

Virtual Lab Direction of Progression

What if the subject is not walking along the lab axes? What if the subject is walking at a diagonal?

To calculate certain measures, such as the trunk angle relative to the lab, the virtual lab should be aligned with the subject's direction of progression. If the subject is not aligned with the principal axes of the lab there are several methods that can be used to do this, a few are described below.

If the subject is walking along the principal axes of the lab, a [dynamic virtual lab](#) can be used instead of the following methods.

Introduction

To calculate the trunk angle relative to the lab, the following steps can be taken:

- Step 1 - Define the direction of progression
- Step 2 - Define the virtual lab
- Step 3 - Define the trunk angle relative to the virtual lab

Sample Files

Each of the following methods can be used with the [sample data files](#). When opening the CMO file, notice that the Virtual Lab segment is aligned with the principal axes of the lab. The scripts for Method 1 and Method 2 will update the alignment of the Virtual Lab segment so that the axes is aligned with the direction of progression. Method 3 calculates the joint angles manually, so it is not necessary to update the alignment of the Virtual Lab since it is not used.

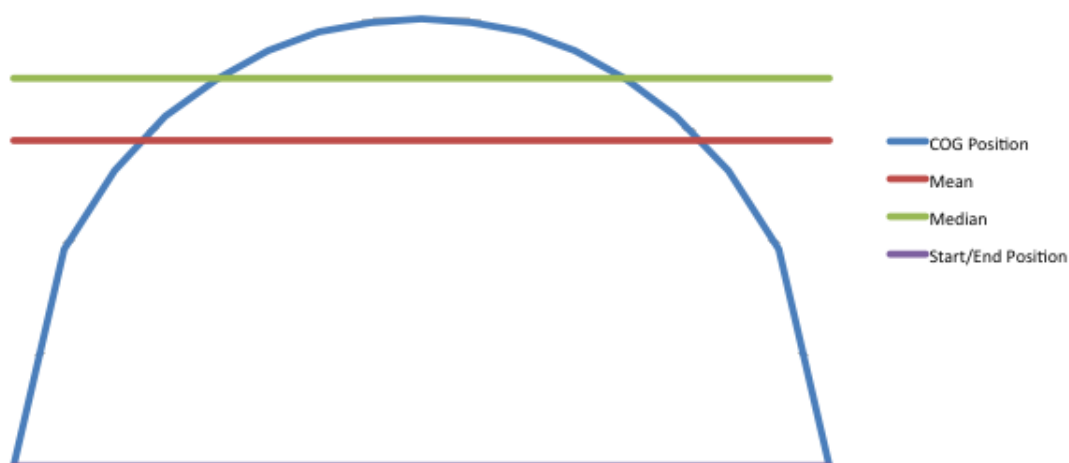
These methods are also explained in [power point form](#) for a more detailed explanation.

Metric Direction of Progression

In each of the following methods, the direction of progression will need to be defined (Step 1). Once the direction of progression has been based off some measure, the user will need to determine what "progression" they are looking for. This means that if the subject walked in an arc (blue), does the user want to:

- Calculate the average direction the subject traveled (red)
- Calculate the median direction the subject traveled (green)
- Calculate the difference between the start/stop position (purple)

Obviously, it is rare that a subject is walking in a complete circle, but depending on the task, the user should choose the method that will result in the solution they are looking for.



Method 1

An example script that demonstrates this method can be found [here](#).

Please review the [power point slides](#) for a more detailed explanation.

Step 1: Direction of Progression

1. Find the rotation between the lab and the trunk segment (Trunk_Rot)
2. Project the trunk's coordinate system onto the floor:

1. Create the Y Unit vector by setting the z component of Trunk_Rot's y component to zero, then find the magnitude of this signal
2. Create the Z Unit vector (0,0,1)
3. Create the X Unit vector by crossing the Y and Z unit vectors
3. Calculate the [metric direction of progression](#) from the X Unit vector

Step 2: Define the Virtual Lab

1. Create the virtual lab using X Unit Vector metric of the direction of progression

Step 3: Define the trunk angle relative to the virtual lab

Using Model Compute Model Based Data, calculate the Joint_Angle between the Trunk and Virtual Lab segments.

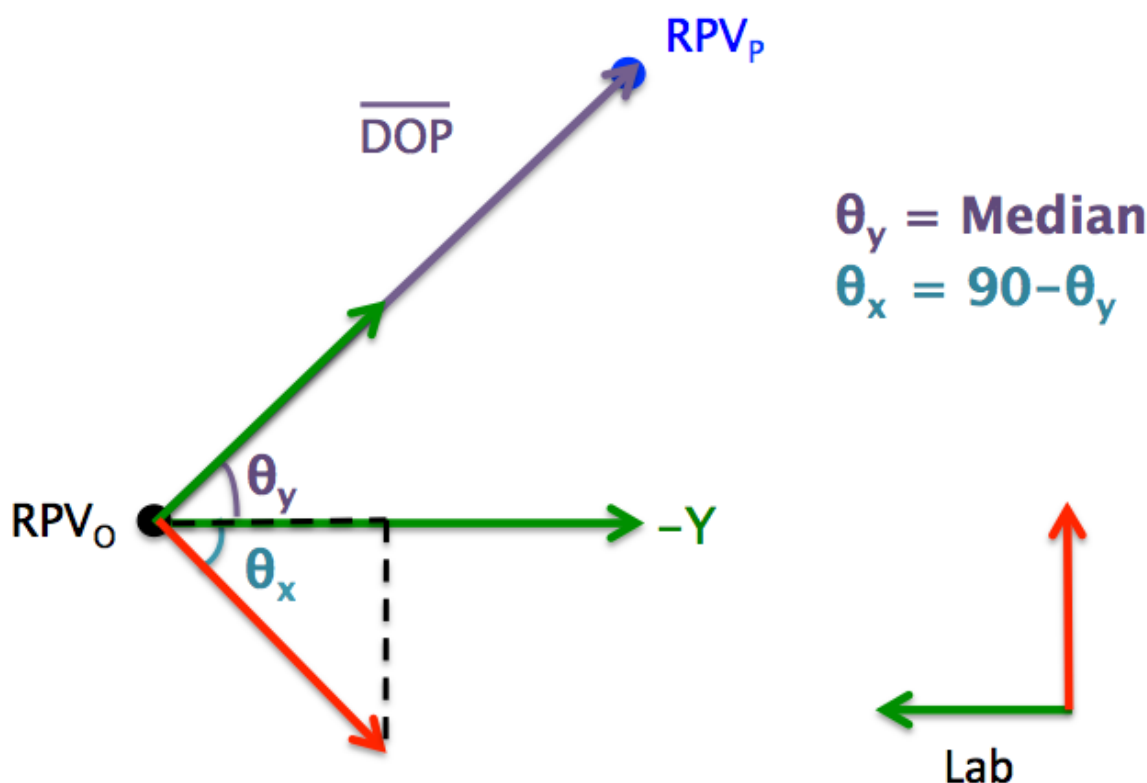
Method 2

An example script that demonstrates this method can be found [here](#).

Please review the [power point slides](#) for a more detailed explanation.

Step 1: Direction of Progression

- Get the initial position of the origin of the pelvis (RPV₀)
- Create this as a time based signal
- Create a vector from the origin of the pelvis that offset in the -Y direction of the lab (-Y)
- Project the pelvis origin onto the floor (DOP)
- Calculate the angle between the origin of the pelvis at t=0, the -Y direction of the lab, and the direction of progression (DOP) at each time point in the trial
- Calculate the **metric direction of progression** based on the angle (Theta Y) between the direction of progression and the Y Axis
- Theta X = 90 - Theta Y



Step 2: Define the Virtual Lab

1. Create landmark to define the orientation of the virtual lab (Dir_Prog_X)
 1. Offset in the X: $-0.1 * \sin((90 - \text{Theta Y}) * \pi/180)$
 2. Offset in the Y: $-0.1 * \cos((90 - \text{Theta Y}) * \pi/180)$
 3. Offset in the Z: 0
2. Create a [virtual lab](#) and use the Dir_Prog_X landmark instead of Lab_Other

Step 3: Define the trunk angle relative to the virtual lab

Using Model Compute Model Based Data, calculate the Joint_Angle between the Trunk and Virtual Lab segments.

Method 3

An example script that demonstrates this method can be found [here](#).

Please review the [power point slides](#) for a more detailed explanation.

Step 1: Direction of Progression

1. Find the location of the model's center of mass at the beginning of the trial
2. Find the location of the model's center of mass at the end of the trial
3. Find the difference between the two locations
4. Create the unit vector of the difference
 1. This result is a metric so this value can be used to define the direction of progression ([see here for more information](#))
 2. There should be no progression in the vertical direction so set Z=0

Step 2: Define the Virtual Lab

The joint angles will be solved for manually in Step 3, so it is unnecessary to create a virtual lab for this method.

Step 3: Define the trunk angle relative to the virtual lab

1. Define rotation from Lab to VLab
 1. Z component is perpendicular to the ground
 2. Y component is the direction of progression
 3. X component is Y cross Z
2. Define rotation from Lab to Trunk
 1. Using Compute Model Based Data command select the Joint_Rotation function and resolve the Thorax/Ab rotation in the Lab coordinate system
3. Solve for the rotation between VLab and the trunk segment
4. Solve for the joint angles:

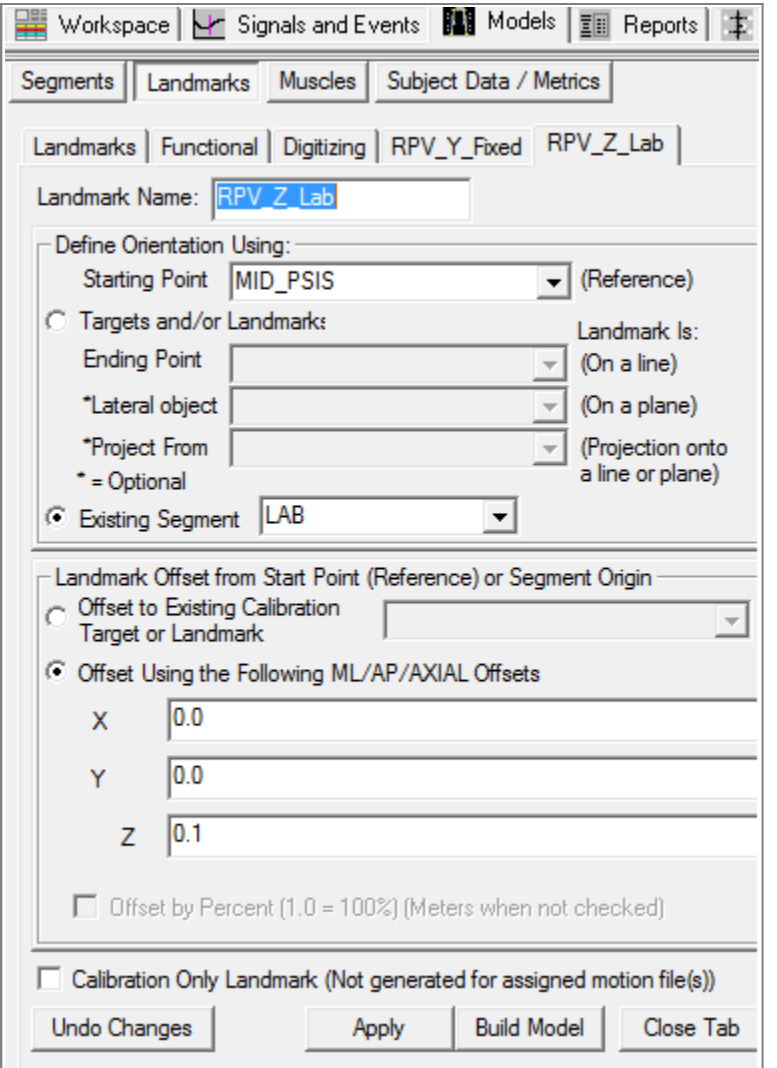
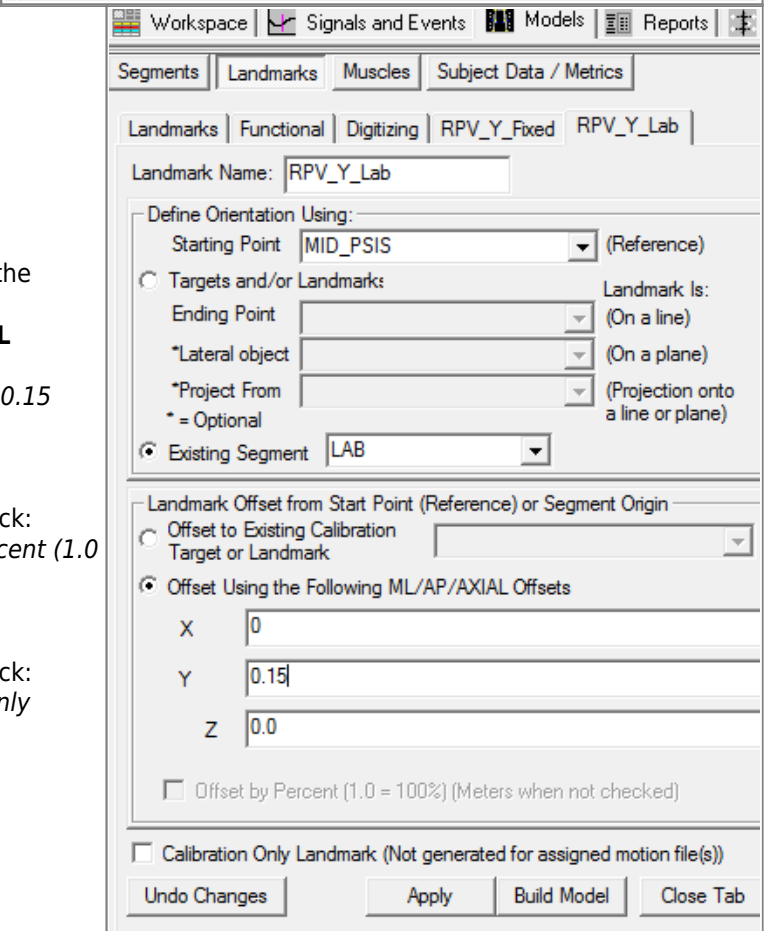
1. Solve for Theta X
2. Solve for Theta Y
3. Solve for Theta Z

Method 4

This method creates a Virtual Lab segment that follows the general orientation of the Pelvis (or any segment) but keeps the transverse plane parallel to the lab floor. In other word, this method constrains the Virtual Lab to only rotations around the vertical axis based on the Pelvis segment. This method thus allows to compute variables using the instant direction of progression.

Create Landmarks

1. Create MID_P SIS: • Click Landmarks button • Click Add New Landmark button • Create Landmark: MID_P SIS	Landmark Name: MID_P SIS Define Orientation Using: Starting Point: RPSIS Ending Point: LPSIS	• Offset Using the Following ML/AP/AXIAL Offsets: ML: 0.0 AP: 0.0 AXIAL: 0.5 • Check: Offset by Percent (1.0 = 100%) • DO NOT Check: Calibration Only Landmark	SegmentsLandmarksMusclesSubject Data / Metrics LandmarksFunctionalDigitizingRPV_Y_FixedMID_P SIS Landmark Name: MID_P SIS Define Orientation Using: Starting Point: RPSIS (Reference) Targets and/or Landmarks: Ending Point: LPSIS (On a line) *Lateral object (On a plane) *Project From (Projection onto a line or plane) * = Optional Existing Segment Landmark Offset from Start Point (Reference) or Segment Origin: Offset to Existing Calibration Target or Landmark Offset Using the Following ML/AP/AXIAL Offsets ML AP AXIAL 0.5 Offset by Percent (1.0 = 100%) (Meters when not checked) Calibration Only Landmark (Not generated for assigned motion file(s)) Undo ChangesApplyBuild ModelClose Tab
---	--	---	---

1. Create RPV_Z_Lab: - Click Landmarks button - Click Add New Landmark button - Create Landmark: <i>RPV_Z_Lab</i>	Landmark Name: <i>RPV_Z_Lab</i> Define Orientation Using: Starting Point: <i>MID_P SIS</i> Existing Segment: <i>LAB</i>	- Offset Using the Following ML/AP/AXIAL Offsets: X: <i>0.0</i> Y: <i>0.0</i> Z: <i>0.1</i> DO NOT Check: <i>Offset by Percent (1.0 = 100%)</i> DO NOT Check: <i>Calibration Only Landmark</i>	
1. Create RPV_Y_Lab: - Click Landmarks button - Click Add New Landmark button - Create Landmark: <i>RPV_Y_Lab</i>	Landmark Name: <i>RPV_Y_Lab</i> Define Orientation Using: Starting Point: <i>MID_P SIS</i> Existing Segment: <i>LAB</i>	- Offset Using the Following ML/AP/AXIAL Offsets: X: <i>0.0</i> Y: <i>0.15</i> Z: <i>0.0</i> DO NOT Check: <i>Offset by Percent (1.0 = 100%)</i> DO NOT Check: <i>Calibration Only Landmark</i>	

1. Create RPV_Y: • Click Landmarks button • Click Add New Landmark button • Create Landmark: RPV_Y	Landmark Name: RPV_Y Define Orientation Using: Starting Point: MID_P SIS Existing Segment: Pelvis	• Offset Using the Following ML/AP/AXIAL Offsets: ML: 0.0 AP: 0.15 AXIAL: 0.0 • DO NOT Check: Offset by Percent (1.0 = 100%) • DO NOT Check: Calibration Only Landmark	Workspace Signals and Events Models Reports Segments Landmarks Muscles Subject Data / Metrics Landmarks Functional Digitizing RPV_Y_Fixed RPV_Y Landmark Name: RPV_Y Define Orientation Using: Starting Point MID_P SIS (Reference) Targets and/or Landmarks: Ending Point (On a line) *Lateral object (On a plane) *Project From (Projection onto a line or plane) * = Optional Existing Segment Pelvis Landmark Offset from Start Point (Reference) or Segment Origin Offset to Existing Calibration Target or Landmark Offset Using the Following ML/AP/AXIAL Offsets ML 0.0 AP 0.15 AXIAL 0 Offset by Percent (1.0 = 100%) (Meters when not checked) Calibration Only Landmark (Not generated for assigned motion file(s)) Undo Changes Apply Build Model Close Tab
1. Create RPV_Y_PROJ: • Click Landmarks button • Click Add New Landmark button • Create Landmark: RPV_Y_PROJ	Landmark Name: RPV_Y_PROJ Define Orientation Using: Starting Point: MID_P SIS Ending Point: RPV_Z_Lab Lateral object: RPV_Y Project From: RPV_Y_Lab	• Offset Using the Following ML/AP/AXIAL Offsets: X: 0.0 Y: 0.0 Z: 0.0 • DO NOT Check: Offset by Percent (1.0 = 100%) • DO NOT Check: Calibration Only Landmark	Workspace Signals and Events Models Reports Segments Landmarks Muscles Subject Data / Metrics Landmarks Functional Digitizing RPV_Y_Fixed RPV_Y_PROJ Landmark Name: RPV_Y_PROJ Define Orientation Using: Starting Point MID_P SIS (Reference) Targets and/or Landmarks: Ending Point (On a line) *Lateral object (On a plane) *Project From (Projection onto a line or plane) * = Optional Existing Segment Landmark Offset from Start Point (Reference) or Segment Origin Offset to Existing Calibration Target or Landmark Offset Using the Following ML/AP/AXIAL Offsets ML AP AXIAL Offset by Percent (1.0 = 100%) (Meters when not checked) Calibration Only Landmark (Not generated for assigned motion file(s)) Undo Changes Apply Build Model Close Tab

1. Create RPV_Y_Fixed: - Click Landmarks button - Click Add New Landmark button - Create Landmark: <i>RPV_Y_Fixed</i>	Landmark Name: <i>RPV_Y_Fixed</i> Define Orientation Using: Starting Point: <i>MID_P SIS</i> Ending Point: <i>RPV_Y_PROJ</i>	- Offset Using the Following ML/AP/AXIAL Offsets: ML: 0.0 AP: 0.0 AXIAL: 0.1 DO NOT Check: <i>Offset by Percent (1.0 = 100%)</i> DO NOT Check: <i>Calibration Only Landmark</i>	
---	--	---	--

Create Virtual Lab Segment

To create the Virtual Laboratory segment, switch to the **Segment** Tab in model builder mode.

1. Create the Virtual Lab Segment: - In the Segment Name combo box type Virtual Lab - Check the Kinematic Only Check Box - Select the Create button	
--	--

4. In the Virtual Lab tab, enter these values:	Define Proximal Joint and Radius Lateral:NoneJoint:MID_P SISMedial:None Radius: 0.01 Define Distal Joint and Radius Lateral:NoneJoint:RPV_Z_LabMedial:None Radius: 0.01 Extra Target to define Orientation Select:LateralRPV_Y_Fixed Select Tracking Targets: Check: Use Calibration Targets for Tracking	5. Click on Build Model . 6. Click on Close Tab before proceeding.	WorkspaceSignals and EventsModelsReports SegmentsLandmarksMusclesSubject Data / Metrics SegmentsSegment PropertiesIK ConstraintsVirtual Lab Define Proximal Joint and Radius LateralJoint CenterMedial NoneMID_P SISNone Radius (Meters)0.01 Define Distal Joint and Radius LateralJoint CenterMedial NoneRPV_Z_LabNone Radius (Meters)0.01 Extra Target to Define Orientation (if needed) LocationLateralRPV_Y_Fixed Select Tracking Targets (Ctrl-Left Mouse Click to Multiselect) Use Calibration Targets for Tracking Depth (Meters): Close TabGuess PropertiesApplyBuild Model
---	---	---	---