INFORMATION EXTRACTION AND SYNCHRONIZATION CONTROL METHOD FOR MULTI-HUMAN ANIMATION

Kihyun Kim and Sangwook Kim

Computer Languages and Multimedia Laboratory
Department of Computer Science, Kyungpook National University
Sankyukdong 1370 Pukgu, Taegu 702-701, Korea
Email: {kimkh, swkim}@woorisol.knu.ac.kr

ABSTRACT

When movement of human is animated, motion information of each body part needs to be synchronized adequately in accordance with the motion type. In this paper, we propose an advanced method with the goal of technology supporting a more natural and intuitive interaction. Motion data is extracted to synchronize each movement of human. Synchronization method can provide an important cue in controlling motion. Dynamic motion editor is a tool that extracts motion data. It can create motion data of human and define a various motion by displaying the process of motion as real-time in virtual environment. Also, such algorithms can apply a collected motion data to a different movement of each human and adequately control the motion information of each human. Central to our approach is the use of Human Motion Control Model which is based on analytically synchronization method for various types of human movements. It offers an easier motion animation and control.

Keywords: motion animation, synchronization, motion data, step control algorithm, knowledge-base

1. INTRODUCTION

Within the field of computer animation the automatic interpretation method of various human movements is one of the most challenging tasks [Brude97]. A central problem in analyzing such movements is due to the fact that the human body consists of body parts linked to each other at joints which allow different movements of the parts. So, it is necessary to understand an intention of user and information to method that moves each body part. To pursuit the facility of animation that applies to a motion characteristic of human is currently a main problem in animation field that was interested in developing a high-level motion control and an efficient method. Method that applies to realistic data using motion capture for movement information of human is suited to real-time motion capture system[Carls97][Hodgi].

According to the type of motion, it is difficult to process an error and modify problems in accompany with capture time[Rose96]. whenever a step, walk velocity, and orientation changes, motion information has problems that should generate new data. Similar motions must capture on and on. A sequenced motion and motion of each body part demands a synchronized processing control method as to moving path of human character in three dimensional space by mixing a generated motion data[Balag95]. We suggest an interface environment that can extract motion data as method to synchronize each body part of human. In this paper, We describe a control method of motion as to moving path and a realistic control to generate motion of human using motion information. chapter 2 describes an interface environment and internal processing that can extract motion data as a method to synchronize each body part. chapter 3 describes a control method that applies an extracted information to generate movement of human. chapter 4 describes a human motion control model that applies control methods to synchronize. Finally, we explain an implementation and conclusion.

2. MOTION INFORMATION EXTRACTION

2.1 Synchronization

To animate movements of human, it is necessary to control a movement of each body part adequately and mediate various relation of whole motion. Also, it must process motion information of each human properly. Synchronization is to apply control methods that can combine motion information according to type of motion. It is an efficient method that schedules movements of each body part. Especially, it can control a moving path of space as to visual script input of user. Here, visual script is a line which depicts the movement path of human character and type of motion in plane.

2.2 Dynamic Motion Editor

A user should be able to interact with objects in three dimensional space. In animation research part, it is a very important problem to support such an element of interaction. Dynamic Motion Editor can define a various motion representation that user wants[Kim98].

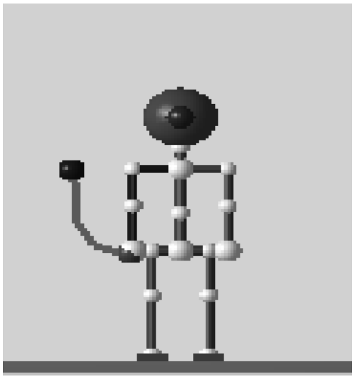
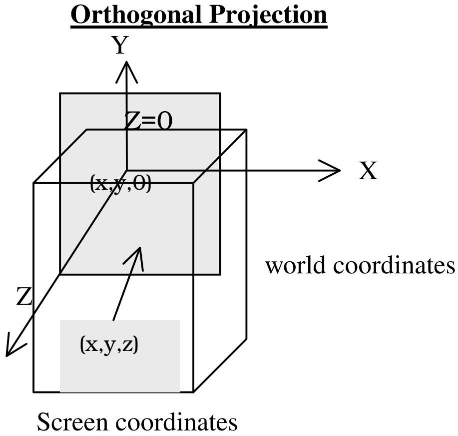


Figure 1: Example of authoring motion

Also, it can be used to synchronize a motion. Dynamic motion editor increases an interaction of user and generates real-time motion data. Fig.1. represents an example that describes an interface screen of motion editor and authoring motion. User selects each body part of human in a dialogue that displays human tree structure in interface. user moves a selected articulation by dragging mouse. Such a series of processing automatically store a path that user moved in a knowledge-Base. various motions can be edited and authored. High level synchronization method controls a property of each body part and interaction of articulation. It can create remarkable motion that sufficiently reflected user's intention by controlling a moving coordination of articulation and a calculation as to time. To generate a soft motion, motion is interpolated by using B-spline.

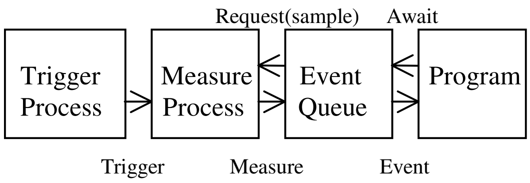
2.3 Extraction and Process of Motion Data

Motion capture system is a most efficient method that extracts motion data. But, it's very cost when a various motion data is demanded. Here, we proposed that motion data is extracted using mouse, there is a constraint condition to generate motion data using mouse that can input only two dimensional value. It is very useful to obtain three dimensional data using other input device such as spaceball or trackball. But, to develop a method that obtains three dimensional data using currently two dimensional mouse resolves a problem of cost that other input device has. Also, It can friendly support an environment for user.



$$\begin{bmatrix} X_p \\ Y_p \\ Z_p \\ 1 \end{bmatrix} = \begin{bmatrix} 1\,0\,0\,0 \\ 0\,1\,0\,0 \\ 0\,0\,1\,0 \\ 0\,0\,0\,1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

(a)



(b)

Figure 2: Extraction Function and Process

Fig. 2. represents functions used to extract motion data. Synchronization controller maps a value of screen coordination that is inputted by a user to world coordinate and extracts three dimensional data. Mapping process automatically performs as a part of viewing process. User must define a little constraint parameter to define such a mapping. That is, user has to fix a certain axis of x, y, z coordinate. After moved into a demanded space, synchronization controller extracts a required data. (a) represents a orthogonal matrix and process that maps input value to three dimensional coordinate. That is, a point is projected on two dimensional plane by fixing a

certain plane in x, y, z axis. In (b), Trigger processing represents that a user specifies a point coordination by dragging mouse. Measurement processing demands an extracted coordination value in program and stores a extracted value in event queue. Again, it formats a series of motion cycle that creates movement of each human. user can edit a demanding motion in interface.

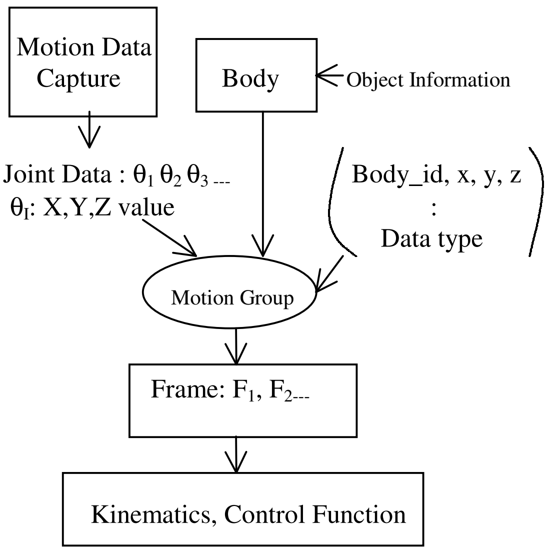


Figure 3: Motion Data Extraction Structure

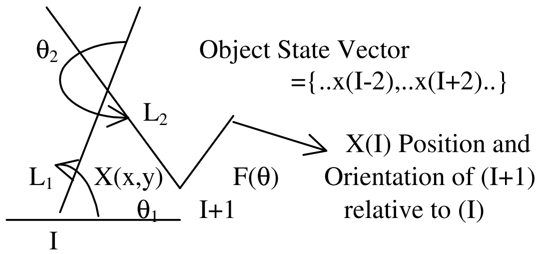
Fig. 3. presents an internal process and structure of motion data. Motion trajectory analyzer searches motion trajectory when using dynamic motion editor to approach a defined path of user. Final trajectory is generated in accordance with a current position and moving trajectory while composes variation of speed and motion properly. Motion Trajectory Analyzer correctly calculates a basic distance and controls as to a trajectory that user demands by specifying variation of motion type or orientation. The key property is defined by adding to synchronization event condition.

It is a possible specification that controls synchronization of a various basic motion. We use B-spline method to generate motion softly. In a 3D-space, Motion Trajectory Analyzer generates as motion frame by interpolating an interval of movement defined by user. Knowledge-Base includes an authored motion data sets. Motion data sets have an articulation angle and a moving coordination on three dimensional space per each frame. Synchronization controller controls all information.

3. CONTROL METHOD FOR SYMCHRONIZATION

3.1 Articulation Control Function

Fig. 4. represents function to control articulation of body part. This searches position of articulation about motion using kinematics method. Function automatically performs calculation in accordance with type of motion. Articulation trajectory describes articulation range of time[Gascu95].



Next Joint Point: $X=(I_1\cos\theta_1 + I_1\cos(\theta_1 + \theta_2))$
$Y=(I_1\sin\theta_1 + I_1\sin(\theta_1 + \theta_2))$
$X(x_i, y_i)$

Figure 4: Function to control articulation

It is derived by specifying a key position and orientation in relation to other body part that use an authored motion data. Each body part moves only one plane and is fixed at other end. Each body part is composed in a function relation. Theta(θ) is a limited angle for moving each body part. Theta(θ) is L(joint Length)/D(movie distance): L is a length of body part that is connected by articulation. D is a distance that is moved by articulation of body parts. That is, each body part of human moves in a range of limited angle. If range of angle is not limited, motion generates an unwanted motion. Angle that Each body part moved :

$$\mathrm{Object}_I(\theta_{total}) = \sum_{\mathrm{Initial}}^{\mathrm{final}} \Delta A(t_I - t_{I-1})$$

T represents time variation of frames and is effected by interpolation of B-spline function. ΔA means variation of each angle value of B-spline. Moving coordination value uses a search processing that is based on kinematics. Searching function for next articulation point X(x, y) coordination is a derived a function that is based in formula of x=rsinθ, y=rcos θ (trigonometric function). Length and angle of each body part is calculated by using motion frame data that stored in knowledge-Base.

3.2 Motion Step Control Algorithm

```
 if(m_bPlay == TRUE){
LPANIMATION pHAnim = this-
>GetCurrentAnimation();  -------------(a)
            m_iPlayStepCount++;  ---------(b)
     if(m_iPlayStepCount >= m_iPlaySpeed){
        m_iPlayStepCount=0;
        pHAnim->m_iCurrentFrame++;---(c)
if(pHAnim->m_iCurrentFrame>pHAnim->
         m_iTotalFrame){   ------(d)
        pHAnim->m_iCurrentFrame = pHAnim->
```

```
                m_iTotalFrame;
        pHAnim->m_iCurrentFrame=1;
                }
        this->UpdateFrameInfo();
    }
```

Figure 5: Motion Step Control Algorithm

Fig. 5. represents an algorithm process to control movement of step wide as to motion step. According to timing, control of velocity is an important part as well as static position data in motion execution. When type of motion have a variety, it is an important that applies to a most suitable velocity in characteristic viewpoint of each motion. For a consistent velocity, foot is located in center of distance that anticipated to move while foot is positioned on floor. Foot must locate in near to hip for increasing velocity.
Step Manager controls motion step in proportion to moving distance that arrived at finally. Motion step controls by using a total frame information as to motion type. Especially, Motion step control algorithm uses a frame number of a demanded motion animation and motion step. It displays current motion type in real-time as to moving distance. It continuously executes motion. (a) express current motion type of human. (b) counts the number of motion step (c) counts current frame number of current motion. (d) compare current frame number with total frame number.

3.3 Scheduling Motion

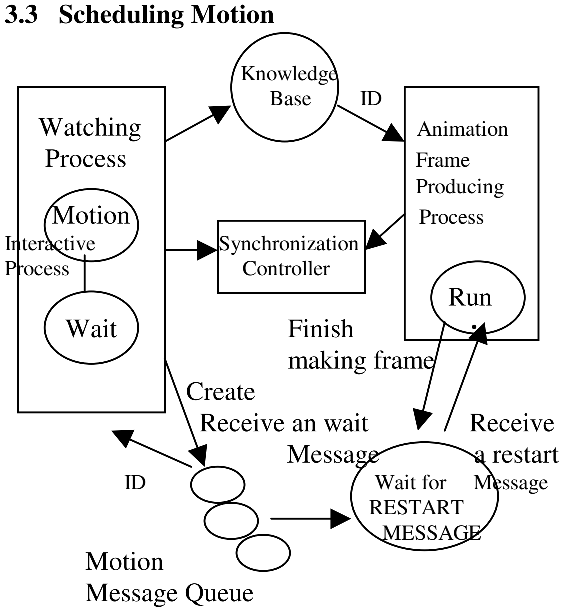


Figure 6: To schedule a processing of Motion

Scheduling motion is to design for analyzing the movement types of human (walking, running, jumping, etc). A movement information of each human need to processing that schedules motion to execute each motion.

While a synchronization controller implements a demanded motion as to step wide, walking velocity, orientation, motion is generated by applying to an efficient motion search algorithm using concept of knowledge-Base. Motion information through knowledge-Base generates motion of each human body part. Real-time motion variation is done through processing of comparison analysis process and scheduling of motion message queue. Fig. 6. represents an internal processing of motion scheduling.

3.4 Knowledge-Base

Knowledge-Base is a method that uses to construct an movement information of articulation and to design complex motions.

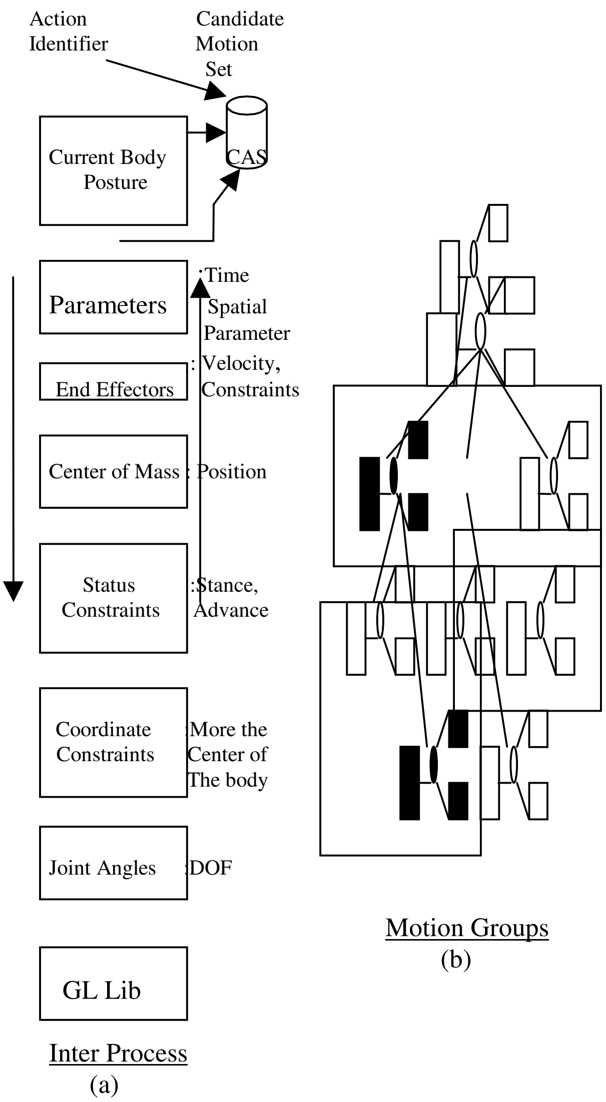


Figure 7: Knowledge-Base and Internal Structure

It has hierarchical structure of each motion. To move each articulation of body part, it is interpolated by B-spline in proportion to time.

Fig. 7. presents a process that manages knowledge-Base and forms motion groups. Real position of a motion group obtains by composing variation of root motion group. (a) represents a process that format each parameter and constant. (b) represents a process that composes internal structure each motion in knowledge-base.

3.5 Procedure to synchronize a movement of human

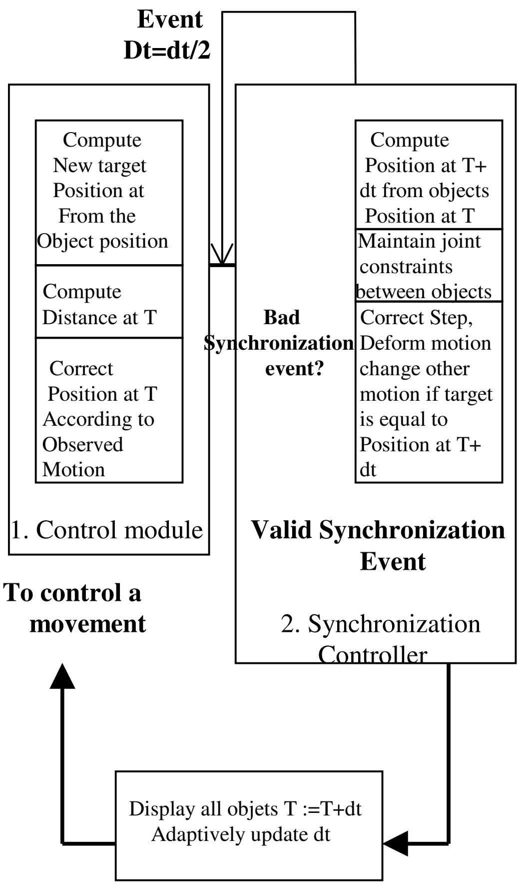


Figure 8: Synchronization Process

Synchronization must control properly according to the type of motion and the number of human. It is applied to movement of each human through a processing of human motion network and a series of search. We describe processing of synchronization method about two human characters. Constraints for synchronization are influenced by a current position of each human character, velocity and distance as to time. Motion forms a series of cycle. It is an important to be applied to movement of each human motion in real time as to time constraint factor. In a processing of Fig. 8., control module automatically calculates each motion frame about movement. If event condition to synchronize is satisfied in human motion network, Synchronization controller reforms velocity and position. Finally, if a demanded motion and position is reached, it is exchanged to display next motion about an input of visual moving path script. Control module processes a variation of distance and position.

P^0= Initial Start Point,
f^*=each variation Position (from n-1)
***if* Object Selected by User, start SCA**
store *Position Values*
***if Variation Point* is smaller than**
***Final Point n* - ---(a)**
repeat
***if* Point i equal to Final Point n then Final Stop;//No more Moving Pointer ---(b)**
***If* $|f^{*-}$ *Search*$(f^{*,}\ P^{i+1})|$ is smaller than n_k or equals to n_k *then* -- - (c)**
<u>*PathMotion*</u>(Actions^i, P^i) according to P^{i+1} --- (d)
{
repeat
calculate <u>*SpaceDistance*</u> between x_1 and x_j
execute <u>*Actions*</u>(Walk, Run,….){
according to P_j^i, f^i{
execute <u>*MotionFunction*</u>(Arm, Leg, Torso,….){
***call* <u>*GLLibrary*</u>(gltranslatef, glrotatef…);//DOF}**
}
***until* $||P^i||$}**
***if*$||P^{i+1}||$ is equal to $||P^i||$ then, execute <u>*NextMotion*</u>; //Next Moving Pointer Start ------ (e)**
***Next Position*(V^{i+1}) Searches (f^*, P^{i+1})**
***until* I=0, 1….**

Figure 9: Path Control Algorithm

Fig. 9. represents processing of path control and specifying path of three dimensional animation. In a space, user must concurrently specify position, orientation and time. In level of each motion animation, synchronization controller controls by (a) using difference between final aim of movement position and human character. (b) Motion step as to variation of time checks final position about aim of movement position. (c) searches movement distance parameter as to high-level motion type that can reach aim distance. (d) executes motion as to path of motion. (e) executes next motion.

Synchronize Multiple human character using visual scripting Information
repeat
Select A, B …;//human characters **----(a)**
Evaluate A, B.. Motions; **----(b)**
If(A, B…) ***Then***{go to Target Position.
A(B) synchronize with B(A)} **– (c)**
Repeat
Evaluate a posture ***post*** A, B and orientation A, B;
Control arm, legs, head actions to adopt **---- (d)**
the posture ***post*** A, B;
Evaluate all Characters

If(Max(des A, des B) current character=Max (des A, des B); **---- (e)**
until (Final Position is Reached)
If(reached Destination) ***Then*** Evaluate Priority(A, B); **-----(f)**
Until Last human character finishes Motion **--- (g)**

Figure 10: Synchronization Algorithm

Fig. 10. represents a process of synchronization control algorithm for movement of each human. (a) applies to two human character. (b) can differently apply as to property of human character to apply authored motion data to other human character. (c) To synchronize movement of two human characters in three dimensional space, it forms the intimate relations between time and moving distance. Especially, when two human characters input a visual script that demands each different motion, it has really an influence to motion frame that they execute. (d) In local coordinate, it is important to specify key sets that defines a demanded path to A, B human character of user according to each key position or orientation, distance between start point and end point. It calculates an average of movement velocity in proportion to time with each human character. motion animation is not effected to visual script input of each A, B human character. A various synchronization events that generate about time must be satisfied in constraint condition-based. (e) When each human character moves in a space, they synchronize in relation to other human character toward aim position as to demanded path. Because algorithms operate during the single motion animation as to time, Fig. 8. controls motion animation by a synchronization relation of internal interaction. Position sets as to a demanded moving path of other human character is controlled by generating event to aim position movement of other human character in generating synchronization event. (f) presents motion priority of A, B character. (g) Final aim of human character is decided when user inputs script to position of motion that wants to movement. Human Motion Control System Model will automatically modify a movement trajectory for animating

4. HUMAN MOTION CONTROL MODEL

To synchronize of human character, system model must manage all information. Human motion control model manages such an information. Fig.11. represents a structure of human motion control model.

(1) human motion control model generates reliable basic motion and supports interaction and real-time control.

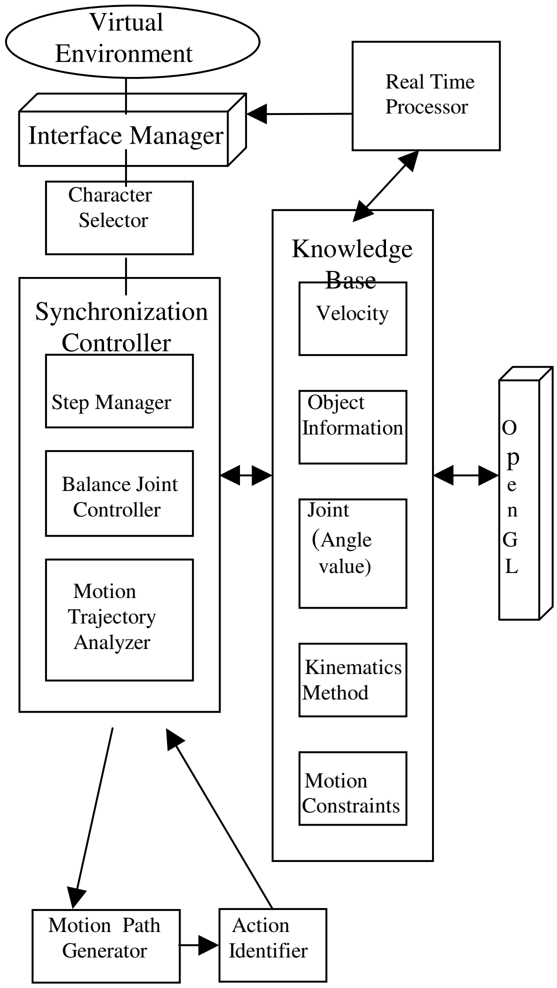


Figure 11: Human Motion Control Model

(2) motion transition is naturally formed to added to synchronization method as to variation of motion.
(3) It is able to add feedback method and constraint condition about various motion type. we use motion control function to animate motion
(4) State variation that is generated by synchronization event condition. Position point control of moving and movement of each human character is controlled by synchronization method processing by connecting frames of motion steps.
(5) It interpolates and controls motion frame using B-spline.

5. CONCLUSION

we implemented using visual c++5.0 and OpenGL library on windows NT. Fig 12 is an example that

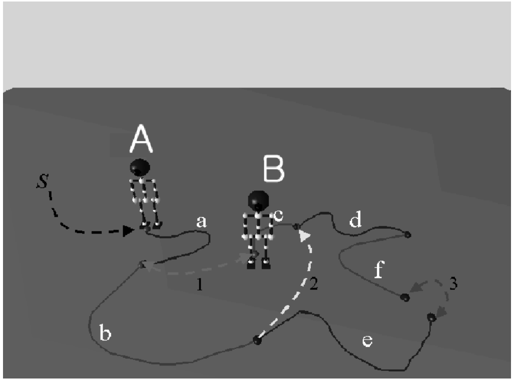


Figure 12: Implementation Example

represents processing that controls motion and movement path as to visual script.

In this paper, we suggested method that is demanded for motion information extract and motion control process to synchronize according to type of human motion. An extracted motion using dynamic motion editor can apply to human character animation. Also, when human character is composed of a various human, it controls motion and position of movement according to script input of each human. In future, method using synchronization relation of environment object needs to develop realistic control algorithm as to a various motion information. Also, it will be necessary to develop method to control motion type between an inanimate object and a living thing.

REFERENCES

[Brud97]Bruderlin,A.:*Knowledge-Driven Interactive Animation of Human Running,* The Fifth Pacific conference on Computer Graphics and applications, pp33-41, October, 1997

[Balag95] Balaguer ,J. F., Gobbetti ,E.: *Sketching 3DAnimation,*TheProceedings of EUROGRAPHICS, 1995

[Carls97] Carlson, D. A., Hodgins, J. K. : *Simulation Levels of Detail for Real-time Animation,* Graphics Interface'97, pp. 1-8

[Gascu95] Gascuel , J. D., Lyon , C.: *A New Set of Tools to Describe and Tune Trajectories,* Proceedings of the IEEE Computer Animation, pp82-89, April, 1995

[Hodgi] Hodgins, J. K., O'Brien, J. F.: *Computer Animation*

[Kim98]Kim,K.H.: *Motion Control Synchronization for human animation*, master theory, computer science ,1998. 12

[Rose96] Rose. C., Guenter. B. , Bodenheimer. B., Cohen. M. F.: *Efficient Generation of Motion Transitions using Spacetime Constraints*, Computer Graphics Proceedings, pp147-153, 1996