5.5 Hierarchical Skeleton Solution

5.5.1 Arbitrary Figure

An arbitrary figure can be drawn at each time frame if certain criteria are met for the data at that time frame. An arbitrary figure is made up of linked segments stored in a single parent-multiple children tree structure. The drawing of the figure consists of connecting the rotation points from parent to children. Each line drawn would then be fixed inside the segment. The process would continue from the root segment to the leaf segments. The leaf segments would have to just draw lines to the existing data points on that segment.

Starting with the original data, each assigned to their corresponding segment ($x_{s\,i}$), the point must be transformed to the coordinate system fixed to the parent of the segment s.

$$y_{si} = \mathbf{M}_{s-1}^{T}\left(x_{si} - p_{s-1}\right) \tag{234}$$

These new vectors ($y_{s\,i}$) are what are passed to the spherical center estimator formula. This formula remains valid and allows the process to continue to the children only if both p and $\mathbf{M}$ can be determined for the parent. This amounts to being able to construct three axes and a point that are fixed on a segment. There are three cases that are acceptable and are the criteria for being able to draw the entire skeleton at a particular time frame.

5.5.1.1 Case 1 – Three or more points

This is the easiest case to determine the segment's coordinate system. First, pick three of the available points, (x_{s1}, x_{s2}, x_{s3}). One of the points must be chosen as the center of the coordinate system

(235) $$p_s = x_{s1}$$

The three coordinate axes must then be constructed in a mutually perpendicular fashion (right-handed coordinate system). Two points can determine the first direction:

(236) $$\hat{x} = \frac{x_{s2} - x_{s1}}{\left| x_{s2} - x_{s1} \right|}$$

A second direction can be determined from the third point and the first direction:

(237) $$\hat{z} = \frac{x_{s3} \times \hat{x}}{\left| x_{s3} \times \hat{x} \right|}$$

The third direction can be determined from the previous two directions:

(238) $$\hat{y} = \hat{z} \times \hat{x}$$

The floating point operations involved in calculating one coordinate system is

(239) $$FLOP_3 = 39$$

Drawing a skeleton using solely this case of markers involves calculating the centers of rotation for all non-leaf segments in the hierarchical figure.

(240) $$FLOPs = N_S 39 + \left(N_S - N_L\right)18$$

where N_S is the number of segments and N_L is the number of leaf segments. A typical human figure with fourteen segments and five leaf segments produces

(241) $$FLOPs = 708$$

5.5.1.2 Case 2 – Two points

This relies on some previously calculated information to determine the coordinate system. Three points are needed and some point must be divined that is fixed on the segment. Luckily, the common rotation point between the segment and its parent has been previously calculated because of the tree traversal. The three available points are now (x_{s1}, x_{s2}, c_s). One of these points must be chosen as the center of the coordinate system

(242) $$p_s = x_{s1}$$

The center of rotation for this segment is determined from the parent's coordinate system with

(243) $$c_s = p_{s-1} + \mathbf{M}_{s-1}\tilde{c}_s$$

where $\tilde{c}_s$ is a constant vector in the parent's coordinate system pointing at the child's rotation point. The three coordinate axes must then be constructed in a mutually perpendicular fashion (right-handed coordinate system). Two points can determine the first direction:

(244) $$\hat{x} = \frac{x_{s2} - x_{s1}}{\left| x_{s2} - x_{s1} \right|}$$

A second direction can be determined from the third point and the first direction:

(245) $$\hat{z} = \frac{c_s \times \hat{x}}{\left| c_s \times \hat{x} \right|}$$

The third direction can be determined from the previous two directions:

(246) $$\hat{y} = \hat{z} \times \hat{x}$$

The floating point operations involved in calculating one coordinate system is

(247) $$FLOP_2 = 57 + FLOP_P$$

where $FLOP_P$ is the FLOPs needed to calculate the coordinate matrix M of the parent. Drawing a skeleton using solely this case of markers involves calculating the centers of rotation for all non-leaf segments in the hierarchical figure.

(248) $$FLOPs = N_S 39 + \left(N_S - N_L + N_2\right)18$$

where N_2 is the number of segments with two data points with a parent with three.

5.5.1.3 Case 3 – One Point

This is the easiest case to determine the segment's coordinate system. First, pick three of the available points, ($x_{s\,1}$, c_s). One of the points must be chosen as the center of the coordinate system

(249) $$p_s = x_{s1}$$

The three coordinate axes must then be constructed in a mutually perpendicular fashion (right-handed coordinate system). Two points can determine the first direction:

(250) $$\hat{x} = \frac{c_s - x_{s1}}{\left|c_s - x_{s1}\right|}$$

(251) $$c_s = p_{s-1} + \mathbf{M}_{s-1}\tilde{c}_s$$

A second direction can be determined from the null vector previously calculated:

(252) $$\hat{\tilde{z}} = v_s$$

(253) $$v_s = \mathbf{M}_{s-1}\tilde{v}_s$$

The third direction can be determined from the previous two directions:

(254) $$\hat{y} = \hat{\tilde{z}} \times \hat{x}$$

The rotation matrix $\mathbf{M}_s$ can now be constructed by placing the three coordinate axes as columns in the matrix.

(255) $$\mathbf{M}_s = \begin{pmatrix} \hat{x}_x & \hat{y}_x & \hat{\tilde{z}}_x \\ \hat{x}_y & \hat{y}_y & \hat{\tilde{z}}_y \\ \hat{x}_z & \hat{y}_z & \hat{\tilde{z}}_z \end{pmatrix}$$

These three cases will allow the reconstruction of an entire skeleton as long as the root segment has at least three markers (Case 1). The recursive nature of the other cases precludes them from the root.

The floating point operations involved in calculating one coordinate system is

(256) $$FLOP_1 = 54 + FLOP_P$$

where $FLOP_P$ is the FLOPs needed to calculate the coordinate matrix M of the parent. Drawing a skeleton using solely this case of markers involves calculating the centers of rotation for all non-leaf segments in the hierarchical figure.

(257) $$FLOPs = N_S 39 + \left(N_S - N_L + N_1\right)18$$

where N_1 is the number of segments with one data points with a parent with three.

5.5.2 Predefined Marker Association

During motion capture, markers are placed over the body and tracked by one of several methods available. The animator of the tracked data either has to have previous knowledge of the marker's associated segment or come up with an algorithm to do the association. For those with no a-priori knowledge, the algorithm can get time consuming. Some authors [60] use this method by classifying markers together that don't move relative to each other. The grouping methods work when there is more than one marker per segment and the hierarchy of segments is well defined in the data. These methods are very slow though.

In order to figure out which markers are associated with which segments, the names of the markers within the data were analyzed for normal naming conventions. The association was then hand written to a text file that is read after the data is read. Naming conventions are fairly straightforward for most sets of data. Examples of normal naming conventions and their segments can be found in Table 1.

Table 1 Marker Associations

Marker Name	Associated Segment	Location on Segment
RFHD	Head	anterior right top
LFHD	Head	anterior left top
RSHO	Chest	mid right top
RFTShould	Chest	anterior right top

RRRShould	Chest	posterior right top
LSHO	Chest	mid left top
LFTShould	Chest	anterior left top
LRRShould	Chest	posterior left top
CLAV	Chest	anterior mid top
RUPA	Upper Right Arm	mid right
LUPA	Upper Left Arm	mid left
RELB	Upper Right Arm	mid right bottom
LELB	Upper Left Arm	mid left bottom
RARM	Lower Right Arm	mid right
LARM	Lower Left Arm	mid left
RWRB	Lower Right Arm	posterior right top
RWRA	Lower Right Arm	anterior right bottom
LWRB	Lower Left Arm	posterior left top
LWRA	Lower Left Arm	anterior left bottom

RFIN	Right Hand	mid right anterior
LFIN	Left Hand	mid left posterior
STRN	Chest	anterior mid bottom
RTHI	Upper Right Leg	mid right top
RGTR	Upper Right Leg	mid right top
LTHI	Upper Left Leg	mid left
LGTR	Upper Left Leg	mid left top
RKNE	Upper Right Leg	mid right bottom
LKNE	Upper Left Leg	mid left bottom
RLEG	Lower Right Leg	mid right top
RTIB	Lower Right Leg	mid right
LLEG	Lower Left Leg	mid left
LTIB	Lower Left Leg	mid left
RANK	Lower Right Leg	mid right bottom
LANK	Lower Left Leg	mid left bottom

RMT5	Right Foot	anterior right top
LMT5	Left Foot	anterior left top
RTOE	Right Foot	anterior left top
LTOE	Left Foot	anterior right top
RHEE	Right Foot	posterior mid top
LHEE	Left Foot	posterior mid top
RBHD	Head	posterior mid right
LBHD	Head	posterior mid left
C7	Chest	posterior mid top
R10	Chest	posterior right top
RBAC	Chest	posterior right top
T10	Chest	posterior mid
RFWT	Hips	anterior right top
LFWT	Hips	anterior left top
RBWT	Hips	posterior mid top

LBWT	Hips	posterior mid top
RPelvis	Hips	posterior right mid

This is a fairly complete list of abbreviations but there are many operators and systems that use either more extension collections of markers or don't even follow the naming convention. A particular association for a dataset can be initially guessed and then with trial-and-error, the association can be improved. This thesis has created association files for each dataset that was analyzed. The files are simple texts files that relate the name given to the marker in the dataset and the "standard" name as given in Table 1. The format is explained in Chapter 7.3.1.

Chapter 6 RESULTS

6.1 Case Study of CMU Data 60-08

The CMU Graphics Lab produced a one minute long motion capture data-set of a salsa dance in 60-08. The data file contains 3421 time slices for 41 markers on two figures. This case study will concentrate on analyzing the performance of the UGDK in determining the rotation points in the female subject. Four passes on the data will collect rotation point calculations, each pass randomly removing from 0 to 99% of the time frames in increments of 1%. 400 calculated rotation points were collected for each segment modeled. The calculated constants are the relative rotation points as referenced in each segment's parent's coordinate system. The 400 calculations were averaged and the

standard deviations were calculated as well. These values are presented in the tables below.

Table 2 Table of Means of Rotation Points

Rotation Point	**Mean x (m)**	**Mean y (m)**	**Mean z (m)**
Waist	0.13324824	-0.063898286	0.130439024
Neck	-0.077279042	-0.003811468	-0.010261576
Left Ankle	0.437510805	-0.032639212	0.040255145
Left Wrist	0.110981565	-0.075779216	0.014702334
Left Elbow	0.283393628	-0.060979142	0.012893845
Left Knee	-0.244076283	-0.07952933	0.009066338
Right Elbow	0.253621623	-0.146993545	-0.001828167
Right Knee	0.188950453	-0.080176833	-0.003163181
Right Ankle	0.241527156	-0.065156728	0.006255086
Right Wrist	0.211446183	-0.069707086	-0.019905695
Left Shoulder	0.01607591	-0.069950812	-0.119403856
Left Hip	0.003814737	-0.019484536	-0.198284078

Right Shoulder	-0.00219989	-0.069832041	0.132929455
Right Hip	0.263017638	-0.022520684	-0.202218743

Table 3 Table of Standard Deviations of Rotation Points

Rotation Point	σ_x	σ_y	σ_z
Waist	0.003896258	0.004605284	0.014676666
Neck	0.001768695	0.001355136	0.001206322
Left Ankle	0.01981794	0.011053845	0.011233588
Left Wrist	0.002071725	0.002539415	0.000947775
Left Elbow	0.015518996	0.004050952	0.00398332
Left Knee	0.011752322	0.002762332	0.003553895
Right Elbow	0.017223326	0.00815392	0.004313482
Right Knee	0.016813465	0.003744518	0.003104466
Right Ankle	**0.129159355**	**0.034143233**	**0.020772927**
Right Wrist	0.00163169	0.000935295	0.000834657
Left Shoulder	0.001960198	0.000852649	0.00314934

Left Hip	0.001577703	0.001635193	0.004697627
Right Shoulder	0.000891381	0.00158762	0.003755364
Right Hip	0.001447724	0.00128305	0.004635324

Most of the standard deviations are less than one centimeter, but there are some significant outliers like the right ankle. Further analysis of the calculated points for the ankles and elbows show that the four runs produced two answer due to different orientations of the parent's reference frame. Therefore the standard deviation presented above for the ankles and elbows are erroneously calculating the deviation from the average of two distinct means. It is more appropriate to calculate the standard deviation from a single mean. When these outliers are removed from the calculation of the deviation, a very informative graph can be produced below. Every calculation for every segment is presented below as a deviation from the single mean rotation point versus the number of sample.

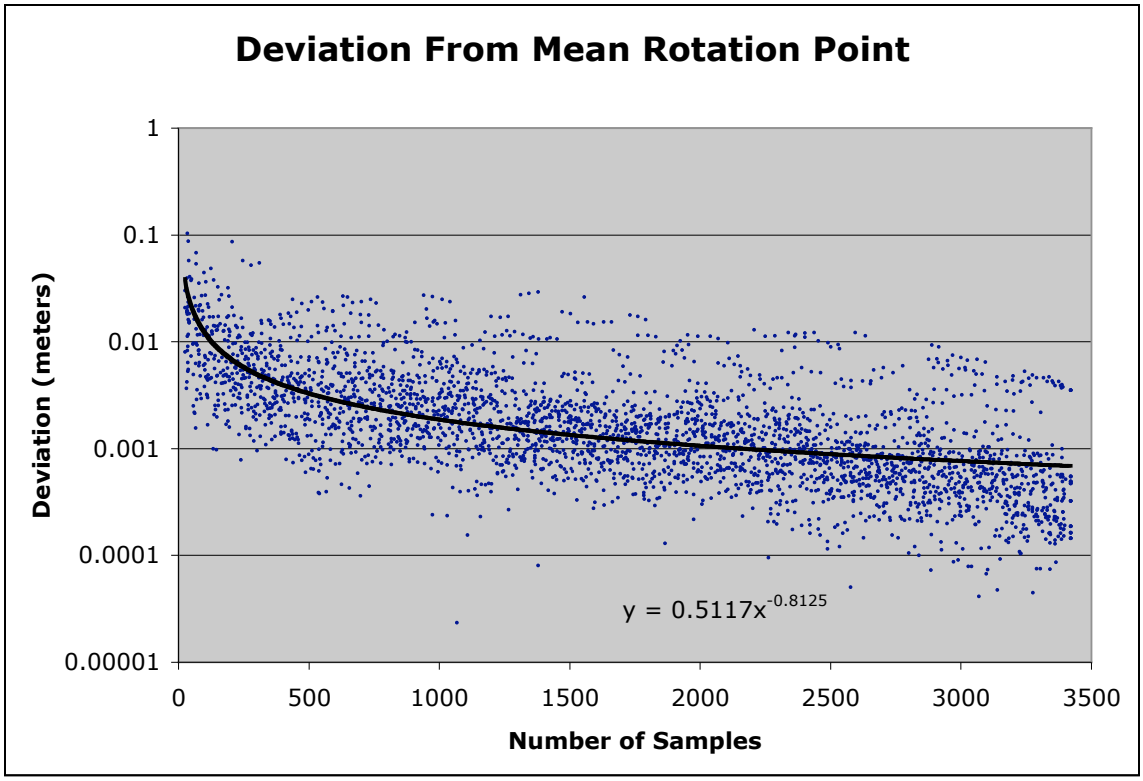

Figure 23 Inverse Power Law for Rotation Point Calculation

As can be readily seen from the above graph, a statistically significant amount of calculations are within one centimeter of accuracy when analyzing more than about 200 samples. The accuracy gets better on average with a power law close to $1/\sqrt{N}$.

6.2 Case Study of Eric Camper Data

The motion capture data in the ericcamper.c3d was analyzed to produce a skeleton. The results of the drawing produced favorable results due to the range of motion (ROM) involved in the exercise recorded. The subject did various martial arts maneuvers that moved every joint involved in drawing. One time frame is presented in the following figure. Although this is a purely qualitative analysis, the picture shows what appears to be a natural pose for all joints during a karate exercise.

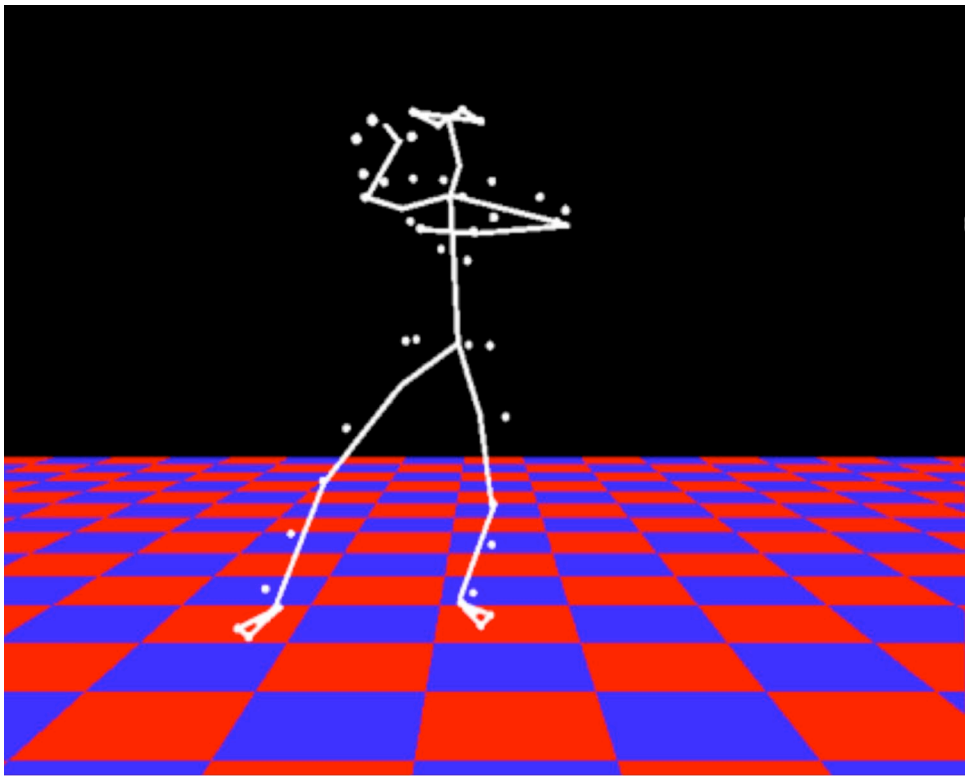

Figure 24 Eric Camper Skeleton

6.3 Comparison

The following table provides a succinct view of all of the methods studied in this thesis. It provides the positive and negative aspects of the methods. This table makes it easier to choose which solution is right for their particular situation. If speed and accuracy is of concern, then the new solution in the UGDK is the right choice. If a singular matrix is suspect, as in the case of cylindrical joints, then the SGDK is the right choice. If multiple markers are available on the joint, the MGDK produces much better answers. If storage is the major concern, as in embedded hardware applications, then the IGDK has constant space requirements.

Table 4 Comparison of Center Estimators

Method	Advantages	Disadvantages
MLE	accepted theory of best solution	slow, initial guess, may not converge
LLS	semi fast $O(236N)$, no initial guess	loss of significant figures
GDKE	fast $O(26N)$, no initial guess	biased $O(\sigma)$
UGDK*	fast $O(26N)$, no initial guess, asymptotically unbiased	slightly biased $O(\sigma/\sqrt{N})$
SGDK*	removes singular matrices	slower – need eigen solution
MGDK*	handles multiple markers, averages out errors	slightly biased $O(\sigma/\sqrt{N})$
IGDK*	space $O(1)$, time $O(123N)$	uses GDKE first

* new solutions developed in this thesis

6.4 Speed

A Monte-Carlo experiment was set up to determine the speed of the various sphere-fit algorithms. Up to a million samples were chosen on a sphere with varying measurement error, confinement angle, sphere center and sphere radius. The measurement error varied from $1x10^{-11}$ to $1x10^{12}$. The confinement angle varied from 0 to 180°. The sphere center varied as much as 2 around the origin. The radius varied 0 to 37. The linear algebra algorithms were all implemented from well-accepted implementations presented in Numerical Recipes in C [94]. The code was compiled optimized for a PowerPC G4 processor and run on a 1GHz Apple PowerBook 12". The next graph presents the timing for the four algorithms.

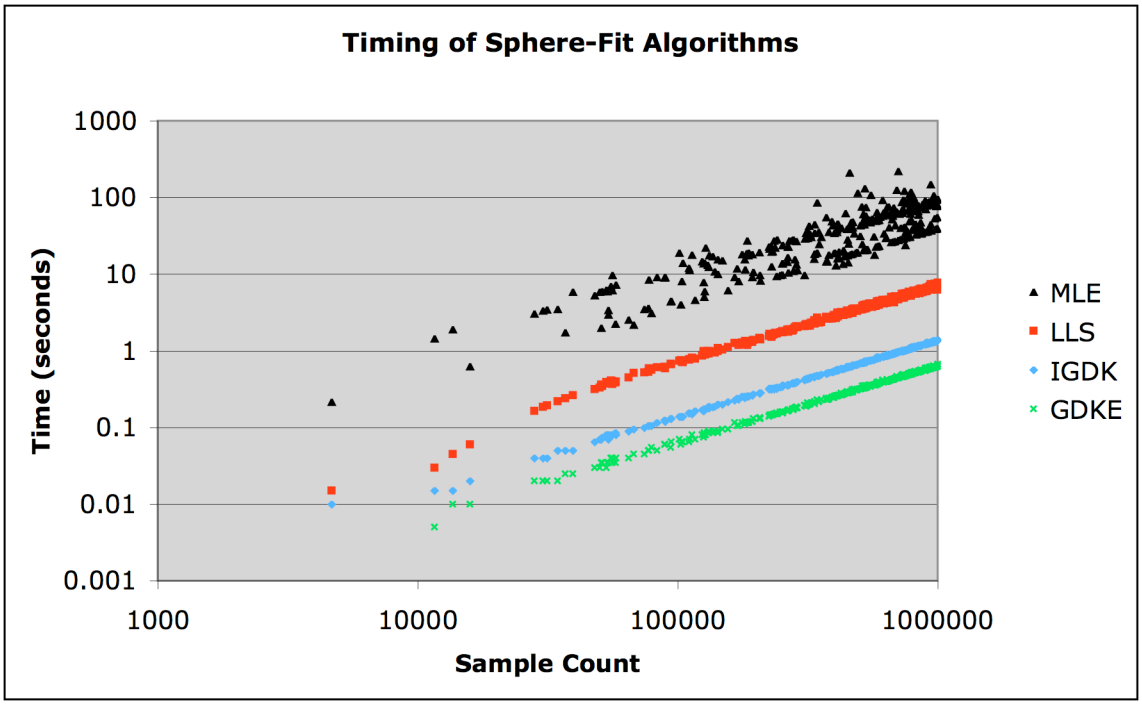

Figure 25 Timing of Algorithms

The previous graph shows that all of these algorithms are linearly dependent on sample count (i.e. $O(N)$). It further shows the MLE as messier with two distinct multi-

plication factors to this dependency. More information can be retrieved if the times are compared to the GDKE's time. The next graph shows this ratio compared to the measurement error.

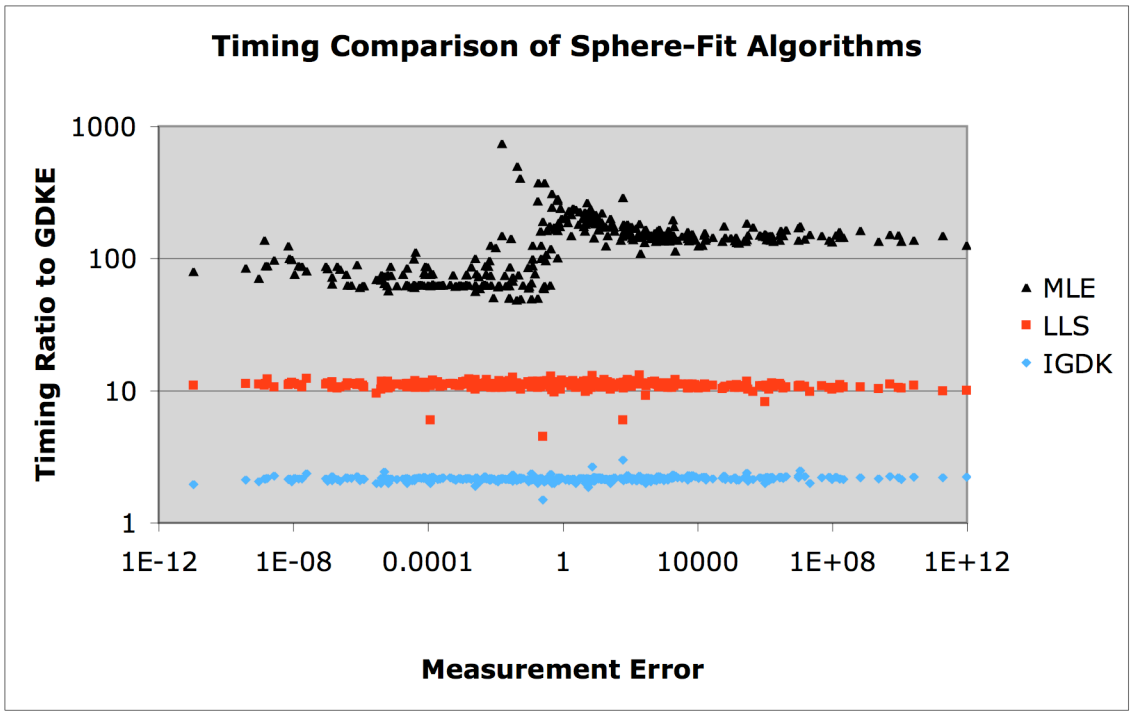

Figure 26 Timing Comparison to GDKE

As can be seen, the GDKE is always faster. The UGDK does not appear in these graphs since it has the same speed as GDKE. The GDKE is 2.17 times faster than IGDK on average. The GDKE is 11.05 times faster than the LLS method and about 80 times faster than the MLE when the measurement error is less than the radius. The FLOP count predicts this closely with LLS = 236/26 = 9.07 and IGDK = 123/26 = 4.73. The differences between the FLOP count and experiment can be accounted for in the hardware and software overhead that FLOP count never accounts for.

6.5 Conclusion

This research into the speed up of motion capture animation resulted in the discovery of the vital low-level hole that needed filling. That hole is the speedy calculations of rotation points directly from motion capture data. This thesis explains the fastest known general method for calculating these rotation points and can be as much as ten times faster than the next fastest method. This thesis demonstrates the mathematics, probabilities, and implementations for this new method for determining the center of a hypersphere. The UGDK method has further impact in a vast collection of fields as diverse as character recognition to nuclear physics where an algorithm is needed for the speedy recovery of the center of a circle or sphere. The UGDK is expected to play a vital role in the process of determining a skeleton from motion capture data. The MGDK adds robustness to the equations allowing to use every bit of available data. The IGDK further has the application of being an ideal algorithm to burn into a silicon chip whose memory requirements are constrained.

6.6 Important Contributions

First and foremost, the best contribution to the science for this dissertation is the fastest, asymptotically unbiased estimator of a hypersphere

$$\hat{c}' = \bar{x} + \tfrac{1}{2}\left(\mathbf{C} - \hat{\Sigma}\right)^{-1}\mathbf{S}$$

$$\hat{r}' = \sqrt{\tfrac{N-1}{N} Tr\left(\mathbf{C} - \hat{\Sigma}\right) + \left(\bar{x} - \hat{c}'\right)^{T}\left(\bar{x} - \hat{c}'\right)}$$

and for multiple markers

$$\hat{c}'_m = \left(\sum_{p=1}^{M} \left(\mathbf{C}_p - \hat{\Sigma} \right) \right)^{-1} \sum_{p=1}^{M} \left(\left(\mathbf{C}_p - \hat{\Sigma} \right) \bar{x}_p + \tfrac{1}{2} \mathbf{S}_p \right).$$

These made it possible for a closed form solution of a skeleton from generic, noisy motion capture data. Understanding when the best measurement conditions are for reducing the risks of measurements is important. This paper presents an easy to measure limit to strive for when trying to calculate the center of a sphere with partial coverage. The relationship

$$\sigma < r \frac{1}{\sqrt{3}} \sin^2\left(\frac{\theta}{2}\right)$$

must be satisfied in order to achieve good results using our algorithm.

Another important contribution is the full analysis of the Cramér-Rao Lower Bound for the confined points on a hypersphere. The circle CRLB is

$$CRLB_2 = \tfrac{1}{N}\sigma^2 \frac{2\theta}{2\theta(\theta + \cos\theta\sin\theta) - 4\sin^2\theta} \begin{pmatrix} \frac{2\theta(\theta + \cos\theta\sin\theta) - 4\sin^2\theta}{\theta - \cos\theta\sin\theta} & 0 & 0 \\ 0 & 2\theta & -2\sin\theta \\ 0 & -2\sin\theta & \theta + \cos\theta\sin\theta \end{pmatrix}$$

and the spherical CRLB is

$$CRLB_3 = \tfrac{1}{N}\sigma^2 \frac{3}{\sin^4\left(\frac{\theta}{2}\right)} \begin{pmatrix} \frac{\sin^2\left(\frac{\theta}{2}\right)}{(2+\cos\theta)} & 0 & 0 & 0 \\ 0 & \frac{\sin^2\left(\frac{\theta}{2}\right)}{(2+\cos\theta)} & 0 & 0 \\ 0 & 0 & 1 & -\cos^2\left(\frac{\theta}{2}\right) \\ 0 & 0 & -\cos^2\left(\frac{\theta}{2}\right) & \frac{3\cos^4\left(\frac{\theta}{2}\right) + \sin^4\left(\frac{\theta}{2}\right)}{3} \end{pmatrix}$$

These covariances of the estimators allow us to determine the best possible error in any estimator for the sphere.

The contributions of this thesis cover every aspect of generating arbitrary skeletons from motion capture data in a speedy fashion without compromising accuracy.

6.7 Further Research

The new solutions presented are an improvement to the existing science but there is more work to be done in the future. The UGDK estimator is strictly dependent on a-priori knowledge of the measurement error. It has been shown that the measurement error trace can itself be estimated but not the whole measurement covariance matrix. An estimator for the whole matrix would be most ideal but was not found in the course of this study. In addition, the statistical properties were not explored in this paper when the points do not have the same measurement covariance. This will surely introduce an additional error or bias in the estimators' answers.