GAMIC: Exaggerated real time character animation control method for full-body gesture interaction systems

Hajime MISUMI, Wataru FUJIMURA, Takayuki KOSAKA
Motofumi HATTORI, Akihiko SHIRAI
Kanagawa Institute of Technology
E-mail : gamic@shirai.la Web : http://www.shirai.la/project/cartoonect/gamic

1 Motivation: intuitive animation control method and tool for KINECT generation

Real time animation in-game avatars driven by full-body gesture interaction systems using real time motion capture data are extremely appealing. Further, it is becoming more popular since researchers and developers can easily access consumer priced depth sensors in Microsoft KINECT and OpenNI frameworks. However, a system that can realize suitable real-time animations for each player actions is required. This article describes a new method and a tool for exaggerated but sophisticated interactive animation control systems for real-time animation playback timings.Our method can improve best player experiences without programming by offline/online GUI's for current video games and interactive systems.

Figure 1 : *A scene of motion design in a game project*

2 Related works

There are some challenges using depth based real time motion capture data for animating avatars in real time game plays. Omek Interactive reported a solution for these issues by predefined animation databases with partial skeleton blending using a depth sensor system[Bleiweiss et al. 2010].

They had reported two issues. The first is due to the limitations of the current tracking techniques in producing smooth, noise free, accurate animation in real time. The second, and more acute problem, stems from the fact that in most games, the movements of the animated avatar are expected to be more expressive than the player's actual movements. In such cases, one would like to visually enhance the player's motion to display super natural motions or even exaggerated special actions like super jump, super punch or finishing blow.

We focused to the second issue and added actual user interfaces to improve exaggerated real time character animations.

3 Concept

We found three key issues through actual game development project in this subject.

The first issue is workload of motion interaction designer (MID) who works for creating intuitive gesture in game plays. MID does several hundred times of trials with programmers and actors to explore each trigger motions to assign the avatar motions. It needs a generic method also for difficulty adjustment and level design process.

The second issue is the algorithms. It should be simplified for realtime experiences. We found most gestures in actual game plays can be explained as a sequence of three postures of kinematics that can call as Start Frame (SF), Trigger Frame (TF) and End Frame (EF).

The third issue is a timing of animation playback. TF is a trigger motion that starts posterior half of avatar animations. If the agent plays the posterior half before the trigger detection, it gives an uncomfortable feeling to the player. It is also uncomfortable if the playback timing delayed. Especially, special actions should be executed with a correct timings by an intuitive gesture input for all players robustly for better game play.

From concept above, we developed a tool "GAMIC" which optimized an animation reproduction timing in the simple algorithm using three frames for less workload of MID.

GAMIC is an abbreviation of Game Action Motion Interaction Controller, and it is a tool which can link the timings between KINECT recognition and avatar animation playback by GUI, WiiRemote and actual motions. It needs linking between physical player action and real time avatar animation playback timings in the development of game system that assumes a full-body gesture interaction using KINECT.

4 Method

GAMIC defines a recognition timing of KINECT for animation playback timing by GUI. MID stores two target gestures of the start and the end (*T1*, *T2*) using WiiRemote in front of KINECT on GAMIC GUI.

$$f_{\mathrm{Target}} = \sum_{i=0}^{k} \left(\frac{T_i \cdot V_i}{\| T_i \| \| V_i \|} \right) \quad (k=15)$$

$$V = \text{Current Kinematics}$$

$$f_{(T,V)} = \sum_{i=0}^{k} \left(\frac{T_i \cdot V_i}{\| T_i \| \| V_i \|} \right) \quad (k=15)$$

$$= \begin{cases} 1: \text{perfect match} \\ -1: \text{completely different} \end{cases} \quad (-1 \sim +1)$$

```
if( f(S,V) > P1 )
    if( f(E,V) > P2 )
        StartAnimation();
```

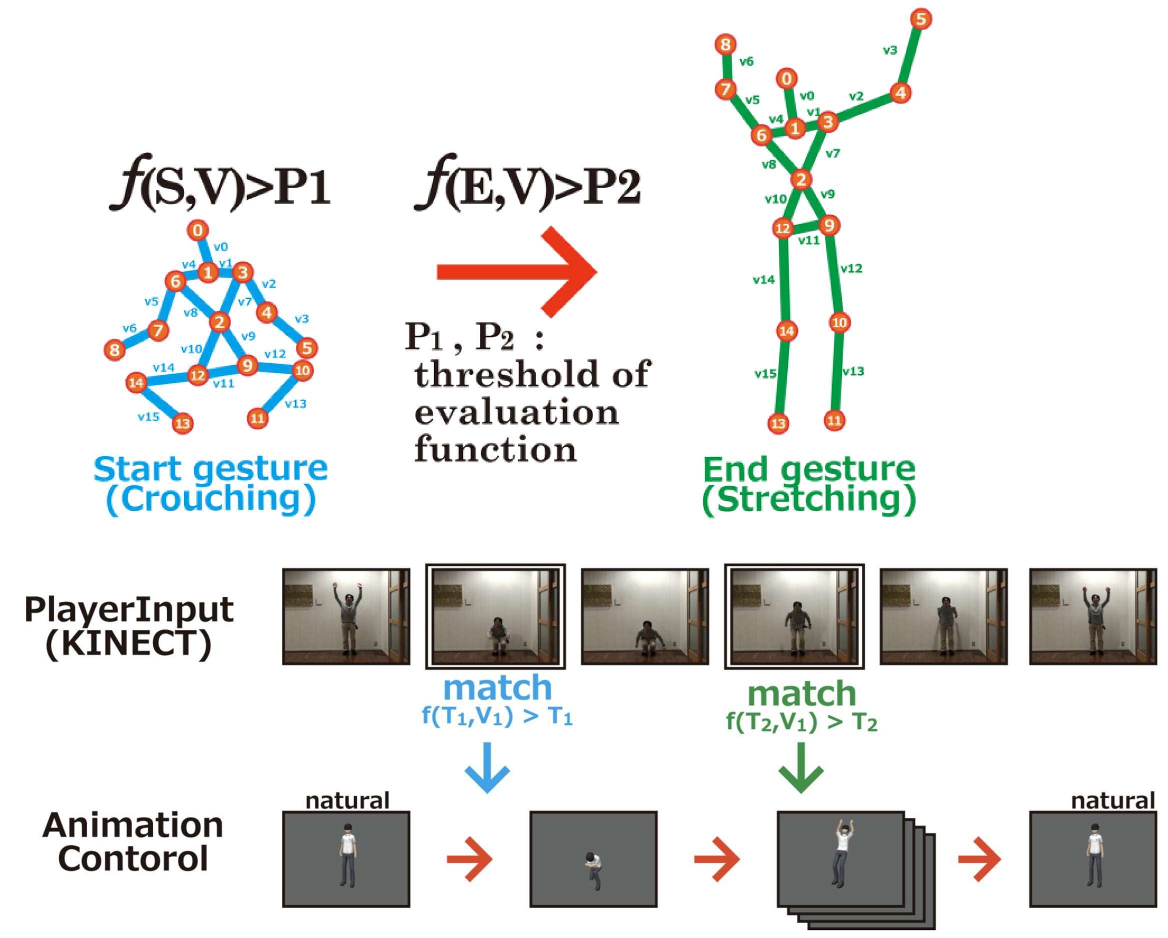

Figure 2 : *Evaluation function and algorithm of GAMIC*

This recognition can be expressed as an evaluation function of current posture with its threshold for a target frame. This evaluation function is a similarity between current player's kinematics and target posture from KINECT input, and it can obtain as a summation of inner products of target and current bones.

If a current posture *V* fits to a target posture T, the evaluation function *f(T; V)* outputs 1. Its threshold *P1* can control recognition difficulty by *f(S; V)* as a starting target frame *S*.

Trigger Frame (TF) is an intermediate posture and it is exist in between SF(*T1*) and EF(*T2*) cause of continuity of human motion. TF is an indescribable and dynamic posture but it can be expressed by evaluation function *f(E; V)* with its threshold *P2*.

TF must be set correctly. It can generate togetherness if the posterior half animation played synchronously with the recognition.

GAMIC can control recognition sensitivity, timing and animation impressions in same time by adjustment of *P2*.

5 GAMIC: Game Action Motion Interaction Controller

Figure 3 is a screenshot of GAMIC prototype.

The upper side shows KINECT inputs and the lower side shows a predefined FBX animation which is assigning to a KINECT action. MID can save his Start Frame (T1;left-top) and End Frame (T2;right-top) by his button actions and his actual motions from KINECT depth images.

P1 can be described as a diameter of red circle on the left. It means a similarity and threshold of the evaluation function to start an animation from player's unknown kinematics.

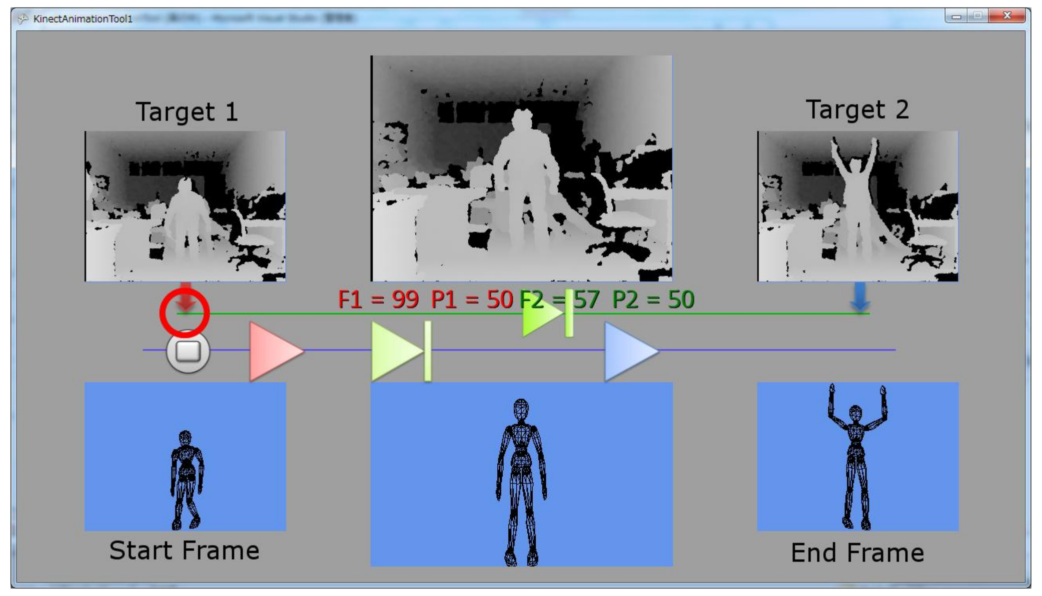

Figure 3 : *Screenshot of GAMIC prototype*

When red circle is large (=*P1* is small), a player gesture will be detected easier if the kinematics is similar with the T1.

Red and blue triangles are markers of animation playback start-end frames. *P2* is a timing to play the posterior half of the animation.

These markers are controllable by WiiRemote buttons.

When a player gesture is recognized as T1, the animation controller plays an animation until triger condition then pause it automatically.

Then the player makes a gesture that is determined as similar with T2 by threshold *P2*, the rest sequence is played.

MID can explore a suitable animation timings by adjusting markers by WiiRemote buttons and KINECT motion inputs.

6 Conclusion

GAMIC realized a higher quality of animation timing implementation with resource effective tool by non programming method.

This process had need a few staff (MIA, Programmer, Actors) to create core of interaction sense in a past project.

This method is generic and open. And it will be applied to develop a new animation software for robot motion designs, interactive attractions and transport past game titles to KINECT generation.

It will be also integrated with physics and/or macine learning based posture stimation and animation blending to create effective interaction experiments in near future.

Template matching, physics and learning based approaches [Reil and Massey ; Natural Motion] have chances to improve the player gesture recognition and dynamic animation but these techniques also needs human decision, huge times of trials and errors to improve.

GAMIC has an advantage of exploring intuitive player actions for real time animations by MID instead of machine learning methods.

This advantage focused to improve assignment and linkage between player actions and predefined animations especially for special actions which is impossible by physical motions. It was also good point to find a value to assign easier gestures to improve game play for handicapped or elder players (Figure 4).

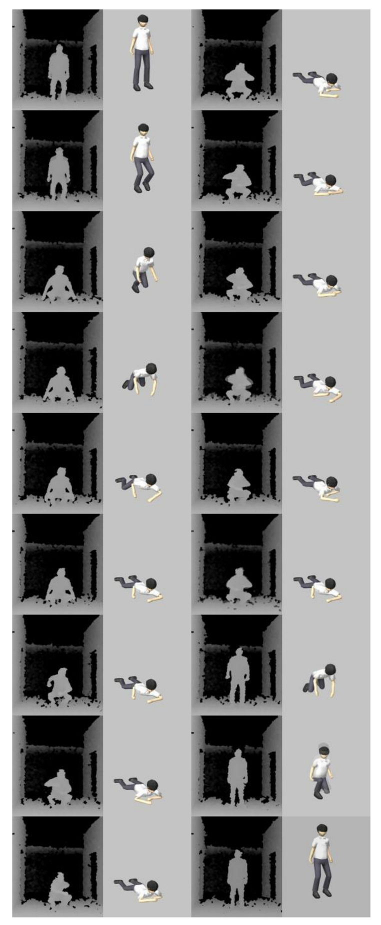

Figure 4 : *A crawling movement, this player action can be assigned to an easier motion for elder and/or handicapped players*