Easy Mocap

Motion capture with one camera

AGI09

Irena Andonov, Arnaud Ramey

{ ireand18 , ramey}@kth.se

1. Ambition level

With our work we have aimed to:

- build a simple application in deepening our own chosen area of computer graphics, such as motion capturing and applying on maya or OpenGL, which includes programming of computer hardware for graphics, etc.
- review some existing methods for AGI
- participate in the development of new methods / applications for AGI as in make a real-time and high-definition avatar move thanks to the captured motion by a human standing in front of a camera.

This corresponds to course objectives 6, 7 and 8.
So, **our ambition level is VG**.

2. Algorithm description

The method is to select the variable points of an actor with different (strong) colors, and let him/her do some movements that are recorded by a camera. The camera tracks down the persons points and movements by the colors. The colors have to differ strongly from each other so this can work. After the recording, we will implement
the recorded movements on visual figures that has the similar structure as the human body.

For example we have thought of using a Gingerbread man (like Gingy in Shrek 2), and maybe the bunny I (Irena) made in the first lab..and more figures like that that move in different ways. Another idea is to have two avatars boxing, one controlled via the motion capture, and the other by an Artificial Intelligence fitting to the skills of the player. This part is under consideration.

3. Real implementation

The program will be structured in two big parts :

3.1 Motion capture

Finding the markers located on the body of the actor will be made with OpenCV [openCV].
The markers are colorful spots, binded to the body of the actor.

The research of these markers will be based on a color analysis (hue filtering).

To find quickly the connected components, we use an algorithm based on the Union Find algorithm [Galler].
Then, to match the markers in pair from one frame to the next one, we will use a distance minimization alogrithm.

When we don't have the coordinates of a marker on an image, it will be interpolated thanks to the other locations of this marker we have on the previous and next images.

3.2 Utilisation of the data

Once we have the 3D locations of the markers in the whole video, we can use it in two ways :

1. **OpenGL animation :** we compute the motion from the frames acquired by the camera.
   Then, we directly send the description of this movement to a OpenGL program.
   This program would for instance make the animation of an avatar imported from Maya, as in the lab #2 from AGI09.
   This application is real-time.

2. **Maya animation :** we export the movement in a given format, in a txt file.
   Then, we import this "movement descriptor" into Maya, and apply it to the skeleton of an avatar to make it move.
   This application is **not** real-time.
   We don't know actually if such import of information is possible with Maya.

4. References

Under construction...

- [Galler] Bernard A. Galler and Michael J. Fischer, ”*An improved equivalence algorithm*”, Communications of the ACM, Volume 7, Issue 5 (May 1964), p301-303 ; 1964.
- [openCV] Gary Bradsky & Adrian Kaehler, ”Learning OpenCV” ; 2008.
  Website : http://opencv.willowgarage.com/wiki/

Fig : Gingy, an avatar from the animation movie Shrek 2.
One of our aims is to be able to animate such a model
in real time with our motion capture system.