Gabe Aron

9/28/2015

ConOps for the Skeletal Animation Framework

**System Requirements**

The Skeletal Animation Framework is an iOS application written in Objective C. It shall render an OpenGL figure that resembles a human skeleton. It shall animate this figure using positional coordinates as well as a body-part index[[1]](#footnote-1) placed manually or automatically in a FIFO data structure. Each positional coordinate paired with a body-part index shall represent one element of the FIFO structure. The animated figure shall move each dictated body part smoothly between each positional coordinate over a prescribed amount of time, and in sequence of First In First Out (FIFO).

This framework shall also allow for a total of fifty-two FIFO structures to be set. The animated figure shall animate all indices of each FIFO structure with the same value concurrently. For example, if two FIFO structures exist – A and B – the animated figure shall move the first positional coordinates of both A and B at the same time, and then the second positional coordinates, and thus forth all the way to the final coordinates of A and B. This ability shall enable an indefinite number of body parts to animate at once, as well as over a prescribed amount of time.

The Skeletal Animation Framework shall be used by a development team that has access to wearable[[2]](#footnote-2) devices that send acceleration and gyroscopic data via Bluetooth to the main interface. This framework shall receive positional coordinates from these devices and allow for the positional developer[[3]](#footnote-3) to add this data to the FIFO structure.

**Justification**

The requirements of the proposed position system include:

* Accepting positional data from one to three wearable devices
* Transferring this data to an iOS devices (an iPhone or iPad)
* Translating this data into useful positional coordinates for each movement of the user
* Using this information to move an animated figure, accurately reflecting the movements of the user.
* Using this user movement to enable the user to receive feedback on the correct forms and sequence of movements of documented soccer kicks.

The Skeletal Animation Framework shall enable the animation of a figure with data put into a FIFO structure from any source. The source in the case of the requirements listed above shall be the positional data from the wearable devices.

**Use Case Scenarios**

*Case 1: Moving one body part once*

The user adds positional data and a desired body-part index to one of the 52 FIFO structures:

[movementQueues[i] addPosition: x: y: z: body-part index];

The user runs the program, and the intended body part moves once over the course of a prescribed amount of time.

*Case2: Moving one body part an indefinite number of times*

The user adds positional data and a desired body-part index to any number of one of the 52 FIFO structures:

[movementQueues[i] addPosition:x1: y1: z1: body-part index1];

[movementQueues[i] addPosition:x2: y2: z2: body-part index1];

[movementQueues[i] addPosition:x3: y3: z3: body-part index1];

. . .

[movementQueues[i] addPosition:xn: yn: zn: body-part index1];

The user runs the program, and the intended body part moves n times over the course of a prescribed amount of time.

*Case3: Moving several body parts in sequence*

The user adds positional data and unique, desired body-part index to any number of one of the 52 FIFO structures:

[movementQueues[i] addPosition:x1: y1: z1: body-part index1];

[movementQueues[i] addPosition:x2: y2: z2: body-part index2];

[movementQueues[i] addPosition:x3: y3: z3: body-part index3];

. . .

[movementQueues[i] addPosition:xn: yn: zn: body-part indexi];

The user runs the program, and the intended body parts move n times over the course of a prescribed amount of time.

*Case4: Moving several body parts concurrently*

The user adds positional data and a desired body-part index to any number of any number of the 52 FIFO structures:

[movementQueues[1] addPosition:x1: y1: z1: body-part index1];

[movementQueues[2] addPosition:x2: y2: z2: body-part index2];

[movementQueues[3] addPosition:x3: y3: z3: body-part index3];

. . .

[movementQueues[52] addPosition:xn: yn: zn: body-part index52];

The user runs the program, and all 52 body parts move at once n times.

**Evaluation**

The Skeletal Animation Framework is able to move each chosen body part of an animated OpenGL figure sequentially or concurrently over a prescribed period of time. The number sequential movements is indefinite, and the number of concurrent movements is restricted to 52 unique body parts.

It is not currently able to automatically insert positional data[[4]](#footnote-4). It is also not currently able to automate the prescribed time of each movement based on the physical movement of the user.

1. There are 54 body part indices (please refer to <ReadMe.rtf> for further information). Each index is mapped to one body part. When placed in the FIFO structure with the positional data, it shall enable dictation of which body part shall move for the given FIFO element. [↑](#footnote-ref-1)
2. Able to attach to ones clothes [↑](#footnote-ref-2)
3. A development team must implement a program that translates accelerometer and gyroscopic data from the wearable devices to appropriate positional data. [↑](#footnote-ref-3)
4. The algorithm for inserting the data must be provided by the developer of the position framework [↑](#footnote-ref-4)