystem is the vision system. It provides the actor with a realistic information flow from the environment. With synthetic vision, we do not need to address these problems of recognition and interpretation. We first introduced the concept of synthetic vision [9] as a main information channel between the environment and the digital actor. Reynolds [1] more recently described an evolved, vision-based behavioral model of coordinated group motion. Tu and Terzopoulos [15] also proposed artificial fishes with perception and vision.

In our synthetic vision, each pixel of the vision input has the semantic information giving the object projected on this pixel, and numerical information giving the distance to this object. So, it is easy to know, for example, that there is a table just in front at 3 meters. With this information, we can directly deal with the problematic question: "what do I do with such information in a simulation system?" Our approach is based on Displacement Local Automata. A DLA is a black box which has the knowledge allowing the digital actor to move in a specific part of his environment. The controller is the thinking part of our system; it makes decisions and performs the high-level actions. In an unknown environment, it analyzes this environment and activates the right DLA. In the simple case of a known environment, the controller directly activates the DLA associated with the current location during the learning phase. From information provided by the controller, a navigator builds step by step a logical map of the environment.

More complex problems come when the digital actor is supposed to know the environment, which means the introduction of a digital actor memory. Using his vision, the digital actor sees objects and memorize them, based on an octree representation [6]. Then, he may use this memory for a reasoning process. For example, a recursive algorithm allows a path to be found from the digital actor to any position avoiding the obstacles based on his memory. The digital actor should also be able to remember if there is no path at all or if there are loops as in a maze. Once a digital actor has found a good path, he may use his memory/reasoning to take the same path. However, as new obstacles could have been added on the way, the digital actor will use his synthetic vision to decide the path, reacting to the new obstacles.

To illustrate the capabilities of the synthetic vision system, we have developed several examples. First, a digital actor is placed inside a maze with an impasse, a circuit and obstacles. The digital actor's first goal is a point outside the maze. After some time, based on 2D heuristic, the digital actor succeeds in finding his goal. When he has completely memorized the impasse and the circuit, he avoided them. After reaching his first goal, he had nearly complete visual octree representation of his environment and he could find again his way without any problem by a simple reasoning process. A more complex example is concerned with the simulation of vision-based tennis playing [7]. We use the vision system to recognize the flying ball, to estimate its trajectory and to localize the partner for game strategy planning.

Synthetic audition

For auditive aspects, we developed a framework for modeling a 3D acoustic environment with sound sources and microphones. Now, our virtual actors are able to hear [7]. Any sound source (synthetic or real) should be converted to the AIFF format and processed by the sound renderer. The sound renderer takes into account the real time constraints. So it is capable to render each time increment for each microphone in "real time" by taking into account the final propagation speed of sound and the moving sound sources and microphones. So, the Doppler effect, for example, is audible. The acoustic environment is composed of sound sources and a propagation medium. The sound sources can produce sound events composed of a position in the world, a type of sound, and a start and an end time of the sound. The propagation medium corresponds to the sound event handler which controls the sound events and transmits the sounds to the ears of the actors and/or to a user and/or a soundtrack file. We suppose an infinite sound propagation speed of the sound without weakening of the signal. The sound sources are all omnidirectional, and the environment is non reverberant.

Synthetic tactile

The simulation of the touching system consists in detecting contacts between the digital actor and the environment using multisensors. Multi-sensors are considered as a group of objects attached to the articulated figure. A sensor is activated for any collision with other objects or sensors. Here we select sphere sensors for their efficiency in collision detection. In our work, the sphere multi-sensors have both touch and length sensor properties, and have been found very efficient for synthetic actor grasping problem. Each sphere sensor is fitted to its associated joint shape, with different radii. This configuration is important in our method because, when a sensor is activated in a finger, only the articulations above it stop moving while others can still move. By doing this way, all the fingers are finally positioned naturally around the object, as shown in Figure 2.

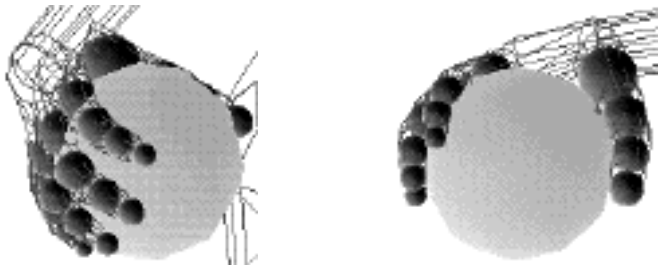

Fig.2. One example shown sensors in grasping

When grasping a free form surface object, the sphere sensors are detecting collision with the object. After the hand center frame aligns with object frame, the fingers are closed according to the different strategies, e.g. pinch, wrap, lateral, etc., while sensor-object and sensor-sensor collisions are detected. The grasping strategies is based on a method introduced by Mas and Thalmann [5]. When one sensor is activated, all articulations above it are blocked. The grasping is completed when the remaining sensors are activated or the joints reach their limit. The final arm moving problem is then performed using inverse kinematics.

3. Perception of Participants in Virtual Environments

3.1 Vision in Virtual Environments

We have seen that virtual vision can be a powerful tool in modeling virtual autonomous actors in virtual worlds. Such actors in virtual worlds can have different degrees of autonomy and different sensing channels to the environment where they behave in a certain manner. In robotics for example, the agent (the robot) only gets information of the world by his sensors.

If he has a vision sensor, he has to extract all the semantic information of the world from an image. This is a very difficult task and thus, according to the actual state of knowledge, his intelligence is very restricted and his behavior is limited to some navigational tasks by avoiding collisions.

In virtual worlds the situation is different as we can provide some extra information to the actors making him more intelligent and faster. Until now, we tried to make the actors completely independent of the virtual worlds' internal representation and they only got the vision image and their position as sensory information. Thus, vision based navigation, collision avoidance, visual memory and tennis playing could be successfully modeled. We now integrate actors in virtual reality (VR) where real time constraints demand fast and intelligent reactions of actors with a set of elementary actions like grasping objects, sitting on chairs, jumping over obstacles, pressing buttons, running, ... To reach this goal we model a certain type of virtual world representation where the actor maintains a low level fast synthetic vision system but where he can access some important information directly from the environment without having to extract it from the vision image.

A human being can participate in VR by the head-mounted display and the earphones. He cannot get any internal VR information. His only source of knowledge from the VR is communicated by the vision and the sound (and perhaps some touching sensory information). His behavior is strongly influenced by this sensory input and his proper intelligence. In order to process the virtual actor vision in a similar way than the vision of the participant, we need to have a different model. In this case, the only information obtained by the virtual actor will be the vision image with the z-buffer values and the shaded and colored pixels (he may also get the sound signal and some touch sensor information). Such a virtual actor would be independent of each VR representation (as a human too) and he could in the same manner communicate with human participants and other virtual actors.

3.2 Audition in Virtual Environments

For virtual audition, we encounter the same problem as in synthetic vision. The real time constraints in VR demand fast reaction to sound signals and fast recognition of the semantic it carries. Thus, we plan in a first step to model a sound environment where the synthetic actor can directly access to positional and semantic sound source information of a audible sound event. This allows him to localize and recognize one or more sound sources in a reliable way and to react immediately.

This access to the sound environment representation, however, makes him dependent of it and lets the communication problem with human participants in VR unresolved. That is why, we try to realize a really independent actor as already mentioned above. This type of actor will get the same sound signal (digitized) as any other human participant in VR through his earphones. From this sound signals (stereo) the actor can estimate the position of a sound source and with an added speech recognition module he should be capable to extract some semantic information of some spoken language. Thus the synthetic actor should be able to understand and speak a reduced set of vocabulary allowing him also to communicate with human participants in VR.

3.3 Tactile in Virtual Environments

Concerning virtual sense of touch, we have already implemented a case of 3D interaction with VR technology. The participant may place an object into the Virtual Space using the CyberGlove and the virtual actor will try to grasp it and put it on a virtual table for example. The actor interacts with the environment by grasping the object and moving it. At the beginning of interactive grasping, only the hand center sensor is active. The six palm values from CyberGlove are used to move it toward the object. Inverse kinematics is used to update the arm postures from hand center movement. After the sensor is activated, the hand is close enough to the object final frame. The hand center sensor is deactivated and multi-sensors on hand are now used, to detect sensor object collision. The following process is similar to the multi-sensor

method discussed in Section 3. The major difference is that the grasping strategy is defined interactively. Our integrated virtual and real multi-sensor approach allow us to process communication between the real actor and virtual actors (Figure 3) as well as communication between virtual actors (Figure 4). More details on the implementation may be found in section ….

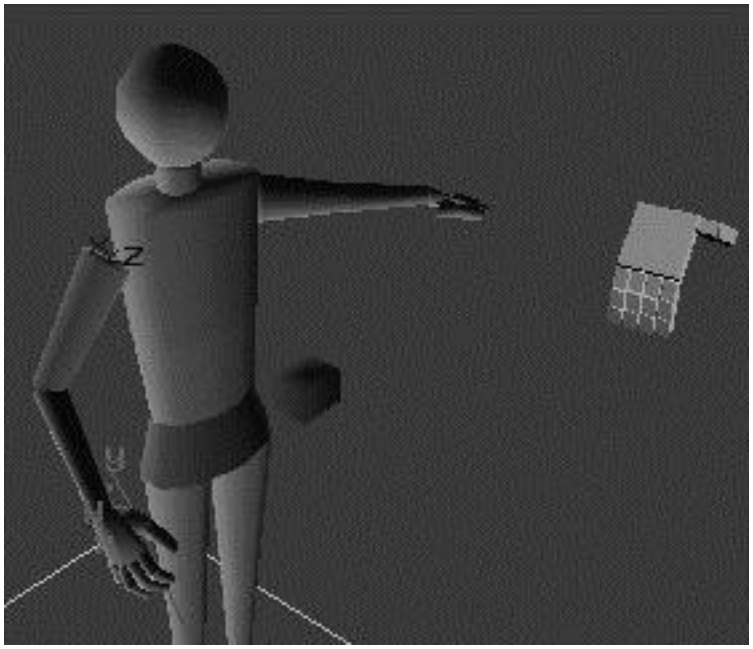

Fig.3. 3D interactive grasping with DataGlove

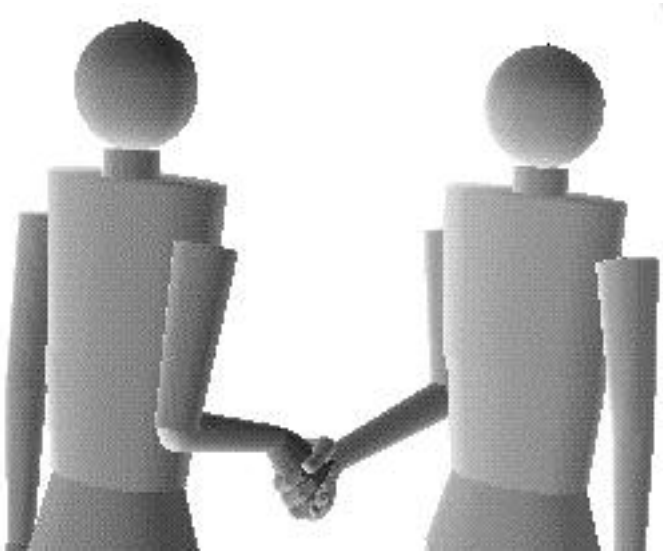

Fig.4. Communication between virtual actors

4. Architecture of the system

4.1 The Hierarchy of Modules

Our system is based on four main modules corresponding to four hierarchical levels, as shown in Fig.5.

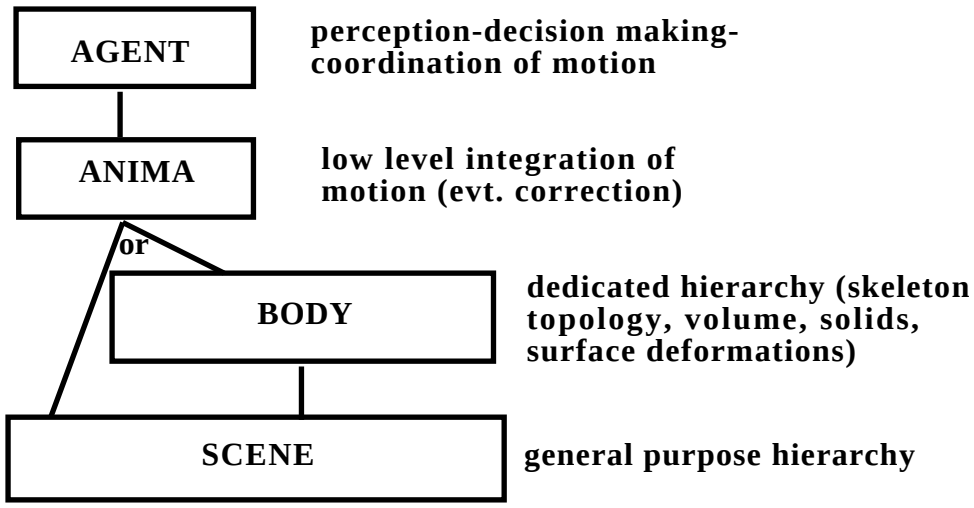

Fig.5. Different levels of information management from SCENE to AGENT

The SCENE module is dedicated to the design of general purpose 3D hierarchy and the handling of a flexible representation of motion propagation.

The BODY module is dedicated to the design of a specialized 3D hierarchy with a fixed skeleton-like topology. The purpose of the `BODY` data structure is to maintain a topological tree structure for a vertebrate body with predefined mobility, a corresponding volume discretisation with mass distribution and a corresponding envelope. A general mechanism allows to customize the skeleton structure at two levels, either at a high level with a small set of scaling parameters or at the low level of the position and orientation of the various articulations defined in the `SKELETON` data structure. In both cases the modifications are propagated to the lower level structure of the volume and envelope. A deformation function can be trigerred to compute skin surface according to the body posture. A `BODY` is the only entity which can compute the deformations of its external surface. This is the most computationally expensive process of an animation with `BODY` entities. The body deformation is based on current position and joint angles of the skeleton. We use a layered model based on 3 interrelated levels:

- the underlying articulated skeleton hierarchy composed of only articulated line segments whose movements are controlled with the `JOINT` data structure. It may be animated using motion generators.
- a layer is composed of metaball primitives attached to the `JOINT` of the skeleton. By transforming and deforming the metaballs, we can simulate the gross behavior of bones and muscles.
- the skin surface of the body automatically derived from the position and shape of the first and second layer. Internally, we define every part of the body as a set of B-spline patches, then tessellate the B-spline surfaces into a polygonal mesh for smooth joining different skin pieces together and final rendering.

The purpose the ANIMA module is to manage the integration of various sources of motion for a BODY data structure in particular or more generally for 3D hierarchical entities. An ANIMA data structure is designed to carry on that function. An application dedicated to the animation of a complex environment with multiple human figures will manage as many ANIMA data structures as animated human figures and animated sub-hierarchies. An ANIMA maintains and coordinates various entities. The most important one is the GENERATOR. This generic entity is designed to facilitate the plug in of various motion control modules into a common framework for motion integration. The following control modules have been implemented: keyframe, inverse kinematics, direct dynamics, walking and grasping. The ANIMA module is an integration scheme of various motion GENERATORs, it does not handle directly the function of producing various kind of motion; it just mixes them and is responsible of the final

update at the scene level. The distributed scheme of animation is necessary in the perspective of behavioral animation where each agent acts autonomously. In such a way, the ANIMA data structure handles all the necessary information for the processing of the motion of the agent. As a simple way to capture the purpose of the ANIMA with respect to the AGENT, we can say that :

an ANIMA is responsible of the "reflex control".
an AGENT is responsible of the "reflexive control"

However, the ANIMA data structure and integration scheme remains at the low level of control. Higher level control paradigm should be explicitly managed in a higher level entity which we will therefore refer to as an AGENT. Basically, an AGENT is responsible of organizing and handling the PERCEPTION information, deriving the DECISION making process and managing the COORDINATION of motion within an ANIMA entity.

The AGENT module provides high-level functions for the behavior and autonomy of digital actors. Future developments are planned in this module. Today, it already provides the following features: vision, auditory and tactile sensors, navigation and a few task-level functions supported at a lower level by the ANIMA module, especially for walking and grasping. Figure 6 shows the main group of functions in the AGENT module.

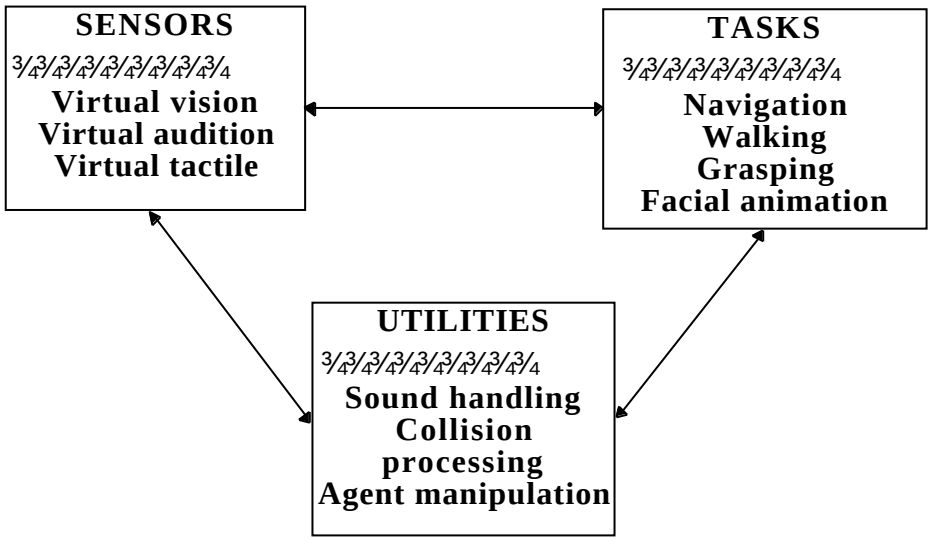

Fig.6. Functionalities of the AGENT module

4.2 IPC Server

A special IPC-Server is implemented as part of LIG 5D Toolkit that supports the application with 3D DataGlove and CyberGlove and head-mounted display. The system configuration is shown in Figure 7. The IPC-Server supports network computing so that our motion control system TRACK and the IPC-Server can run on different workstations. In our case, we are only interested in using the derived data from hand posture input. There are totally 16 data from DataGlove, 10 from finger flexion with 2 for each fingers, and 6 from the palm position and orientation. There are 3 flexing DOFs of each finger most significant to our multi-sensor based grasping method, more than those from DataGlove. We solve this problem by the fact that the flexing DOFs on each finger are not independent to move, especially the second and third DOFs. By putting these two in a linear function, the number of DOFs is 10. The 10 data from DataGlove finger bending are ranged from 0 to 90 degree, but the respective DOFs of the hand model have different range. So before using the data from DataGlove, we perform the proper scaling to guaranty the consistency of the real and the virtual hand postures.

The system flow is shown in Figure 8. For interactive grasping six palm values from DataGlove are used to move it towards the object and close the fingers at the final step.

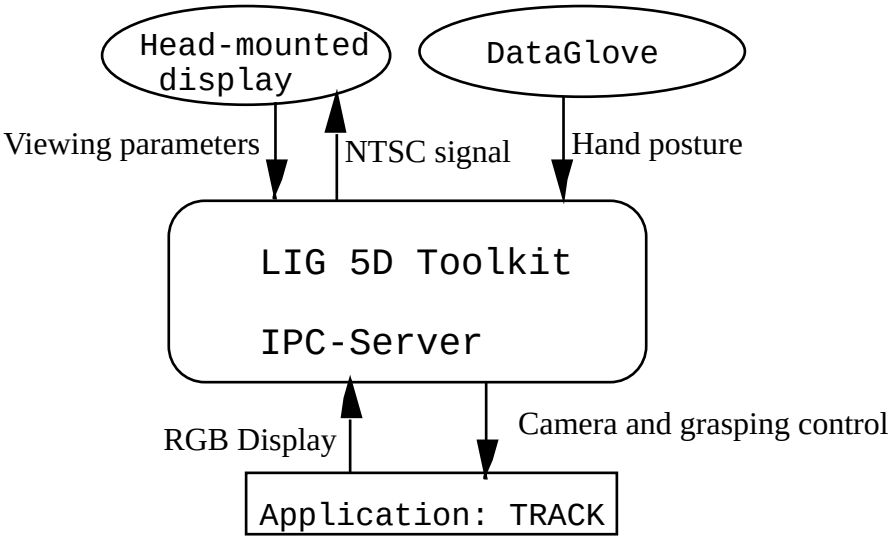

Fig.7. the framework of interactive grasping with DataGlove

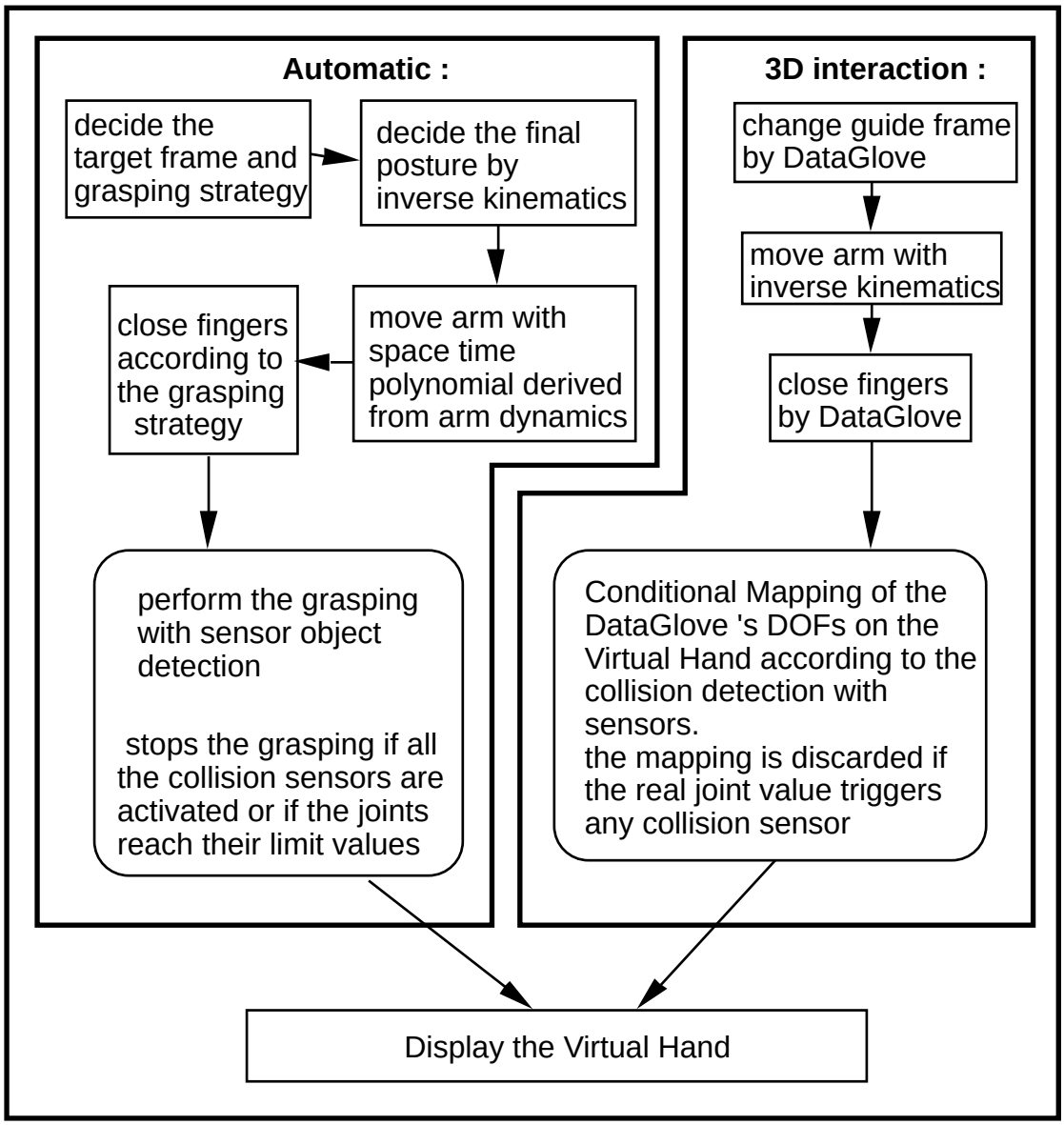

Fig.8. The system flow

5. Conclusion

We have shown in this paper how to create autonomous virtual humans and make them communicate each other and with real humans using vision, audition and tactile perception. We are now developing interaction between a real human with real sensors (Flock of Birds) and digital actors with synthetic sensors.

6. Acknowledgments

The authors are grateful to Shen Jianhua, Tom Molet and Serge Rezzonico for their contribution. The research was sponsored by the Swiss National Research Foundation, the Federal Office for Education and Science and the ESPRIT projects HUMANOID and HUMANOID-2.

7. References

1. Bergamasco M (1994) Manipulation and Exploration of Virtual Objects, in: Magnenat Thalmann N, Thalmann D (eds) Artificial Life and Virtual Reality, John Wiley.
2. Haumann DR, Parent RE (1988) The Behavioral Test-bed: Obtaining Complex Behavior from Simple Rules, The Visual Computer, Vol.4, No 6, pp.332-347
3. Maes P (ed.) (1991) “Designing Autonomous Agents”, Bradford MIT Press.
4. Magnenat Thalmann N, Thalmann D (1991) Complex Models for Animating Synthetic Actors, IEEE Computer Graphics and Applications, Vol.11, No5, pp.32-44.
5. Mas SR, Thalmann D (1994) A Hand Control and Automatic Grasping System for Synthetic Actors, Proceedings of Eurographics '94, pp.167-178.
6. Noser H, Renault O, Thalmann D, Magnenat Thalmann N (1995) Navigation for Digital Actors based on Synthetic Vision, Memory and Learning”, Computers and Graphics, Pergamon Press, Vol.19, No1, pp.7-19.
7. Noser H, Thalmann D (1995) Synthetic Vision and Audition for Digital Actors", Proc. Eurographics ‘95, Maastricht.
8. Pandzic IS, Kalra P, Magnenat-Thalmann N, Thalmann D (1994) Real Time Facial Interaction, Displays, Vol. 15, No. 3, pp. 157-163.
9. Renault O, Magnenat Thalmann N, Thalmann D (1991) A Vision-based Approach to Behavioural Animation,” The Journal of Visualization and Computer Animation, Vol 1, No 1, pp 18-21.
10. Reynolds C (1987) Flocks, Herds, and Schools: A Distributed Behavioral Model, Proc.SIGGRAPH '87, Computer Graphics, Vol.21, No4, pp.25-34
11. Reynolds CW (1993) An Evolved, Vision-Based Behavioral Model of Coordinated Group Motion, in: Meyer JA et al. (eds) From Animals to Animats, Proc. 2nd International Conf. on Simulation of Adaptive Behavior, MIT Press.
12. Ridsdale G (1990) Connectionist Modelling of Skill Dynamics”, Journal of Visualization and Computer Animation, Vol.1, No2, pp.66-72.
13. Sims K (1994) Evolving Virtual Creatures”, Proc. SIGGRAPH ‘94, pp. 15-22.
14. Slater M, Usoh M (1994) Body Centred Interaction in Immersive Virtual Environments”, in: Magnenat Thalmann N, Thalmann D (eds) Artificial Life and Virtual Reality, John Wiley.
15. Tu X, Terzopoulos D (1994) Artificial Fishes: Physics, Locomotion, Perception, Behavior”, Proc. SIGGRAPH '94, Computer Graphics, pp.42-48.
16. Wilhelms J (1990) A "Notion" for Interactive Behavioral Animation Control", IEEE Computer Graphics and Applications , Vol. 10, No 3, pp.14-22