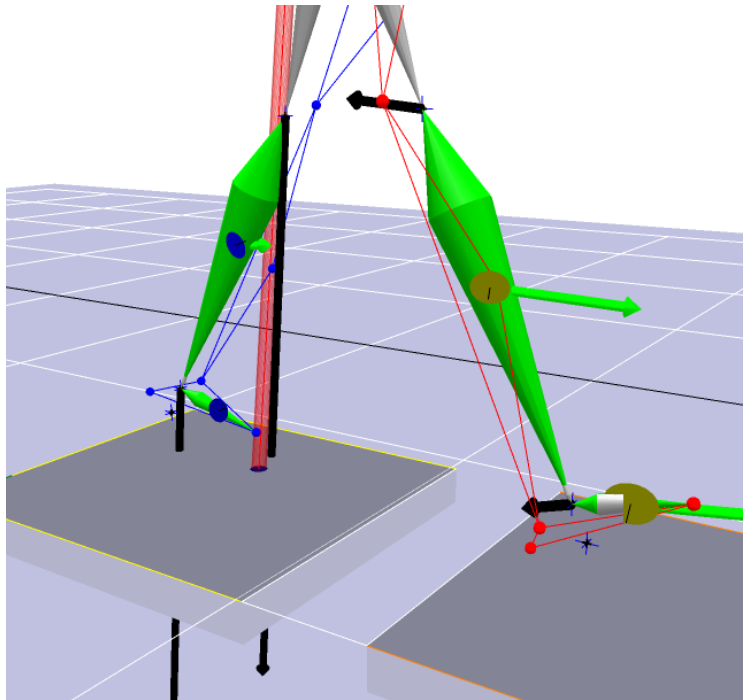


Kinetics 1.0

User's Guide

For use with Cortex 1.0



Contents

1.0 Introduction to Kinetics in Cortex	p. 2
2.0 Overview of Calculations	p. 3
2.1 Source of Forces and Moments on Segments	
2.2 Three-Dimensional Inverse Dynamics Analysis	
2.2.1 Solving for the Unknowns	
3.0 Building a Skeleton Builder Model	p. 5
3.1 Kinetics Modeling	
3.1.1 Defining Joint Centers	
3.1.1.1 Hip Joint Center	
3.1.2 Defining Skeleton Builder Segments	
3.1.2.1 Types of Skeletons used to Calculate Kinetics	
3.1.2.2 Creating a Skeleton Model to Calculate Kinetics	
3.1.2.3 Defining a Segment Coordinate System	
4.0 Mass Model Information and Details	p. 17
4.1 Mass Model Editor	
4.1.1 Zatsiorsky Mass Model Data	
4.1.2 Terminology	
4.1.3 Adjustments to Mass Inertial Information	
4.2 Calculate Kinetics Button	
4.2.1 Calculate Kinetics using the Calculate Button	
4.2.2 Calculate Kinetics using a Sky Script	
4.2.3 Viewing the Results of the Kinetics Analysis	
4.2.4 Graphical Analysis of Kinetics Data	
4.2.5 Visual Analysis of Kinetics Data	
4.2.6 Exporting Calculated Kinetics Data	
4.2.7 Calculating Kinetics, Exporting Kinetics Data, Launching MS Excel from Sky	
Appendix A: Marker Placement	p. 26
Appendix B: Sample Pictures	p. 28

Note: As Kinetics continues to evolve, the Manual is subject to change without notice. For the most up-to-date Manual, please refer to our ftp site: <ftp.motionanalysis.com>.

1.0 Introduction to Kinetics in Cortex

Kinetics is a full-body, three-dimensional engine capable of calculating kinematic and kinetic information on models that are created and saved in the project file(s). The kinetic calculations can be done on either of the two skeleton types available in Cortex: Skeleton Builder and Calcium. The Skeleton Builder models are generally simpler to use as they automatically scale the bone lengths to the subject's actual bones, so these are the models that are included and discussed. The Calcium based models use an entirely different computation method of Global Optimization. The results are similar if care is taken using either skeleton model.

The Cortex 1.0 Kinetics release includes two human models with more and less detail. These models are built using the Skeleton Builder (abbreviated SkB) modeling software in Cortex. Skeleton Builder enhancements for Cortex include two important new features to aid in building models:

- 1- User defined Euler angle rotation order, and
- 2- User definable Bone Axis.

Cortex has three check boxes above the Post Processing "Calculate" button. Calculate VMs is to calculate the Virtual Marker locations, the "Skeleton" check box tells the software that you would like to Calculate the joint angles or kinematics of the skeleton defined in your project. Finally, the "Kinetics" check box tells the software to use the subject's weight along with the entries in the Mass Model Editor table to calculate the forces and moments for each segment or bone in the model. The forces and moments are first calculated from the center of mass of each segment in the global coordinate system, and then translated to the joint center at the end of the bone.

2.0 Overview of Calculations

The Kinetics engine in Cortex applies the physics analysis of Rigid Body Dynamics to calculate the forces and moments that occurs at the joints in a body. These are the forces and moments that are necessary to make the motion of the body actually happen.

The basic physics for forces is:

$$ma = \sum forces \quad 2.1$$

And for moments, Euler's equation:

$$I \dot{\omega} + \omega \times I \omega = \sum torques \quad 2.2$$

In Biomechanics, the term moments is generally used instead of torques. Here, I will freely interchange the two because of the basic equation for torque uses both terms:

$$torque = R \times F \quad 2.3$$

where R is the moment arm vector.

In Cortex, we calculate the forces and moments at the joint between each segment and its parent. These are the forces and moments applied onto the segment by the segment's parent. There are six values: 3 forces, 3 moments.

We model our skeleton as an idealized network of rigid bodies connected by idealized joints at the joint centers.

2.1 Source of Forces and Moments on Segments

1. Gravity

Each segment has a measured mass (m) and center of mass (CoM).

$$force\ of\ gravity = mg \quad 2.4$$

where

$$g = -9.8\ m/s^2 \quad 2.5$$

2. Contact with the ground

If a segment touches the ground, then there is a ground reaction force (GRF). We include force platform data that gives us the forces and moments due to ground contact. The force platform data are processed data where all the forces are combined into the F_X , F_Y , F_Z components, and also analyzed modeling point forces on the plate. The center of pressure is calculated and the moments about the center of pressure are calculated.

3. Segment to Segment

The skeleton has a network of forces and moments that all must balance nicely. The forces and moments on a segment by its parent have the equal-and-opposite forces and moments applied onto the parent.

Each of these sources contributes to both the forces and moments. Gravity and ground contact are external forces and the segment-to-segment effects are internal (internal moments).

2.2 Three-Dimensional Inverse Dynamics Analysis

When performing a three-dimensional inverse dynamics analysis, calculations are performed from distal segment to proximal segments and the results from the distal segment are used in the analysis of the proximal segment. Winter [1] shows a very nice example of the equations used in a three-dimensional inverse dynamics analysis:

$$I_X \alpha_X + (I_Z - I_Y) \omega_Z \omega_Y = \sum M_X = R_{Xd} L_d + R_{Xp} L_p + M_{Xp} - M_{Xd}, \quad 2.6$$

$$I_Y \alpha_Y + (I_X - I_Z) \omega_X \omega_Z = \sum M_Y = R_{Yd} L_d + R_{Yp} L_p + M_{Yp} - M_{Yd}, \quad 2.7$$

$$I_Z \alpha_Z + (I_Y - I_X) \omega_Y \omega_X = \sum M_Z = R_{Zd} L_d + R_{Zp} L_p + M_{Zp} - M_{Zd}, \quad 2.8$$

Where I_Y , I_X , and I_Z are the components of the moment of inertia, α_X , α_Y , and α_Z are the components of angular acceleration, ω_X , ω_Y , and ω_Z are the components of angular velocity, M_{Xd} , M_{Yd} , and M_{Zd} are the distal joint moments, M_{Xp} , M_{Yp} , and M_{Zp} are the proximal joint moments, R_{Xd} , R_{Yd} , and R_{Zd} are the distal joint reaction forces, R_{Xp} , R_{Yp} , and R_{Zp} are the proximal joint reaction forces, and L_d and L_p are the distal and proximal distances from the center of mass to the distal and proximal joints, respectively.

2.2.1 Solving for the Unknowns

In a typical three-dimensional inverse dynamics analysis, there are six unknowns that we must solve for:

1. R_{Xp} , R_{Yp} , R_{Zp}
2. M_{Xp} , M_{Yp} , M_{Zp}

where R_{Xp} , R_{Yp} , and R_{Zp} are the proximal joint reaction forces and M_{Xp} , M_{Yp} , and M_{Zp} are the proximal joint moments. Before we can solve for the unknown proximal joint moments, we must first solve for the proximal joint reaction forces. Using Newton's Second Law of Motion, we can solve for the unknown proximal joint reaction forces:

$$R_{Xp} - R_{Xd} = \sum F_X = m_{ax}, \quad 2.9$$

$$R_{Yp} - R_{Yd} - mg = \sum F_Y = m_{ay}, \quad 2.10$$

$$R_{Zp} - R_{Zd} = \sum F_Z = m_{az}, \quad 2.11$$

where m is the segment mass and a_X , a_Y , and a_Z are the linear accelerations of the segment center of mass. After determining the proximal joint reaction forces, the proximal joint moments can be determined using Formulas 2.6, 2.7, and 2.8.

3.0 Building a Skeleton Builder Model

3.1 Kinetics Modeling

Building a Kinetics model in Cortex involves three separate and sequential steps:

1. Defining Joint Centers (using Virtual Markers)
2. Defining Skeleton Builder Segments
3. Specifying the Mass Model to be Used (Discussed in **4.0 Mass Model Information and Details**)

3.1.1 Defining Joint Centers

For more information, refer to the Cortex Tutorial ‘Defining Joint Centers using Virtual Markers’ found under Help → Tutorials → Cortex Skeleton Builder (SkB) and Kinetics – Tutorials.

Joint centers in Kinetics are defined using Virtual Markers, of which there are four types:

- i. Three Marker Value
- ii. Three Marker Ratio
- iii. Two Marker Value
- iv. Two Marker Ratio

The majority of Virtual Markers used to define joint centers will use the Three Marker Value and Two Marker Ratio definitions, although other Virtual Marker definitions may be used. Figure 1 shows the Virtual Marker Definitions Edit box.

The screenshot shows the 'Virtual Marker Definitions' dialog box. At the top, there are five radio buttons for selecting a definition type: 'Three Marker Value', 'Three Marker Ratio', 'Two Marker Value', 'Two Marker Ratio', and 'EMR'. Below these are four input fields: 'Enter Name of Virtual Marker', 'Origin Marker', 'Long Axis Marker', and 'Plane Marker'. Under 'Snap to this Marker (optional)', there are three sliders: 'Long Axis' (set to 0.00), 'Plane' (set to 0.00), and 'Perpendicular' (set to 0.00). At the bottom, there are five buttons: 'Calculate Virtual Markers', 'New V-Marker Definition', 'Delete V-Marker Definition', 'Undo', and 'Help'. Below the buttons, there are four bullet points explaining the definitions: '+ A Virtual Marker is created relative to other markers.', '+ Origin Marker has a fixed distance to the Virtual Marker.', '+ Long Axis Marker and Origin Marker define a line.', and '+ Plane, Long Axis and Origin Markers define a plane.' At the very bottom, there is a paragraph explaining that the form helps define a Virtual Marker based on other markers and that the definition becomes a permanent part of the project once saved. It also explains that the definition is used to calculate the Virtual Marker's position in each frame of a data set, with Ratio relationships treating offsets as a percentage of the distance between defining markers, and Value relationships treating offsets as a distance from the defining markers.

Figure 1: Four Virtual Marker definition types. The Origin Marker and Long Axis Marker are present for each definition, while the Plane Marker is only present in the Three Marker Value and Three Marker Ratio definitions. Since the Origin Marker is a fixed distance from the location of the Virtual Marker, it is ideal that the Origin Marker be located on a rigid part of the anatomy, i.e. a bony prominence.

Although there are many ways to define a joint center, joint centers in Kinetics will be defined by locating the boundaries of a joint (i.e., medial and lateral landmarks), and creating a landmark half the distance between those two markers. In general, joint centers can be defined by Formula 1:

$$C = L_m + p (M_m - L_m), \quad (3.1)$$

where C is the location of the joint center, L_m is the lateral joint marker, M_m is the medial joint marker, and p is the percentage offset between the lateral and medial joint markers [2]. It is worth noting that the hip joint center is defined differently, and will be discussed later. A simple example of this definition is the knee joint. The knee joint is defined by placing markers on the lateral and medial femoral condyles and then creating a joint center half way between these markers. See Figure 2 for an example.

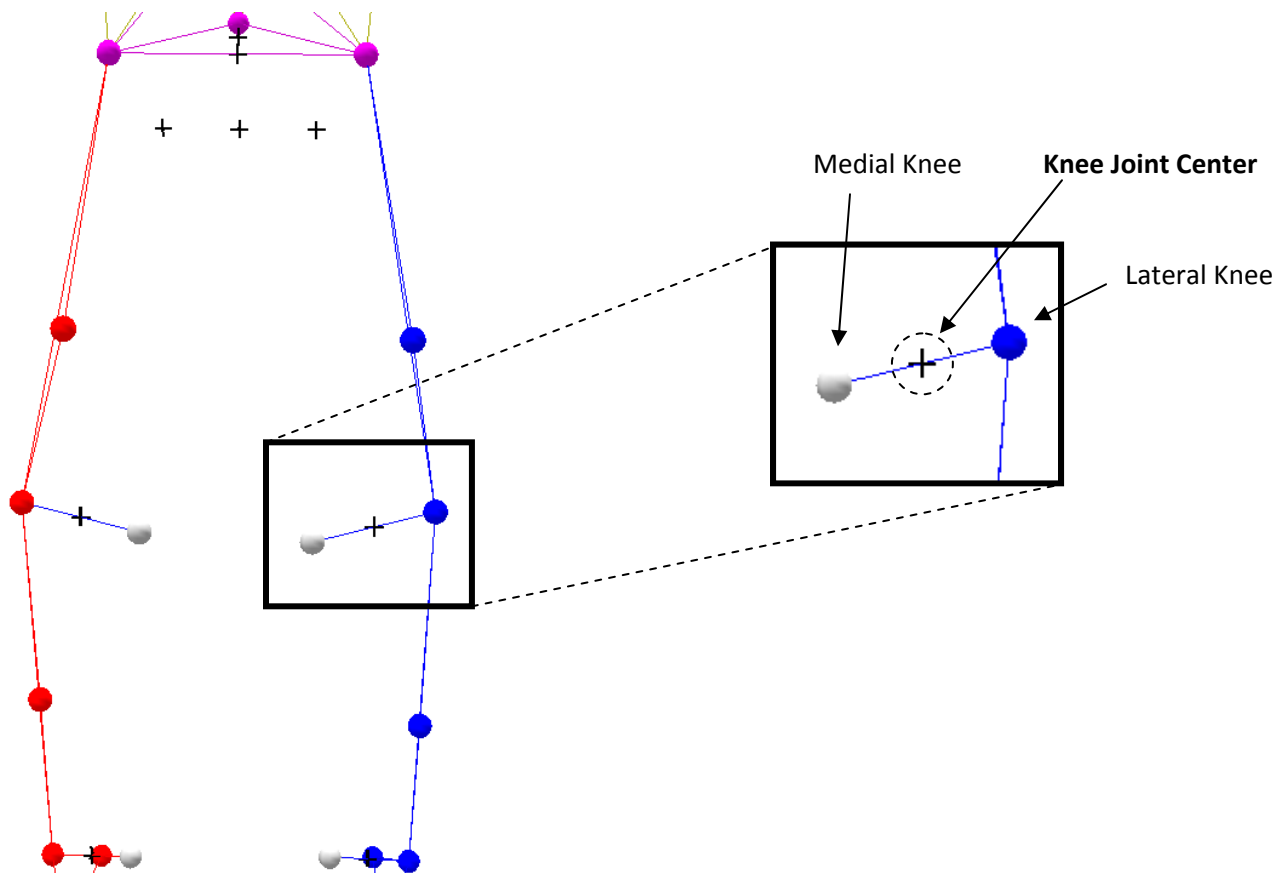


Figure 2: The Lateral Knee and Medial Knee markers are used to define the Knee Joint Center marker. A Two Marker Ratio Virtual Marker with the Lateral Knee marker set to the Origin Marker and the Medial Knee marker set to the Long Axis Marker. The Knee Joint Center is located 50% of the distance between the Lateral Knee and Medial Knee markers.

To create the joint center illustrated in Figure 2, a Two Marker Ratio Virtual Marker is used. This type of Virtual Marker definition is defined by locating the Origin and Long Axis Markers, and then specifying the percent offset along the long axis from the Origin Marker (see Formula 1). Figure 3 uses the joint center definition from above to show how the Left Knee Joint Center is created using a Two Marker Ratio Virtual Marker.

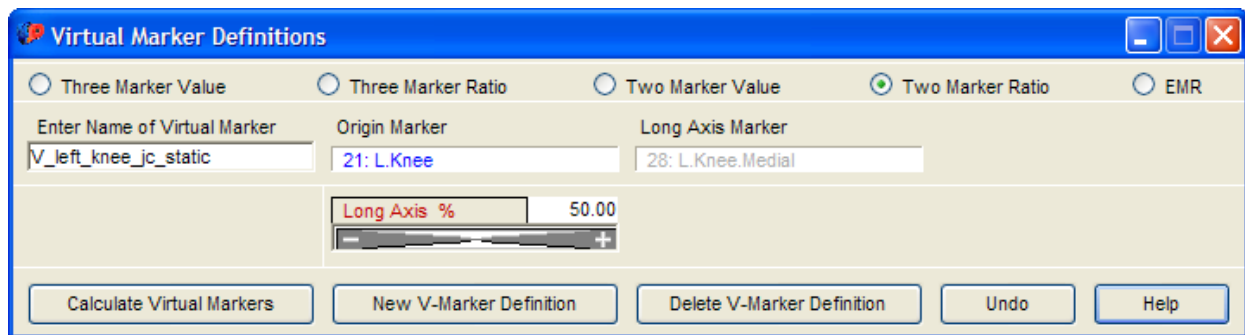


Figure 3: The Static Knee Joint Center Virtual Marker. A Two Marker Ratio is used to define the Knee Joint Center, with the L.Knee (lateral marker) as the Origin Marker and the L.Knee.Medial (medial marker) as the Long Axis Marker. The joint center is located 50% of the distance between the two specified markers.

Notice that the Left Knee Joint Center above is defined by using the medial joint marker. During many Biomechanical studies, the medial joint marker is removed after the static trial for various reasons. This causes the Left Knee Joint Center definition to become undefined. For this reason, it is essential that another Virtual Marker be created in the same location that will allow us to track the location of the joint center during dynamic trials. This second Virtual Marker, or Dynamic Virtual Marker, will use segmental markers assumed to be a fixed distance from the original joint center. See Figure 4.

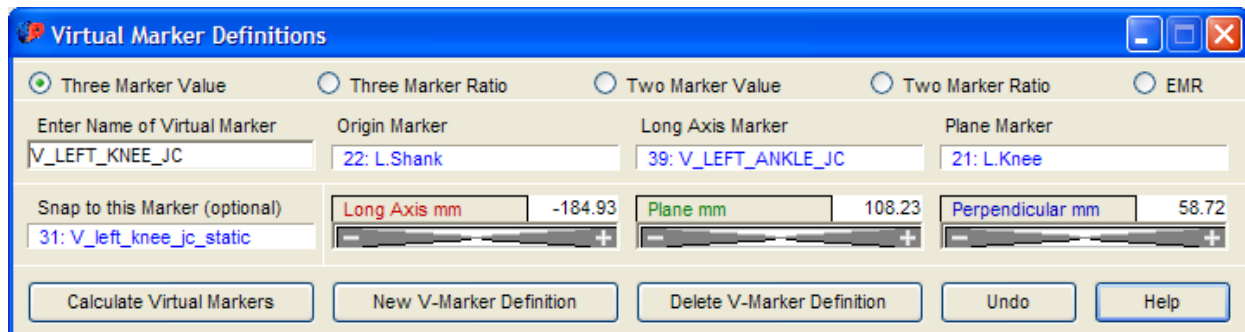


Figure 4: The Dynamic Knee Joint Center Virtual Marker. A Three Marker Value (with snap-to feature) is used to define the Dynamic Knee Joint Center. This example shows the L.Shank marker as the Origin Marker, with the Left Ankle Joint Center Virtual Marker as the Long Axis Marker and the L.Knee (lateral marker) as the Plane Marker. Since the L.Knee marker is also a fixed distance from the Dynamic Knee Joint Center Virtual Marker, it could be called the Origin Marker and the L.Shank could be called the Plane Marker.

To create the Dynamic Knee Joint Center, a Three Marker Value Virtual Marker Virtual Marker will be used. The Origin Marker is assumed to be a fixed distance from the Virtual Marker location. The Long Axis Marker defines a line and is typically the tracking marker located furthest from the Virtual Marker location. The Plane Marker will be used to define a plane with the Origin and Long Axis Markers. Note that when defining a joint center using a Virtual Marker definition, the Plane Marker does not necessarily need to define the frontal plane. If you notice beneath the Origin, Long Axis, and Plane Markers in the Virtual Marker Definition box (Figure 4), there are offsets (in mm) that you can apply to the newly defined Virtual Marker (V_LEFT_KNEE_JC). These offsets will need to be such that the new Dynamic Virtual Marker is placed in the same location as the static Virtual Marker. To do this, you will need to use the 'Snap To' feature, located in the bottom left of the Virtual Marker Definitions box. When selected, the Snap To box will become highlighted. To position the new Virtual Marker in the correct location (Snap To the location of the static Virtual Marker) you will need to select the Static Virtual Marker either from the Marker Set list or from the 3D display. Once the Static Virtual Marker is selected, the Snap To box will display its name and the position of the new Dynamic Virtual Marker will overlay on top of the Static

Virtual Marker. Press Calculate Virtual Markers and the setting for the Dynamic Virtual Marker will be saved. Notice that the offsets for the Origin, Long Axis, and Plane Markers are in mm, not percentages. The procedure of ‘Snapping To’ will need to be performed each time the marker set is reapplied to the subject or each time a new subject is tested. After the Dynamic Virtual Marker is created, the Snap To marker will be removed from the definition and the offset values will be saved in the project file. We recommend, as a precaution, that the offset values be manually recorded (i.e., in the subject data sheet) in case the Dynamic Virtual Marker needs to be recreated and the Static Virtual Marker is not available.

Although the model will have two Virtual Marker definitions for the same point, only one of these will be used during the dynamic trials. The Static Virtual Marker definition will still show up in the Marker Set list but will not be present in the 3D display. The fact that the Static Virtual Marker is still in the Marker Set list will not affect the template. The template used in Cortex uses only Real Markers – that is, physical markers that were applied to the body – and not Virtual Markers.

Because the above process of creating joint centers can sometimes be rather confusing, a summary is presented below.

Joint centers in Kinetics require a two-step process (Figure 5):

- Static Joint Center
 - o Created from joint line markers.
- Dynamic Joint Center
 - o Because medial (and sometimes lateral) markers are removed during the dynamic trial, the Static Joint Center definition will become undefined. Therefore, a second Virtual Marker definition needs to be created to track the joint center location during the dynamic trial.
 - o Created from 3 markers rigidly attached to a segment. Because the placement of these markers can vary, offsets from these markers cannot be reliably input to find a joint center. Therefore, once the 3 markers are attached to the body, the ‘Snap To’ feature will be used to place the Dynamic Joint Center in the same location as the Static Joint Center.

Figure 5 shows the process of creating the Left Knee Joint Center marker described above.

Step 1	Two Marker Ratio Virtual Marker
<i>Static Joint Center</i>	
Name	Origin Marker Long Axis Marker
V_left_knee_jc_static	L.Knee L.Knee.Medial

Step 2	Three Marker Value Virtual Marker
<i>Dynamic Joint Center</i>	
Name	Origin Marker Long Axis Marker Plane Marker
V_LEFT_KNEE_JC	L.Shank V_LEFT_ANKLE_JC L.Knee
Snap To Static Joint Center Marker V_left_knee_jc_static	

Figure 5: Joint Center Definition. Joint Center definitions are created in two steps. The first joint center (Static Joint Center) is created using a Two Marker Ratio Virtual Marker. The second joint center (Dynamic Joint Center) is created using a Three Marker Value Virtual Marker and 'Snapping To' the location of the Static Joint Center.

3.1.1.1 Hip Joint Center

There are many published methods for determining the hip joint center [2-7]. The method that will be described here is the method published by Bell [8, 9]. Coincidentally, this is the same method used in OrthoTrak. Using the ASIS breadth as the reference, the hip joint center is determined by moving 32% laterally, 22% posteriorly, and 34% inferiorly. However, Virtual Marker definitions are created slightly differently in Cortex, although the same relative percent offsets will be used. Using the midpoint between the right and left ASIS markers, the percentage offsets will now become 64% lateral, 44% posterior, and 68% inferior. These offsets are shown in Figures 6 and 7 below.

We will begin by creating a Virtual Marker located half-way between the left and right ASIS markers (V_ASIS_JC). Using this marker as the Origin, the right or left ASIS marker as the Long Axis Marker, and the Sacrum marker as the Plane Marker, we can create a joint center for the right and left hip. Figures 6 and 7 below show the Three Marker Ratio Virtual Marker definitions for the Right and Left Hip Joint Centers, respectively.

Figure 6: Right Hip Joint Center

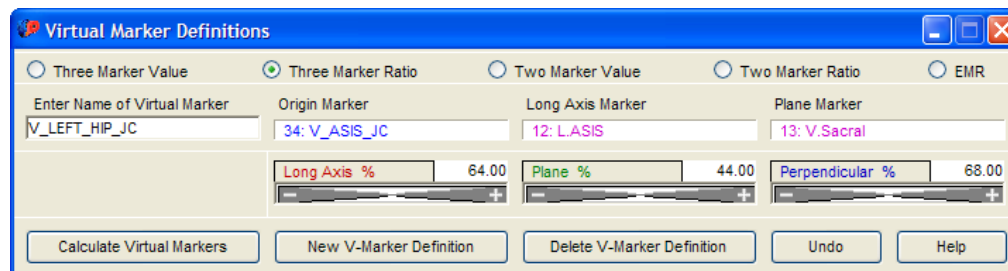


Figure 7: Left Hip Joint Center

Figure MP shows the $V_RIGHT_HIP_JC$ and $V_LEFT_HIP_JC$ Virtual Markers calculated using the Three Marker Value Virtual Marker definitions above, and the markers used to create these Virtual Markers.

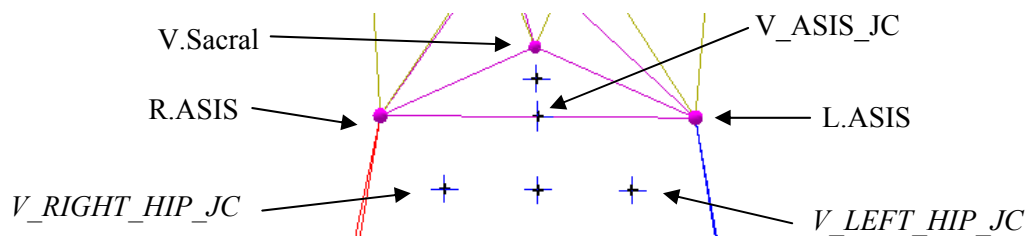


Figure 8: Right and Left Hip Joint Center Markers.

3.1.2 Defining Skeleton Builder Segments

3.1.2.1 Types of Skeletons used to Calculate Kinetics

There are four different skeletal models that can be used in Kinetics:

1. Calcium Skeleton
2. Skeleton Builder Skeleton
3. SIMM-Calcium Skeleton
4. Hybrid Skeleton

Calcium Skeleton: Calcium is a skeleton engine, used primarily in animation applications, that possesses three defining characteristics, all of which are different than the more traditional Skeleton Builder segment definitions used in Biomechanics. They are:

1. The bones are kept at constant length for the entire trial, as defined in the 'Model Pose'.
2. The degrees of freedom for each joint are user definable. For example, a joint can be constrained to 1 degree of freedom, thereby becoming a hinge joint.
3. The method of fitting the bones into the 'marker cloud' is the Global Optimization Method.

With Calcium, the user must manually position a segment within a group of markers – the 'marker cloud' – and uses the Global Optimization method of calculating the best fit position of the segment within the marker cloud during dynamic motions. Because the bone is hand-fit within the marker cloud and the length of the Calcium bone is user definable, using Calcium to define a Kinetics skeleton may not be ideal. However, Calcium does have its benefits when used in conjunction with the Skeleton Builder skeleton; we call this a Hybrid Skeleton and will be discussed later.

Skeleton Builder Skeleton: The second type of skeleton engine built into Cortex is called Skeleton Builder (SkB). This is the main skeleton engine used to calculate Kinetics for several reasons. First, the bones are scaled to the subject using joint center markers. Second, SkB segments are extremely simple to create. The user must only specify three markers to define a bone: Origin marker, Long Axis marker, and Plane marker. Third, the user can freely rotate the bone about the X, Y, or Z axes using either the rotational gizmo or by manually inputting the rotational offsets. Lastly, a SkB model is simple to learn and edit. A typical full-body SkB skeleton can be created by a novice user in less than 1 hour, and the more experienced the user, the less time it takes to create and edit the model.

SIMM-OrthoTrak Skeleton: Another type of skeleton that can be used in Cortex to calculate Kinetics is the SIMM-Calcium skeleton, formerly known as the SIMM-OrthoTrak skeleton in EVaRT. The SIMM-Calcium skeleton was developed by MusculoGraphics (the creators of SIMM) and uses an enhanced Helen Hayes marker set. Using the supplied ‘mocap.jnt’ file, the user will supply Cortex with the Static Pose trial (a.k.a., the Init Pose), and Cortex will scale the .jnt file, and therefore the model, to the subject. A description of how to create the SIMM-Calcium model can be found in **Chapter 7: Setup Tab** in the **Cortex User’s Manual**. Please note that this skeleton is not user definable and the segmental mass properties provided in the Mass Model Editor (described later) do not match the names of these segments.

Hybrid Skeleton: The last type of skeleton we will discuss is called the Hybrid skeleton. A Hybrid skeleton is just as the name implies: it is a mixture of two separate skeletons. The (desirable) characteristics of the Hybrid Skeleton are:

1. Global Optimization Method (used in Calcium) of iteratively solving for the best fit of the bone (segments) into each frames’ marker cloud, and
2. The user can specify the number of degrees of freedom for each joint (as in the Calcium Skeleton)
3. Segments are automatically scaled to the subject, based on the joint center measurements in the first frame of data (as in the Skeleton Builder Skeleton).

To create a Hybrid skeleton, the user must first define a SkB skeleton for the subject of interest. Next, an .htr file will be exported that will include the desired Euler Angle rotation sequence. The Calcium skeleton engine will be enabled and the same .htr file will be imported back into Cortex. So just as the SIMM-Calcium model is scaled for a particular subject, the Hybrid skeleton is also scaled for a particular subject. After the Hybrid skeleton is created, the user will now have the ability to edit the skeleton just as they would with a normal Calcium skeleton, bypassing the labor-intensive fitting of the skeleton segments within the marker cloud. Since the skeletal segments will have the correct segment length (determined by the static trial) and placed in the correct location, the user will be able to use this skeleton to accurately make kinematic and kinetic calculations. There are two bonuses in using the Hybrid skeleton. The first is that users can now limit the degrees of freedom (DOF) of the joints. SkB skeletons, by default, compute 6 DOF segmental information (7 DOF if you include segment length), while a Calcium model will allow the user to specify the number of DOFs which are allowed. The second bonus of the Hybrid skeleton is that the user can now assign more than 3 markers to track the segment. With Calcium using the Global Optimization Method of fitting the bone within the marker cloud, the user can benefit from a potentially more accurate segmental definition. Unfortunately, this type of skeleton is rather tedious to create and needs further testing.

Note: From this point on, all discussions of skeletons within Cortex will refer to SkB skeletons, so the words skeleton and SkB skeleton will be used interchangeably.

3.1.2.2 Creating a Skeleton Model to Calculate Kinetics

For more information, refer to the Cortex Tutorial ‘Creating a Skeleton Builder (SkB) Model’ found under Help → Tutorials → Cortex Skeleton Builder (SkB) and Kinetics – Tutorials.

Once the joint centers have been created, the user is now free to create a SkB skeletal model. The skeleton definitions are typically laid out in a parent – child relationship, with the ability to make any segment the parent of another segment. The user has the following options when creating segments:

- Define segments relative to the Global Coordinate System (GCS)
- Define segments relative to the Proximal segment
- Define segments relative to the Distal segment

Typically, models are initially created by parenting the GCS to the Pelvis segment. The Pelvis segment is considered the Root segment of the body and all other segments ‘branch out’ from this segment. In the Kinetics modeling scheme, the Pelvis is *proximal* to all body segments (if no other segments are defined relative to the GCS). Moving caudal (towards the foot) from the Pelvis, the parent segment of the Thigh would be the Pelvis, the parent segment of the Shank would be the Thigh, and so on. Moving cranially (towards the head) from the Pelvis, the parent segment of the Trunk would be the Pelvis, the parent segment of the Head would be the Trunk, and so on. Table 1 lists common parent – child relationships using the described convention.

Table 1. Parent-Child Relationships.

Common parent-child relationships, using the convention of the proximal segment as the parent and the distal segment as the child. For example, the Knee joint would be the Shank motion relative to the Thigh motion, or the Shank relative to the Thigh.

Joint	Parent Segment	Child Segment
Hip	Pelvis	Thigh
Knee	Thigh	Shank
Ankle	Shank	Foot
Waist	Pelvis	Trunk
Neck	Trunk	Head
Shoulder	Trunk	Upper Arm
Elbow	Upper Arm	Forearm
Wrist	Forearm	Hand

An example of a joint angle computed using the proximal – distal, parent – child relationship is the knee joint angle. The parent segment would be the Thigh and the child segment would be the Shank. The child segment moves with respect to the parent segment to define the joint angle. This means that the knee joint angle will be computed as the shank segment motion with respect to the thigh segment motion (shank relative to thigh).

Although the aforementioned convention is common, and is the convention found in the Kinetics sample data, it is by all means not the only accepted convention. It is up to the user to decide which convention makes most sense for their particular application and then design the model around that.

After we have decided which convention to use, and before we create a model, we must first decide on our Bone Axis (long axis) and Segment (Euler) Rotation Order. Under Model Edit → TreeView, we select the root name at the top of the list, which is the same name as the project file. This will bring up the information shown in Figure 9 below:

MarkerSet Name	Static.prj
Skeleton Engine	Skeleton Builder (SI)
Rotation Order	XYZ
Bone Axis	Z
Skin File	
Skin Transparency	0
MarkerSet Name	
MarkerSet Name	

Figure 9: Bone Properties. The (Euler) Rotation Order and Bone (Long) Axis are specified here. The Rotation Order can be set as ZYX, XYZ, YXZ, YZX, ZXY, or XZY. The Bone Axis can be set as X, Y, or Z. In our sample models, we have used Z as the bone axis to correspond with our Z-up coordinate system. We also adopt a Y forward/backward, and X medial/lateral convention for all segment coordinate systems, resulting in the XYZ (Euler) Rotation Sequence to correspond with the common 'clinical' sequence of Flexion/Extension as the first (usually largest) angle, Ab/Adduction as the second angle, and Internal/External Rotation as the third angle.

The Bone Axis and Segment Rotation Order will be the same for all segments defined in the model, so it might be necessary, depending on the movement of interest, to create two separate models (in separate project files) for processing data requiring separate Rotation Sequences.

We will use the Shank segment as an example of how to create a skeleton segment. In this example, we will create the segments proximal to distal, parenting the GCS to the Pelvis segment, the Pelvis to the Thigh, and the Thigh to the Shank. Figure 10 shows a picture of the previously defined Pelvis and Thigh segments, and the creation of the Shank segment.

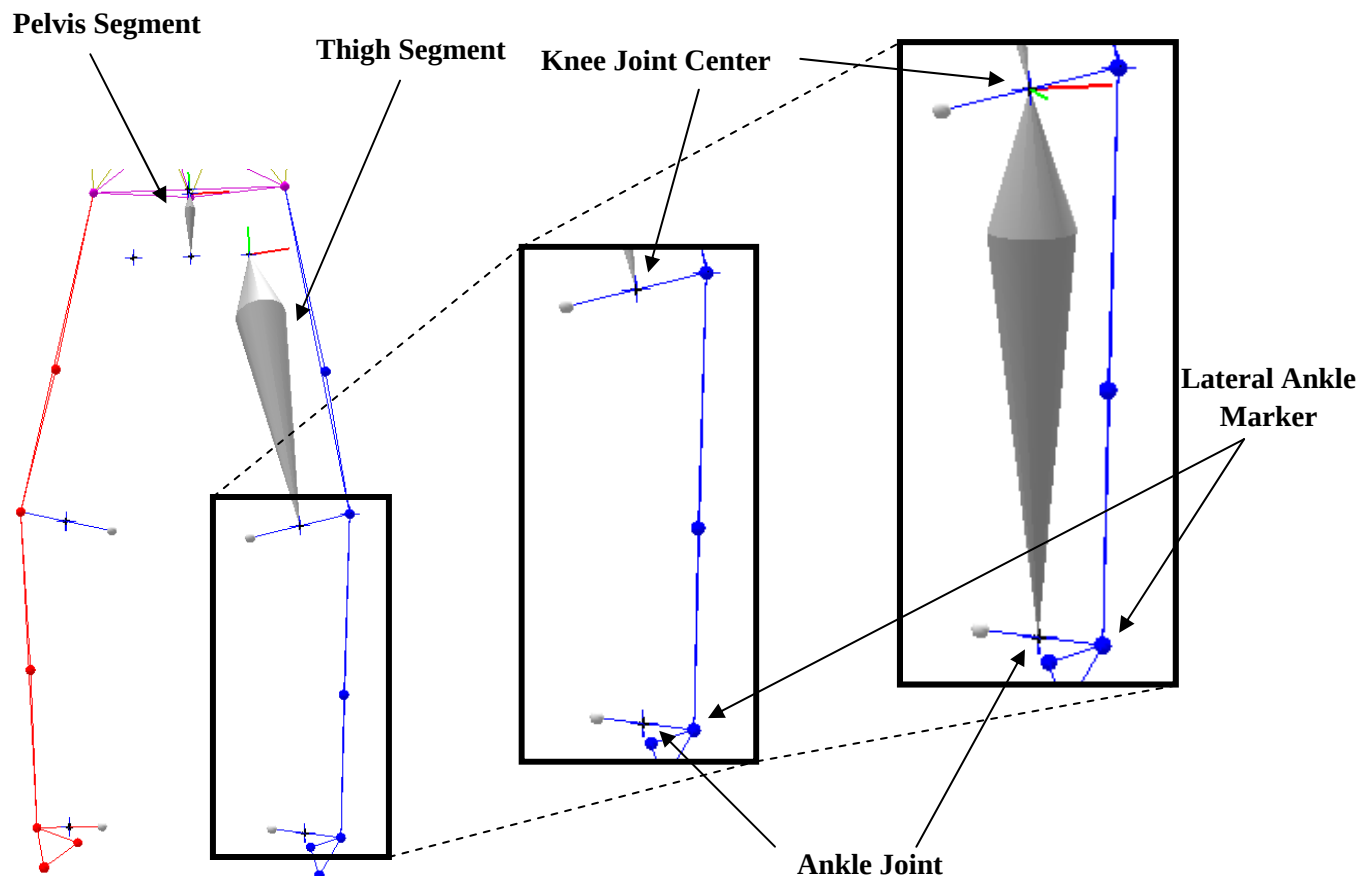


Figure 10: Defining the Shank Segment. The Shank Segment is defined with the Knee Joint Center as the Origin Marker, the Ankle Joint Center as the Long Axis Marker, and the Lateral Ankle Marker as the Plane Marker. With the Z axis as the Bone (Long) Axis, the Plane Marker will define the XZ plane. Therefore, the X axis will point towards the Lateral Ankle Marker and the Y axis will point forwards or backwards (anterioposterior axis).

To create a new segment, go to Model Edit → TreeView → SkB Segments. Right click on SkB Segments and select Insert. The new segment will appear with the default name 'MyNewSegment-1', have the Parent Segment of GLOBAL, and the Origin, Long Axis, and Plane Axis markers will be the first marker listed in the Marker Set list (in this case, the first marker in the marker set list is Top.Head). To edit the bone name, click on Name portion at the bottom of the TreeView and type in the preferred segment name. To set the Parent Segment, Origin Marker, Long Axis Marker, and Plane Axis Marker, simply click on the names at the bottom of the TreeView and choose the appropriate segment or marker name from the drop-down box. If a segmental rotation is desired, double-click on the default value (zero) next to RX, RY, or RZ Offset, and either type in the rotation angle you wish to apply (in degrees) or use the rotational gizmo to rotate the segment. Figure 11 shows the rotational gizmo for the L.Shank.

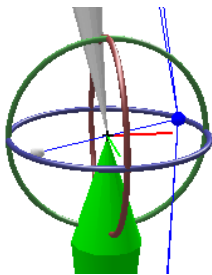


Figure 11: Rotational Gizmo. The Rotational Gizmo is used to rotate the bone segment about the axis of interest. If a rotation about the Z axis is desired, for example, grab the blue portion of the Rotational Gizmo and drag either in a clockwise or counterclockwise direction. A rotation to the bone can also be applied manually by giving a value between -360 – 360° in the RX, RY, or RZ Offsets.

The process that was used to create the L.Shank is the same process that needs to be followed to create each segment of the body. Table 2 shows the Segment Name, Parent Segment, Origin, Long Axis, and Plane Markers, and the RX, RY, and RY Rotational Offsets for the sample Helen Hayes data found in the Cortex\Samples\Skeleton Builder Kinetics folder.

MotionAnalysis

Markers TreeView

- Static.prj
 - Markers (29)
 - VMarkers (16)
 - Links (46)
 - SkB Segments (3)
 - Pelvis
 - L.Thigh
 - MyNewSegment-1**
 - Calcium Segments (inactive)

Name	MyNewSegment-1
Index	3
Parent Segment	GLOBAL
Origin Marker	Top.Head
Long Axis (Y)	Top.Head
Plane Axis (XY)	Top.Head
RX Offset	0
RY Offset	0
RZ Offset	0

RX Offset
X Rotation Angle

Table 2. Segmental Definitions.

The Segment, Parent Segment, Origin, Long Axis, and Plane Markers that define the segment, and the Rotational Offset (in degrees) for that segment found in the sample data.

Segment	Parent	Segment Markers			Offset (degrees)		
		Origin	Long Axis	Plane	RX	RY	RZ
Pelvis	GLOBAL	V_PELVIS_JC	V_MID_HIP	V.Sacral	0	0	180
L.Thigh	Pelvis	V_LEFT_HIP_JC	V_LEFT_KNEE_JC	L.Knee	0	0	90
R.Thigh	Pelvis	V_RIGHT_HIP_JC	V_RIGHT_KNEE_JC	R.Knee	0	0	-90
L.Shank	L.Thigh	V_LEFT_KNEE_JC	V_LEFT_ANKLE_JC	L.Ankle	0	0	90
R.Shank	R.Thigh	V_RIGHT_KNEE_JC	V_RIGHT_ANKLE_JC	R.Ankle	0	0	-90
L.Foot	L.Shank	V_LEFT_ANKLE_JC	L.Toe	L.Ankle	0	0	90
R.Foot	R.Shank	V_RIGHT_ANKLE_JC	R.Toe	R.Ankle	0	0	-90
Torso	Pelvis	V_PELVIS_JC	V_DISTAL_THORAX_JC	R.Shoulder	0	0	-90
Head	Torso	V_DISTAL_THORAX_JC	Top.Head	Front.Head	0	0	180
L.UpperArm	Torso	L.Shoulder	L.Elbow	R.Shoulder	0	0	-90
R.UpperArm	Torso	R.Shoulder	R.Elbow	L.Shoulder	0	0	90
L.Forearm	L.UpperArm	L.Elbow	L.Wrist	R.Elbow	0	0	-90
R.Forearm	R.UpperArm	R.Elbow	R.Wrist	L.Elbow	0	0	90
L.Hand	L.Forearm	L.Wrist	V_LEFT_HAND	R.Wrist	0	0	-90
R.Hand	R.Forearm	R.Wrist	V_RIGHT_HAND	L.Wrist	0	0	90

Note: Table 2 shows segmental information using the standard Helen Hayes marker set, which only includes two markers per segment on upper extremity segments instead of the required three. To create the upper extremity segments, it was necessary to use a contralateral (opposite limb) joint marker as the plane marker.

The rotational offsets applied in Table B are such that the X (mediolateral) axis for each bone are pointed to the subjects left, with Z along the bone and Y either forward or backward. This is the convention of my choosing and is not the only convention that can be used in Cortex. We have chosen to include this convention because it corresponds with the common XYZ rotation sequence:

- X: Flexion/Extension
- Y: Abduction/Adduction
- Z: Internal/External Rotation

Note: The rotations applied to each segment were intended to align the segmental coordinate systems, in effect, setting the 0° angle for each joint.

3.1.2.3 Defining a Segment Coordinate System

The coordinate systems in Cortex follow the Right Hand Rule.

The local (or segment) coordinate system (LCS) originates at the Origin Marker of the segment. In the Shank example from above, the Shank coordinate system originates at the Knee Joint Center marker. The long axis (Z axis) of the segment will be directed in a line from the Origin Marker to the Long Axis Marker (Ankle Joint Center). The frontal plane is defined by the plane formed by the lateral axis and long axis. The anterioposterior axis (Y) is directed anterior from the Origin Marker perpendicular to the Z axis and frontal plane, or the cross between the X and Z axes. Finally, the plane axis (X) is directed lateral from the Origin Marker perpendicular from the sagittal plane, or the cross between the Y and Z axes.

See Figure 12 for a step-by-step description of how the LCS is defined in Cortex. This example corresponds with the sample data found in the Cortex\Samples\Skeleton Builder Kinetics folder.



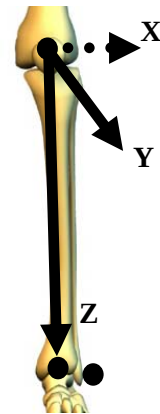
- 1) The long axis (Z) along the bone is defined as the Origin Marker to the Long Axis Marker.



- 2) The frontal plane (XZ) is defined as the plane formed by the long axis and plane axis.



- 3) The anteroposterior (Y) axis forward is defined as the cross between the Z axis and X axis.



- 4) The mediolateral (X) axis to the left is defined as the cross between the Y axis and Z axis.

Figure 12: Defining a Segment Coordinate System

4.0 Mass Model Information and Details

4.1 Mass Model Editor

The Mass Model Editor built into Cortex is the heart of the Kinetics package. It is here where the segments in the model are displayed, and the segmental characteristics are specified to perform the Kinetics analyses. To launch the Mass Model Editor, shown in Figure HU, go to Tools → Mass Model Editor...

Segment	Mass	Center of Mass			Radius of Gyration		
		X	Y	Z	X	Y	Z
Pelvis	0.1117	0	0	0.5	0.551	0.615	0.587
R.Thigh	0.1416	0	0	0.4095	0.329	0.329	0.149
L.Thigh	0.1416	0	0	0.4095	0.329	0.329	0.149
R.Shank	0.0433	0	0	0.4395	0.251	0.246	0.102
L.Shank	0.0433	0	0	0.4395	0.251	0.246	0.102
R.Foot	0.0137	0	0	0.4415	0.257	0.245	0.124
L.Foot	0.0137	0	0	0.4415	0.257	0.245	0.124
Torso	0.3229	0	0	0.5047	0.294	0.342	0.233
Head	0.0694	0	0	0.5976	0.362	0.376	0.312
R.UpperArm	0.0271	0	0	0.5772	0.285	0.269	0.158
L.UpperArm	0.0271	0	0	0.5772	0.285	0.269	0.158
R.Foot	0.0137	0	0	0.4415	0.257	0.245	0.124

Figure 13: Mass Model Editor. Mass and inertial properties for each segment are specified in this table, along with the subject information (name, gender, height, and weight). The published Zatsiorsky-Seluyanov's segmental inertial data (modified by de Leva) is predefined in the Mass Model Editor, although custom mass inertial information can be specified. Data shown in this table correspond to the sample data provided in the Cortex\Samples folder, where Z is the axis along the bone, X is the mediolateral axis, and Y is the anteroposterior axis.

In the Mass Model Editor, the user will specify the subject name, height, weight, and gender. It also allows the user to choose from one of two options in specifying mass inertial information: the Zatsiorsky mass inertial information (described below) or input Custom mass inertial information. When the project file is saved, the information input into the Mass Model Editor is stored and reloaded when the Mass Model Editor is launched.

4.1.1 Zatsiorsky Mass Model Data

The mass model information used in Kinetics 1.0 is taken from **De Leva, Paolo. (1996). Adjustments to Zatsiorsky-Seluyanov's segment inertia parameters. J Biomechanics 29, 1223 – 1230**, but we plan to add more in the near future.

When the Zatsiorsky radio button is selected in the Mass Model Editor, the predefined mass inertial information in Kinetics will be loaded for the predefined segment names. The Zatsiorsky mass inertial data are based on a 15-segment, full-body model, and includes the following segments:

Pelvis	R.Foot	L.UpperArm
R.Thigh	L.Foot	R.Forearm
L.Thigh	Torso	L.Forearm
R.Shank	Head	R.Hand
L.Shank	R.UpperArm	L.Hand

To use the Zatsiorsky model, the segment names that are entered into your Skeleton Builder or Calcium model must match the names listed above, or the mass properties for that segment will not be automatically loaded. If any or all of the above names are not present, you will receive a warning message stating that the Mass Model Editor cannot find certain segments. However, this does not mean that you will not be able to input mass and inertial information for this model. By selecting the Custom radio button, you can hand edit the Mass Model Editor and input your desired mass model information.

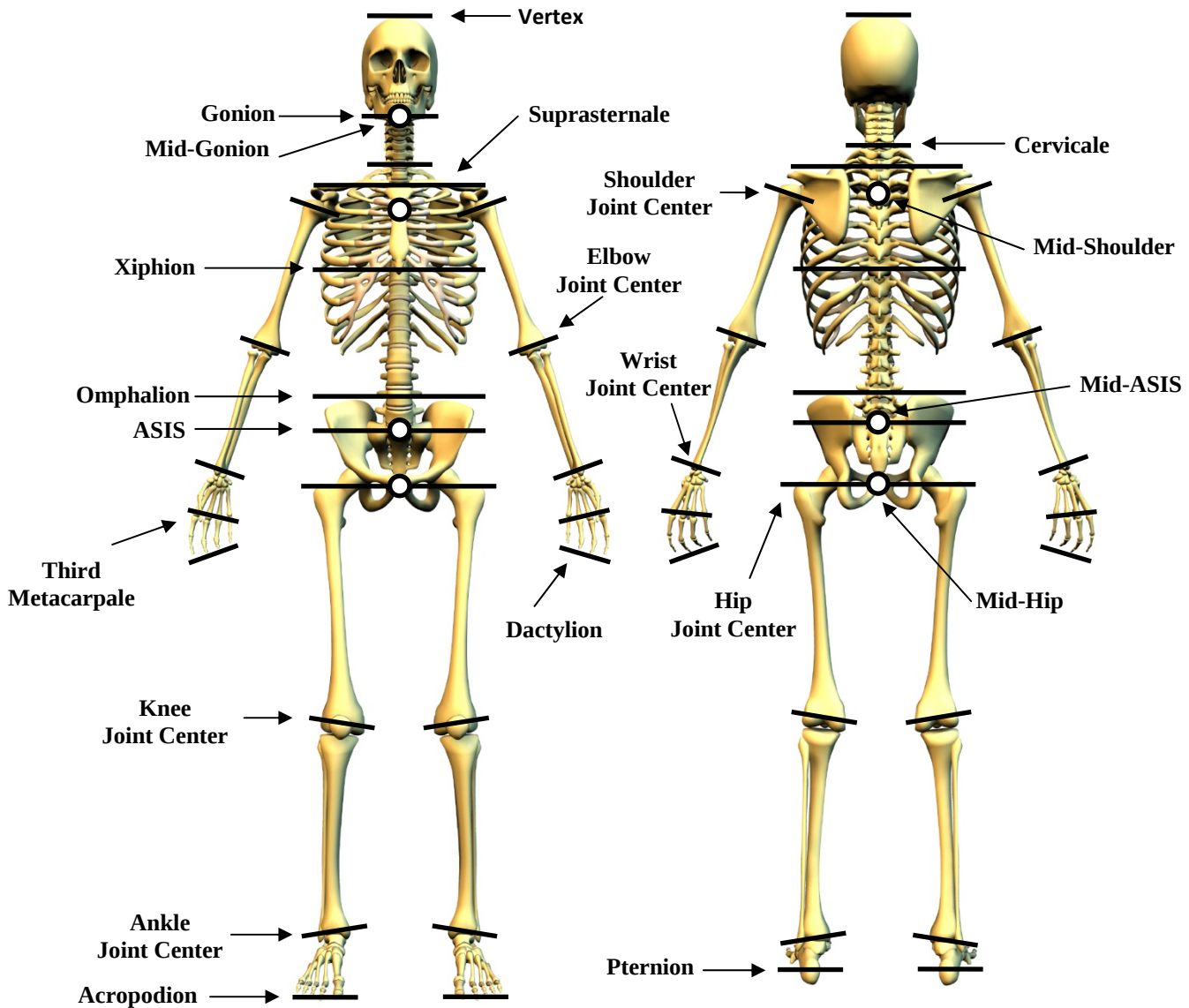
4.1.2 Terminology

To help better understand the terminology in Zatsiorsky's body segment descriptions, the following definitions have been included in Table BP, followed by a graphical description of the location of landmarks (Figure 14). (adapted from de Leva, 1996).

Table 3. Body Segment Terminology used to Identify Landmarks

Landmark Name	Abbreviation	Description
Acromion	ACRO	Most lateral point on the lateral edge of the acromial process of the scapula
Acropodion	TTIP	Tip of longest toe
Bispinous breadth	BB	Distance between two ASIS
Cervicale	CERV	Superior tip of the spine of the 7 th cervical vertebra
Dactylion (3 rd)	3DAC	Tip of 3 rd digit
Gonion	GONI	Most lateral point on the posterior angle of mandible
Iliospinale	ASIS	Inferior point of one of the anterior superior iliac spines
Malleoli	MMAL, LMAL	Medial and lateral bony projections of the malleolus
Metacarpale (3 rd)	3MET	Distal palpable point on the metacarpal of the 3 rd digit on the dorsal hand
Mid-gonion	MIDG	Point midway between 2 gonion
Mid-hip	MIDH	Point midway between 2 hip joint centers
Mid-shoulder	MIDS	Point midway between 2 shoulder joint centers
Omphalion	OMPH	Center of navel
Pternion	HEEL	Posterior point of the heel
Radiale	RADI	Lateral tip on the proximal head of the radius
Sphyrion (tibia)	TSPH	Distal tip of the tibia – distal to medial malleolus
Sphyrion fibulare	FSPH	Distal tip of the fibula – distal to lateral malleolus
Stylian	RSTY	Distal dip of the styloid process of the radius
Suprasternale	SUPR	Most caudal point on the jugular notch on the sternum
Tibiale (medial)	MTIB	Most proximal point on the medial superior border of the head of the tibia
Tibiale laterale	LTIB	Most proximal point on the lateral superior border of the head of the fibula
Trochanterion	TROC	Superior border on the greater trochanter of the femur
Vertex	VERT	Uppermost part of the head
Xiphion	XYPH	Lowermost end of the sternum

Figure 14. Graphical Description of Landmark Location



4.1.3 Adjustments to Mass Inertial Information

Table 4 below shows the mass inertial information contained in de Leva's paper. The information in this table comes directly from the paper with the exception of the following segments:

1. The Torso segment was added as the default segment in place of the Upper Trunk (UT) and Middle Trunk (MT). The segment masses for the UT and MT for both males and females were combined to create the Torso. The percent mass of the Torso for females is 30.10% and for males is 32.29%. The position of the segment CoM was also carried over.
2. The Torso segment is defined as Suprasternale to mid-ASIS.
3. The Pelvis segment is defined as mid-ASIS to mid-Hip Joint Center. The Pelvis segment has the same mass as the Lower Trunk segment in de Leva's paper.

Table 4. Segmental Inertia Information. Data are from de Leva, 1996. [10]

All measurements are indicated as fraction of the whole. CoM = center of mass, k = radius of gyration

	Segment Endpoints		Mass		Longitudinal CoM		Sagittal k		Transverse k		Longitudinal k	
	Cranial	Caudal	F	M	F	M	F	M	F	M	F	M
Head	VERT	MIDG	0.0668	0.0694	0.5894	0.5976	0.330	0.362	0.359	0.376	0.318	0.312
UpperArm	SJC	EJC	0.0255	0.0271	0.5754	0.5772	0.278	0.285	0.260	0.269	0.148	0.158
Forearm	EJC	WJC	0.0138	0.0162	0.4559	0.4574	0.261	0.276	0.257	0.265	0.094	0.121
Hand	WJC	MET3	0.0056	0.0061	0.7474	0.7900	0.531	0.628	0.454	0.513	0.335	0.401
Torso	SUPR	MIDA	0.3010	0.3229	0.3741	0.4166	0.307	0.328	0.292	0.306	0.147	0.169
Pelvis	MIDA	MIDH	0.1247	0.1117	0.5000	0.5000	0.433	0.615	0.402	0.551	0.444	0.587
Thigh	HJC	KJC	0.1478	0.1416	0.3612	0.4095	0.369	0.329	0.364	0.329	0.162	0.149
Shank	KJC	AJC	0.0481	0.0433	0.4352	0.4395	0.267	0.251	0.263	0.246	0.092	0.102
Foot	HEEL	TTIP	0.0129	0.0137	0.4014	0.4415	0.299	0.257	0.279	0.245	0.139	0.124
Alternate Segments: Note, the information supplied below may not necessarily correspond with the sample models provided												
Head	VERT	CERV	0.0668	0.0694	0.4841	0.5002	0.271	0.303	0.295	0.315	0.261	0.261
Trunk	SUPR	MIDH	0.4257	0.4346	0.4151	0.4486	0.357	0.372	0.339	0.347	0.171	0.191
Trunk	CERV	MIDH	0.4257	0.4346	0.4964	0.5138	0.307	0.328	0.292	0.306	0.147	0.169
Trunk	MIDS	MIDH	0.4257	0.4346	0.3782	0.4310	0.379	0.384	0.361	0.358	0.182	0.197
UpperTrunk	SUPR	XYPH	0.1545	0.1596	0.2077	0.2999	0.746	0.716	0.502	0.454	0.718	0.659
UpperTrunk	CERV	XYPH	0.1545	0.1596	0.5050	0.5066	0.466	0.505	0.314	0.320	0.449	0.465
MiddleTrunk	XYPH	OMPH	0.1465	0.1633	0.4512	0.4502	0.433	0.482	0.354	0.383	0.415	0.468
LowerTrunk	OMPH	MIDH	0.1247	0.1117	0.4920	0.6115	0.433	0.615	0.402	0.551	0.444	0.587
Forearm	EJC	STYL	0.0138	0.0162	0.4592	0.4608	0.263	0.278	0.259	0.267	0.095	0.122
Hand	WJC	DAC3	0.0056	0.0061	0.3427	0.3624	0.244	0.288	0.208	0.235	0.154	0.184
Hand	STYL	DAC3	0.0056	0.0061	0.3502	0.3691	0.241	0.285	0.206	0.233	0.152	0.182
Hand	STYL	MET3	0.0056	0.0061	0.7534	0.7948	0.519	0.614	0.443	0.502	0.327	0.392
Shank	KJC	LMAL	0.0481	0.0433	0.4416	0.4459	0.271	0.255	0.267	0.249	0.093	0.103
Shank	KJC	SPHY	0.0481	0.0433	0.4481	0.4524	0.275	0.258	0.271	0.253	0.094	0.105

Note: When the Zatsiorsky radio button is selected in the Mass Model Editor, the first 9 segments listed in Table 4 will be loaded. The Alternate Segments can be used in place of the default segments by selecting the Custom tab, and inputting the alternate segmental information.

4.2 Calculate Kinetics Button

Once the skeleton model is defined and the mass model information has been entered into the Mass Model Edit table, a Kinetics analysis can now be performed. There are three ways in which Kinetics can be calculated within Cortex:

1. Calculate Kinetics button in the Mass Model Editor
2. Calculate button located at the bottom left of the dashboard with the Kinetics box checked
3. Run a Sky script

4.2.1 Calculate Kinetics using the Calculate Button

In the Mass Model Editor, there is a built-in button for calculating Kinetics. Figure 15 below.

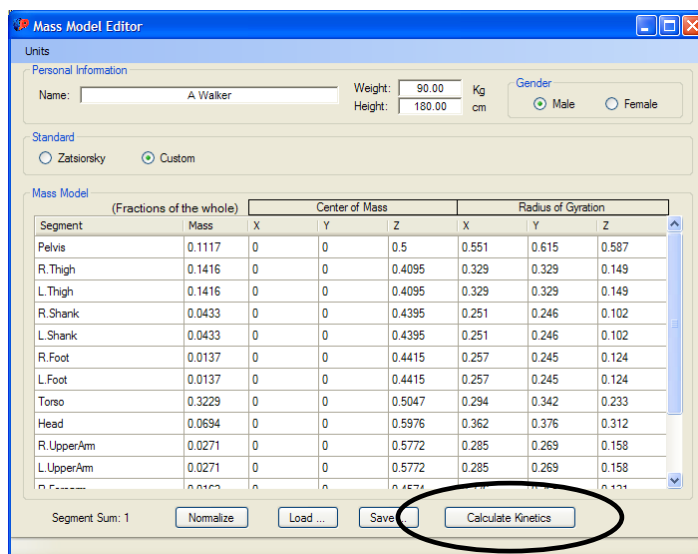


Figure 15: Calculate Kinetics button located in the Mass Model Editor.

After the mass inertia information are specified, kinetics and kinematics data for an individual trial can be calculated. Another option to calculate Kinetics is to use the Calculate button located at the bottom left of the dashboard (Figure 16).

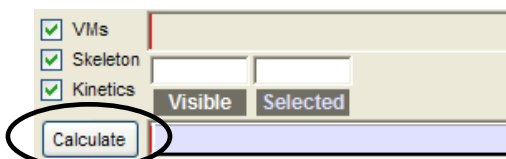


Figure 16: Calculate Button located in the Dashboard. This button will calculate Virtual Markers, the Skeleton (SkB or Calcium), and Kinetics data. Options can be toggled on and off by selecting /deselecting them.

The Calculate button on the dashboard performs three separate functions: calculate Virtual Markers, calculate the defined Skeleton, and calculate Kinetics. This Calculate button was designed to work very smoothly with Skeleton Builder or Calcium Kinetics so that once your mass model information was defined for a particular subject, you would no longer need to launch the Mass Model Editor to each time you wish to perform a Kinetics analysis.

Note that the Calculate Kinetics button located in the Mass Model Edit table only performs the Calculate Kinetics function, and *does not* calculate the Virtual Markers or Skeleton.

4.2.2 Calculate Kinetics using a Sky Script

The final way you can calculate Kinetics in Cortex is using the Sky interface. See **Chapter 14: Sky** in the **Cortex User's Manual** for a description of Sky. There are three Sky functions related to Kinetics:

1. swKinetics_Calculate
 - Initiates the calculation of Kinetics data
2. swKinetics_SetFileFormatStyle
 - Specifies the file format of Kinetics data when exported
3. swKinetics_ExportKineticsFile
 - Exports Kinetics data to the format specified above

These functions can be used to create a customizable Sky script to automate the process of calculating Kinetics and exporting the data. If assistance with these or any other scripting functions within Sky is needed, please contact Motion Analysis Customer Support at support@motionanalysis.com, or refer to Chapter 14: Sky in the Cortex User's Manual.

4.2.3 Viewing the Results of the Kinetics Analysis

Within Cortex, there are two separate ways Kinetics data can be viewed. The first is in a graphical form within the Skeleton Graphs view and the second is in the 3D display on the skeleton itself.

4.2.4 Graphical Analysis of Kinetics Data

One of the more functional ways to view Kinetics data is in a graphical form. Within Cortex, this can be done using the Skeleton Graphs data view or outside Cortex using the .kin file, which will be explained later. To display the Skeleton Graphs, either the F6 hot key can be pressed or the view can be displayed by selecting Data Views → Layouts → Skeleton Graphs (F6). Since this display does not graph marker data, the Segments tab must also be displayed. The Segments tab is located next to the Marker tab in the Post Process Panel on the right (discussed below).

Kinetics data will be displayed according to the parent – child relationship that was specified when the segment was created. For example, if you choose to view the left knee joint data, select the child segment (L.Shank) from the Segments tab in the Post Process Panel. In Chapter 3, we defined the left knee joint as the L.Shank relative to the L.Thigh.

The Skeleton Graphs, shown in Figure 17 below, display the kinematics and kinetics data for each selected segment and joint. The following information is contained in the Skeleton Graphs:

- Joint Angles (deg)
- Joint Forces (N)
- Joint Moments (N·m)
- Linear Movement (segment w.r.t. Global Coordinate System)
 - Linear Velocity ($\text{mm}\cdot\text{s}^{-1}$)
 - Linear Acceleration ($\text{mm}\cdot\text{s}^{-2}$)
- Angular Movement (segment w.r.t. Global Coordinate System)
 - Angular Velocity ($\text{deg}\cdot\text{s}^{-1}$)
 - Angular Acceleration ($\text{deg}\cdot\text{s}^{-2}$)

The data contained within the Skeleton Graphs can be written out in a .kin file (described later), depending on how the segments are defined. The Skeleton Graphs show two types of information: segment w.r.t. segment data (joint angles, forces, moments) and segment w.r.t. Global Coordinate System (GCS) data (linear velocity and acceleration, angular velocity and acceleration). If the segments are defined in a parent – child relationship, the data in the .kin file will contain joint information. If the segments are defined w.r.t. GCS, the data in the .kin file will contain segmental information only (no joint information).

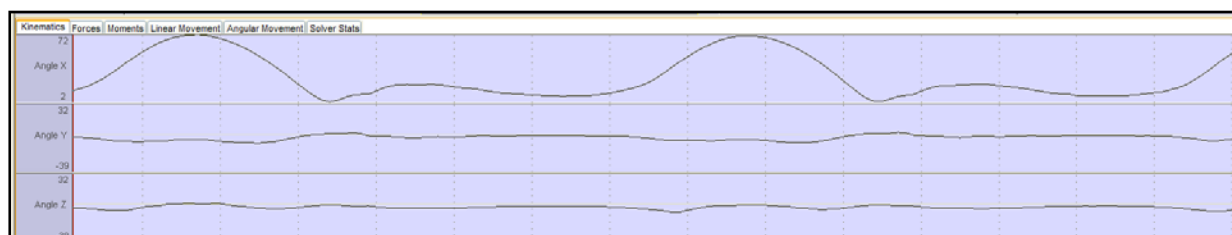


Figure 17: Skeleton Graphs. Kinetics data can be viewed in a graphical form by displaying the Skeleton Graphs and select the segment, and therefore joint, you wish to view. The joint angle (in deg), joint forces (in N), joint moments (in N·m), linear velocity (mm/s) and linear acceleration (mm/s²), and angular velocity (deg/s) and angular acceleration (deg/s²)

The Segments Tab located in the Post Process Panel will allow the user to specify which segments are displayed in either the 3D Display or in the Skeleton Graphs. Segments can be selected in the same fashion as markers: Ctrl + left mouse click, Shift + left mouse click, or left mouse click and drag. When a segment is selected in the Segments tab, it will become highlighted and turn green in the 3D Display. Figure 18 shows the Segments Tab. The L.Shank segment is selected in the Segments Tab and the results are shown in the 3D Display.

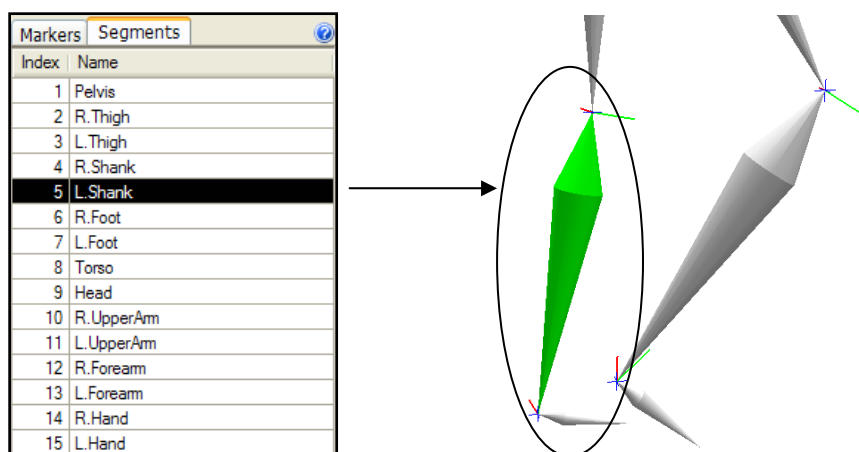


Figure 18: Segments Tab. Select the segment of interest to display the joint kinematics and joint kinetics in the Skeleton Graphs. Multiple segments can be selected at one time by either Ctrl + left mouse click, Shift + left mouse click, or holding down the left mouse button and dragging down the Segments list.

4.2.5 Visual Analysis of Kinetics Data

During data analysis, it is often times advantageous to study a dynamic illustration rather than a graphical representation. This is a handy feature within Kinetics. By selecting a segment (and therefore a joint), you have the ability within Cortex to visualize the magnitudes of the kinematic and kinetic data. In the 3D display, right click to activate the 3D Display Show Properties box. At the bottom of the window, there will be a Kinetics section, shown in Figure 19 below, which will allow you to select the available kinematic or kinetic variables to visualize.

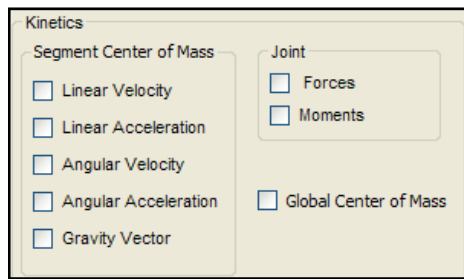


Figure 19: Kinetics display box in the 3D Display Show Properties box.

Segment Center of Mass options on the left include kinematic data while Joint options on the right include kinetic data. The Segment Center of Mass and Global Center of Mass options are calculated directly from the information provided in the Mass Model Editor while the Joint options are calculated from the inverse dynamics analysis. Figure 20 below shows an example of the Linear Velocity (green) and Linear Acceleration (blue) in addition to the Global (whole body) Center of Mass (red dot) and Joint Forces (black) for the L.Thigh, L.Shank, and L.Foot segments.

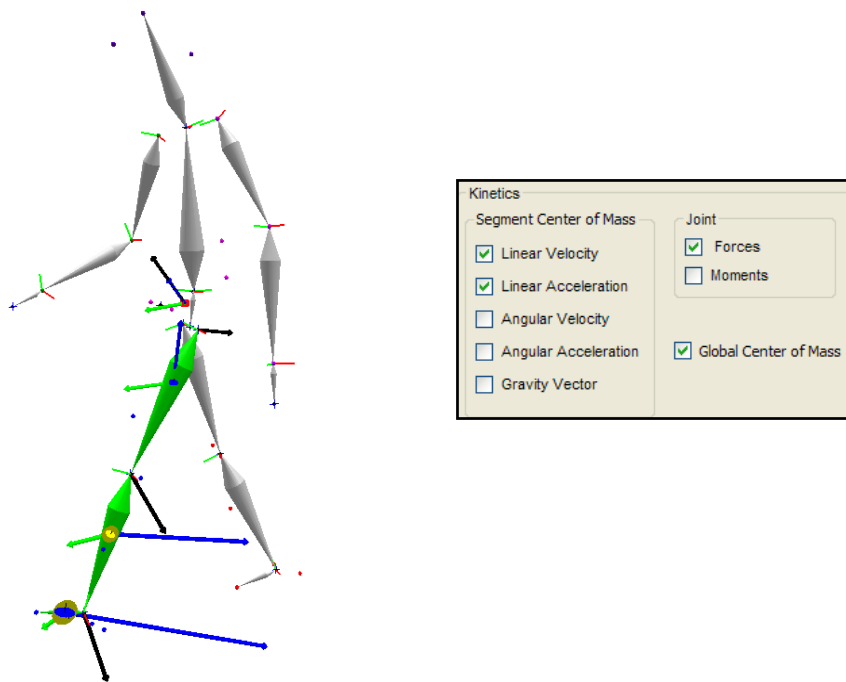


Figure 20: Kinetics and kinematics options in the Kinetics display box.

The magnitudes of the vectors correspond with the calculated data exported in the .kin file.

4.2.6 Exporting Calculated Kinetics Data

After calculating Kinetics, you will have the option to export the calculated data in one of two ASCII-friendly formats:

- Kinematics and Kinetics...
 - Exports center of mass (CoM) position, joint angles, segment length, joint forces, and joint moments.
- Kinetics only...
 - Exports ONLY joint forces and joint moments.

You can find these two options under File → Export Kinetics File... Both .kin files follow the Frame Oriented style of displaying data. Figure 21 shows a sample portion of the ‘Kinematics and Kinetics’ .kin file for the Pelvis segment.

[Data]													
#	Pelvis												
#Frame	x	y	z	rx	ry	rz	l	fx	fy	fz	mx	my	mz
	CoM position of each segment (in mm) with respect to the Global Coordinate System			Angle of each segment with respect to its parent segment (joint angle – in degrees)			↓ Segment length (in mm)	Joint force (in N) located at the joint center. This shows the overall force effects of the segment.			Joint moment (in Nm) located at the joint center.		

Figure 21: Sample portion of the ‘Kinematics and Kinetics’ .kin file. The data included in this file format include CoM position of each segment, joint angles (in degrees), segment length (in mm), joint forces (in N), and joint moments (in N·m).

The .kin files exported from Cortex can be read into MATLAB, Labview, or any other graphical programming tool, as well as into Microsoft Excel. Please email support@motionanalysis.com for a sample MATLAB script that will read in the ASCII-friendly .kin file and graph the sample data. Note that both .kin files are subject to change with future Cortex releases as more features are added to Kinetics.

4.2.7 Calculating Kinetics, Exporting Kinetics Data, Launching Microsoft Excel from Sky

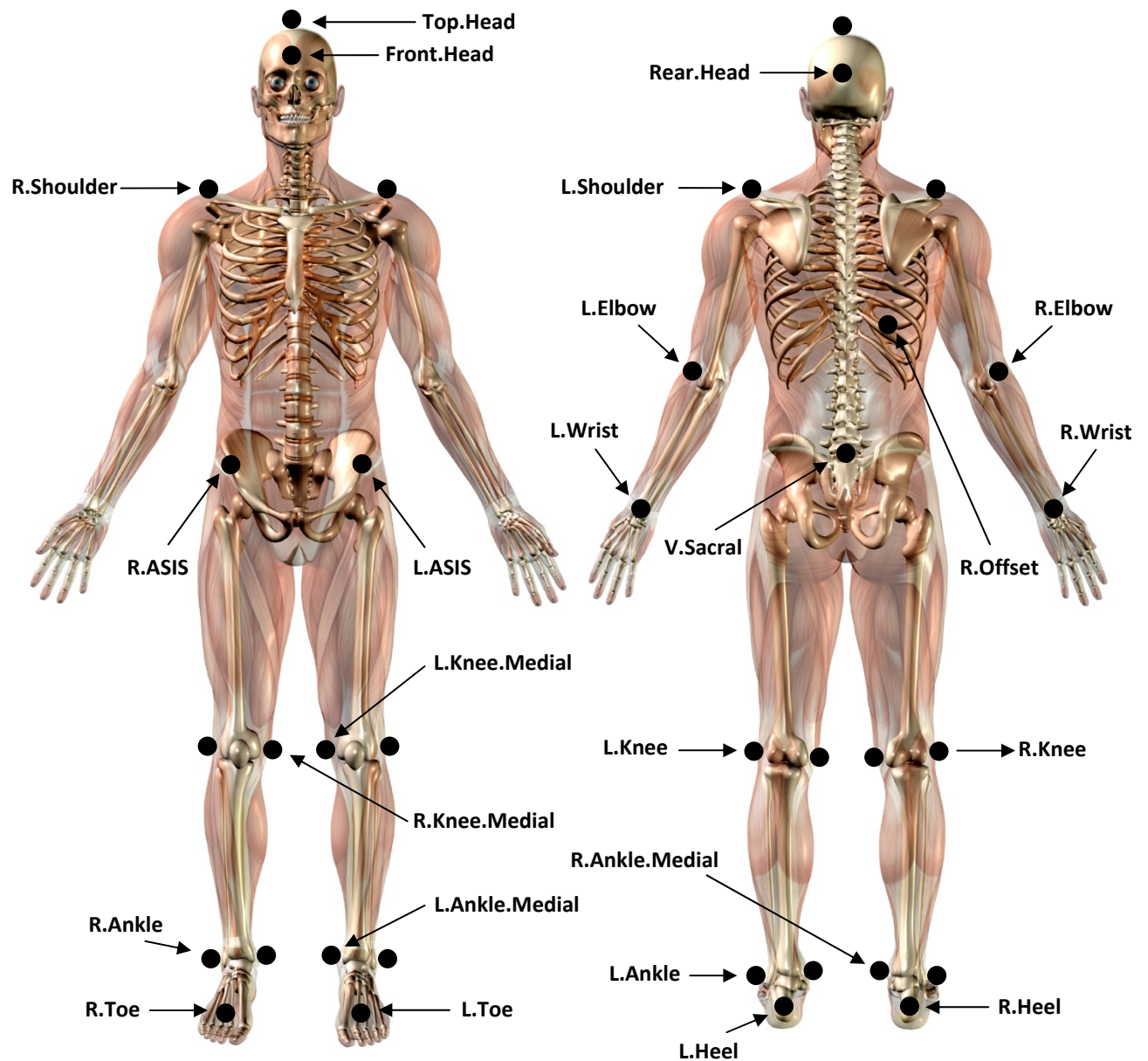
For more information, refer to the Cortex Tutorial ‘Calculating SkB Kinetics using Sky and Excel’ found under Help → Tutorials → Cortex Skeleton Builder (SkB) and Kinetics – Tutorials.

One of the attractive features of the new Sky Scripting Interface in Cortex is its ability to interact with outside programs. Because Sky is a Visual Basic based scripting language, it can not only function within Cortex but it can also seamlessly with other Windows based programs, most notably Microsoft Excel. A Sky script has been written that will calculate Kinetics for up to two cycles (i.e., gait cycle), write out a kinematics and kinetics .kin file, launch Excel, and compute and graph lower body kinematics and kinetics.

The Sky file of interest is called Kinetics_HH_Marker_Set_Graphs_Excel200x.sky. There are three separate Sky files, one each for MS Excel 2000, 2003, and 2007, and each can be found in the Tools → Sky → Global Sky Files folder. Although this Sky file is designed to be used with the sample Kinetics data, it can be used in real world applications in its current form or with modifications. After the Sky file calculates Kinetics for two separate cycles, it will export a ‘Kinematics and Kinetics’ .kin and .forces file for each cycle, and launch Excel. VBA code within Excel will bring in the .kin and .forces files and graph joint angles, joint moments, and joint powers for the ankle, knee, and hip joints. The Excel file that will graph the Kinetics data (\Cortex\UserFiles\Kinetics\Kinetics_HH_Marker_Set_Graphs.xls) can be used for real world applications also, but is mainly intended to get users started. The source code is hidden but can be accessed by emailing Motion Analysis Customer Support at support@motionanalysis.com. As the graphs continue to be updated with new features, current Kinetics users will receive a copy of the Excel file.

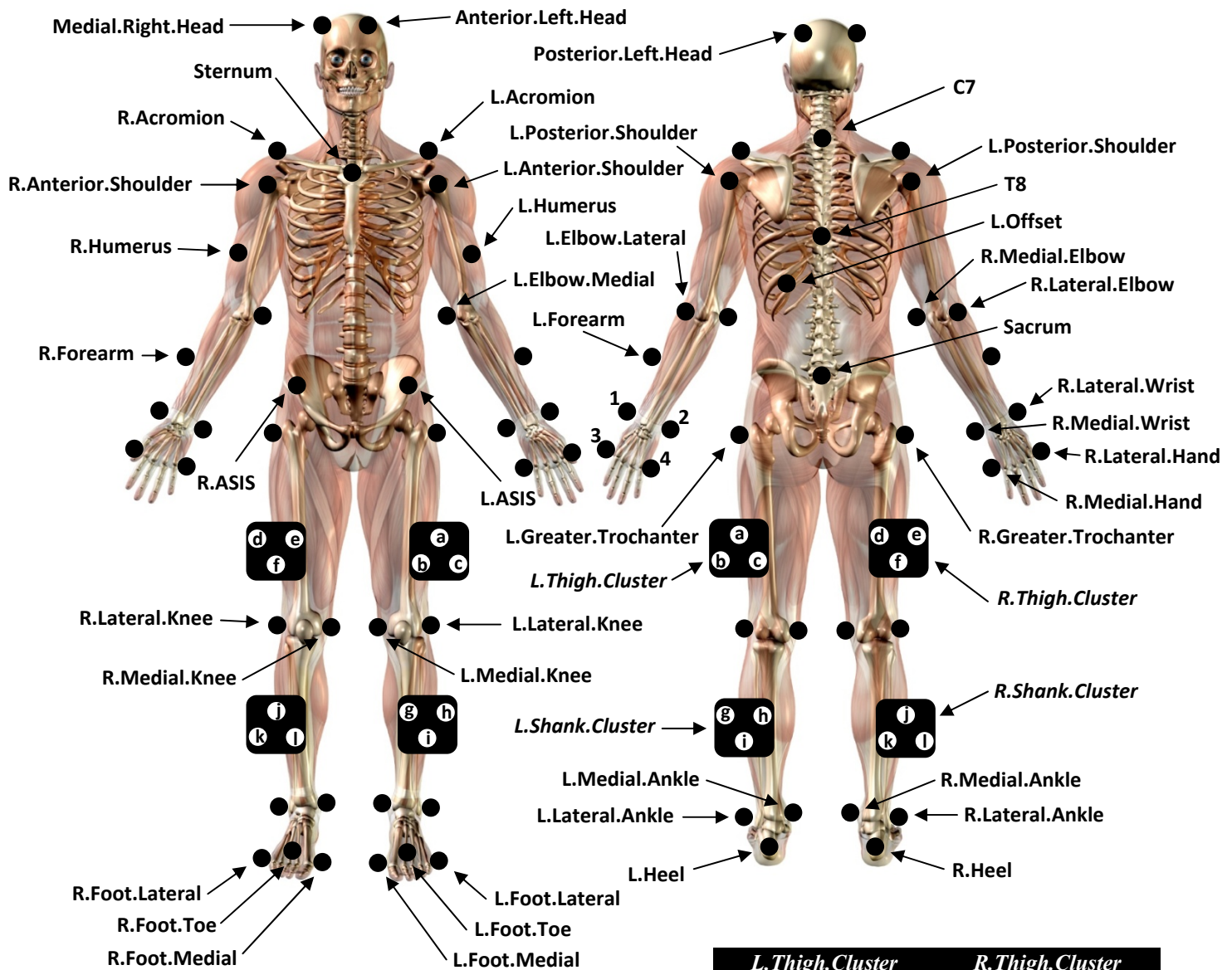
Appendix A

Helen Hayes Marker Set Placement



Note: This marker set is used in the HH Samples Kinetics Data found in \Cortex\Samples\Skeleton Builder Kinetics AND in the Helen Hayes Marker Set found in \Cortex\Samples.

Modified Cleveland Clinic Marker Set Placement

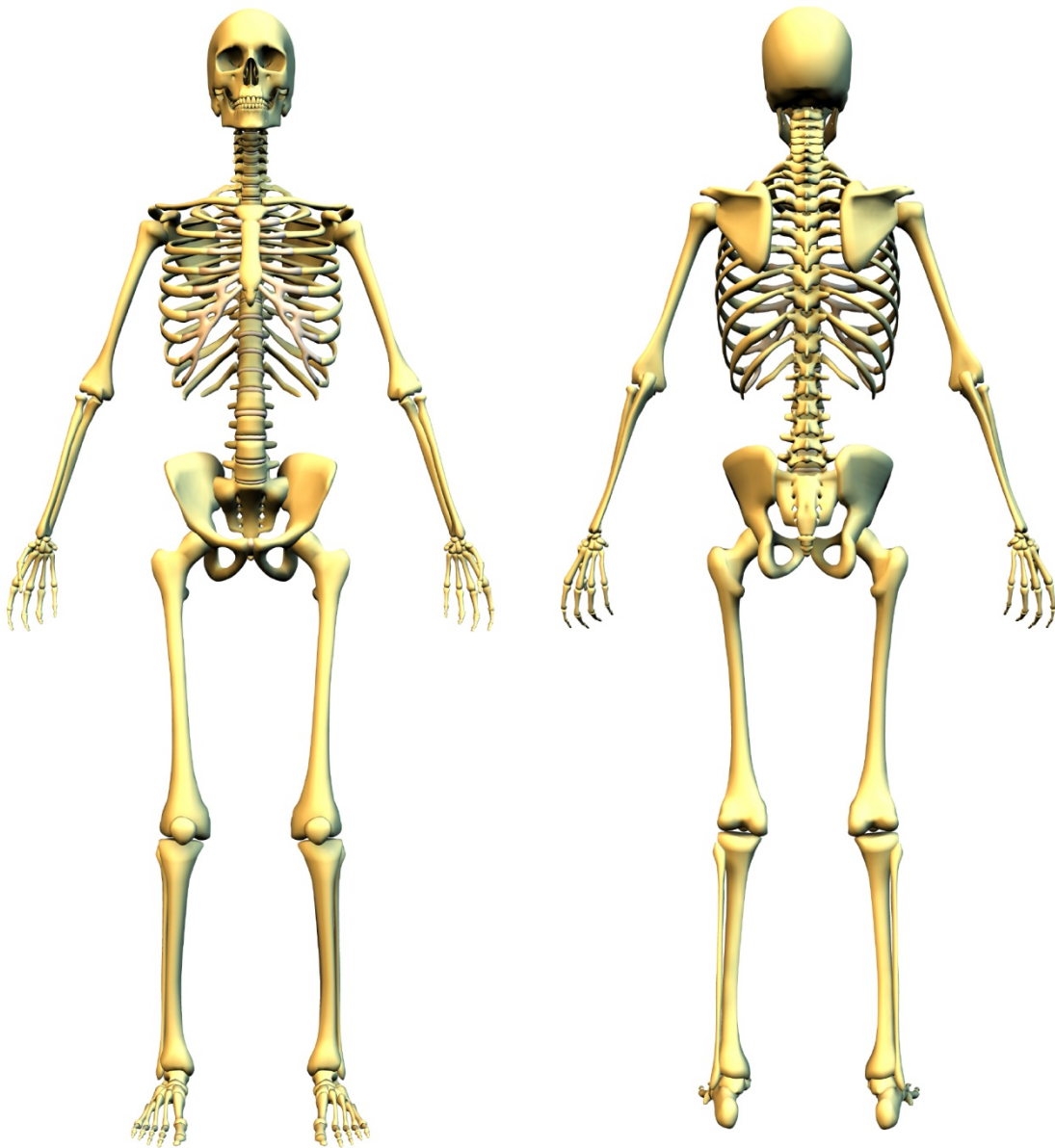


Hand Marker Abbrev.	
1	L.Lateral.Wrist
2	L.Medial.Wrist
3	L.Lateral.Hand
4	L.Medial.Hand

L. Thigh.Cluster		R. Thigh.Cluster	
a	L. Thigh.Superior	d	R. Thigh.Posterior
b	L. Thigh.Anterior	e	R. Thigh.Anterior
c	L. Thigh.Posterior	f	R. Thigh.Inferior
L. Shank.Cluster		R. Shank.Cluster	
g	L. Shank.Anterior	j	R. Shank.Superior
h	L. Shank.Posterior	k	R. Shank.Posterior
i	L. Shank.Inferior	l	R. Shank.Anterior

Note: This marker set is used in the CC Sample Kinetics Data found in Cortex\Samples\Skeleton Builder Kinetics.

Appendix B: Sample Pictures





Works Cited

1. Winter, D., *Biomechanics of Motor Control and Human Movement*. 3 ed. 2005, Hoboken, NJ: John Wiley and Sons, Inc.
2. Hamill, J. and W.S. Selbie, *Three-Dimensional Kinetics*, in *Research Methods in Biomechanics*. 2004, Human Kinetics: Champaign, IL. p. 145-160.
3. Camomilla, V., et al., *An optimized protocol for hip joint centre determination using the functional method*. J Biomech, 2006. **39**(6): p. 1096-106.
4. Leardini, A., et al., *Validation of a functional method for the estimation of hip joint centre location*. J Biomech, 1999. **32**(1): p. 99-103.
5. Schwartz, M.H. and A. Rozumalski, *A new method for estimating joint parameters from motion data*. J Biomech, 2005. **38**(1): p. 107-16.
6. Seidel, G.K., et al., *Hip joint center location from palpable bony landmarks--a cadaver study*. J Biomech, 1995. **28**(8): p. 995-8.
7. Siston, R.A. and S.L. Delp, *Evaluation of a new algorithm to determine the hip joint center*. J Biomech, 2006. **39**(1): p. 125-30.
8. Bell, A.L., R.A. Brand, and D.R. Pedersen, *Prediction of hip joint center location from external landmarks*. Hum Mvmt Sci, 1989. **8**: p. 3-16.
9. Bell, A.L., D.R. Pedersen, and R.A. Brand, *A comparison of the accuracy of several hip center location prediction methods*. J Biomech, 1990. **23**(6): p. 617-21.
10. de Leva, P., *Adjustments to Zatsiorsky-Seluyanov's segment inertia parameters*. J Biomech, 1996. **29**(9): p. 1223-1230.