Smoothing Noisy Skeleton Data in Real Time

Tom Hoxey & Dr. Ian Stephenson
NCCA, Bournemouth

Abstract

The aim of this project is to be able to visualise live skeleton tracking data in a virtual analogue of a real world environment, to be viewed in VR. Using a single RGBD camera motion tracking method is a cost effective way to get real time 3D skeleton tracking data. Not only this but people being tracked don't need any special markers. This makes it much more practical for use in a non studio or lab environment. However the skeleton it provides is not as accurate as a traditional multiple camera system. With a single fixed view point the body can easily occlude itself, for example by standing side on to the camera. Secondly without marked tracking points there can be inconsistencies with where the joints are identified, leading to inconsistent body proportions. In this paper we outline a method for improving the quality of motion capture data in real time, providing an off the shelf framework for importing the data into a virtual scene. Our method uses a two stage approach to smooth smaller inconsistencies and try to estimate the position of improperly proportioned or occluded joints.

CCS Concepts

•***Computing methodologies*** → ***Motion processing;*** *Motion capture;*

1. Introduction

There currently exists many methods to improve the skeleton tracking ability of RGBD cameras. However most focus on a computer vision approach working with the captured footage. Our method focuses on smoothing the actual skeleton data itself once it has been acquired. It should be noted that a computer vision approach to increase the accuracy of the skeleton acquisition is also an important part of the process, but our method is designed to be applied after this stage has occurred. Smoothing the acquired skeleton is a lesser developed area but we believe it still has value in both real time and offline applications. In real time the skeleton is a much smaller data set than the whole captured image, enabling complex filtering operations to be performed with a smaller overall cost. This is especially important in applications that transfer the data across a network. We propose a two stage method, consisting of a correction and filtering stage.

2. Existing Methods

The Kinect has gained a huge amount of popularity in motion tracking research. However it has been identified in papers such as [OKO*12] that the existing Kinect estimation is lacking in non-perfect scenarios, i.e. not standing front on to the camera.

There is existing research into using the Kinect in conjunction with a Kalman filter to smooth tracking data such as in [SHA14]. The emphasis in existing work is surrounding removing jitter, however we found another useful characteristic of the filter to be the interpolation across multiple frames.

Other optical tracking methods that estimate skeletal parameters such as [KOF05] often use some sort of body tracking markers. To keep our solution usable in a non studio environment markers prove to be far too impractical.

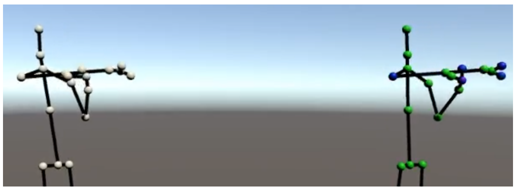

Figure 1: *Tracking data of a Shoulder Posterior Stretch. (Left Skeleton is raw data, Right is the corrected skeleton, joints in blue have been Average Fitted)*

3. Correction

The first correction is a simple adjustment to tracked joint positions. Over a series of frames an average distance to the parent joint is recorded. At each frame we check if the new position is going to be within five percent the current average: if so we set the joint to be at the current average distance away along its tracked direction. This helps to correct the error of proportions shifting during runtime. See Figure 1.

If we have no tracked data for a particular joint we have to try and approximate its position. To do this we use the averages that have been collected and the last recorded vector from the joint to either its parent or child joint (as approriate for each specific joint).

DOI: 10.2312/egp.20181014

Algorithm 1 Pseudocode for Estimating the position of Untracked Joints

procedure IS LIMB($CurrentJoint$)
  if $CurrentJointParent \rightarrow ChildCount > 0$ **then**
    $CurrentJoint \Rightarrow Limb$
  else
    $CurrentJoint \Rightarrow Fixed$
procedure ESTIMATE POSITION(Skeleton)
  for all $Joints$ in $Skeleton$ **do**
    if IS LIMB($CurrentJoint$) **then**
      $Current\vec{JointDirection} \Rightarrow Parent\vec{Direction}$
    else
      $Current\vec{JointDirection} \Rightarrow Last\vec{Direction}$

    $CurrJointPos \Rightarrow AverageDistance * CurrJoint\vec{Direction}$

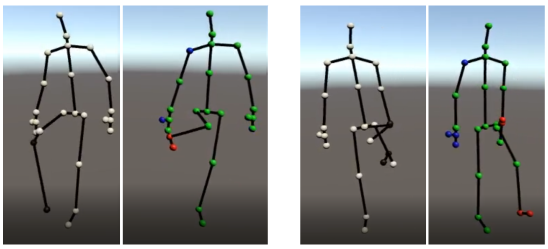

Figure 2: *Images showing capture data of a person performing a quadricep stretch . In each pair: Left Skeleton is raw data white joints are tracked, black are untracked, Right is the corrected skeleton, joints in blue have been Average Fitted and red have been estimated*

This is summarised in algorithm 1. The results of this can be seen in Figure 2, and show considerable improvement over the Kinect's raw tracking.

4. Filtering

To further smooth the data we use a Kalman filter on three dimensional coordinates over time. They are applied to a wide range of graphics problems, including motion capture problems to track image features across frames. [BW*01] This has two advantages to smoothing at the image processing stage. Firstly it reduces the amount of data that is processed. To stream even a compressed 1080p RGBD capture data requires a considerably higher data rate than the 24kB/s required to transmit the coordinates. Secondly the Kalman filtered points give smoother animation arcs than the raw data, which is much more visually appealing, as illustrated in Figure 3.

5. Conclusions

While our system provides a measured improvement over the raw tracking data, it does have limitations. This is especially notable regarding the estimation of joint position. The estimations that it provides typically return the joint to a rest position as seen in the far right image of Figure 2, This reduces the occurence of extreme deformations to the skeleton, at the cost of accuracy.

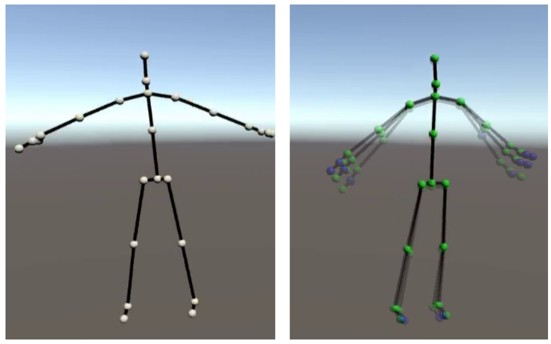

Figure 3: *3 frames of overlapped footage, showing that the smoothed implementation updates 3 times while the raw data only updates once*

There is research into using multiple Kinect cameras to create a more accurate overall picture, such as in [BRS*11], however for every new sensor we add the cost of setting up the system increases. This is in terms of both cost to buy the equipment, and time required to set-up the system. As such we decided to only use a single camera for a more plug and play approach.

A continuing goal of the system is to improve the rate at which we can correctly estimate the location of the joint. However since the Kinect does not provide rotation data for the joints it would most likely require the implementation of a full forward kinematics system. This was not implemented in the current version of the project as we were prioritising verisimilitude over strict accuracy.

References

[BRS*11] BERGER K., RUHL K., SCHROEDER Y., BRUEMMER C., SCHOLZ A., MAGNOR M. A.: Markerless motion capture using multiple color-depth sensors. In *VMV* (2011), pp. 317–324. 2

[BW*01] BISHOP G., WELCH G., ET AL.: An introduction to the kalman filter. *Proc of SIGGRAPH, Course 8*, 27599-23175 (2001), 41. 2

[KOF05] KIRK A. G., O'BRIEN J. F., FORSYTH D. A.: Skeletal parameter estimation from optical motion capture data. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) 2005* (June 2005), pp. 782–788. URL: http://graphics.cs.berkeley.edu/papers/Kirk-SPE-2005-06/. 1

[OKO*12] OBDRÅ¿ÃĄLEK Å., KURILLO G., OFLI F., BAJCSY R., SETO E., JIMISON H., PAVEL M.: Accuracy and robustness of kinect pose estimation in the context of coaching of elderly population. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (Aug 2012), pp. 1188–1193. doi:10.1109/EMBC.2012.6346149. 1

[SHA14] SHU J., HAMANO F., ANGUS J.: Application of extended kalman filter for improving the accuracy and smoothness of kinect skeleton-joint estimates. *Journal of Engineering Mathematics 88*, 1 (Oct 2014), 161–175. URL: https://doi.org/10.1007/s10665-014-9689-2, doi:10.1007/s10665-014-9689-2. 1