HRI Lab #7, SP2024 525.786: Human Robotics Interaction

Introduction:

In this lab we will explore interfacing various types of motion tracking tools. These tools, in combination with the bioinstrumentation inputs you've used previously, can be used for natural and intuitive Human Robotic Interaction. The tools that we'll be using include the following devices:

```
- Inertial measurement Unit (3-axis, 6 Degrees of Freedom) on the Myo
- Kinect (20 Degrees of Freedom, Whole Body)
```

Setup the MiniVIE environment

```
thisPath = cd;
cd('C:\GitHub\MiniVIE');
MiniVIE.configurePath;
cd(thisPath);
```

Part 1: Inertial Measurement Unit on the Myo

Operating Principle: the Myo armband contains an Angular rate sensor, accelerometer, and magnetometer to resolve orientation. These sensors can be used to track motion of the arm in addition to the EMG signals

Before running the code below, be sure to start MyoUdp.exe in the MiniVIE/+Inputs folder

It might be more straightforward to run this in the Command Window (select, then hit F9)

```
% Setup the Myo Armband in matlab
hMyo = Inputs.MyoUdp.getInstance();
hMyo.initialize()
pause(1)

% Ensure data is streaming
myoData = hMyo.getData();
if isempty(hMyo.Orientation)
    error('Setup Failed.  MyoUdp Running?');
end

% Setup figure to show orientation
figure(1);
clf
daspect([1 1 1])
view(-170,15)

% Plot reference triad
PlotUtils.triad(eye(4),0.5)

% Plot 'moving' triad
myoTriadHandle = PlotUtils.triad(eye(4),2);
```

```
StartStopForm([])
while StartStopForm()
    drawnow
    hMyo.getData();  % update stream receive
    Rxyz = hMyo.getEulerAngles; % Get roll pitch yaw
    disp(Rxyz)
    Rmat = LinAlg.makeRotationMtx(Rxyz); % create rotation matrix
    transMat = [Rmat [0; 0; 0]; 0 0 0 1];
    set(myoTriadHandle,'Matrix', transMat);
end
hMyo.close()

cleanup;
```

Questions:

1. What is the 'global' position of the myo armband?
2. What is its local coordinate frame?
3. What processing would you do to use the myo armband as a sensor to control the elbow flexion angle of a robotic device?

Use the Myo Orientation Sensors to Plot Arm Position

Edit the code below to show (in a streaming fashion) the real-time orientation and position of the forearm.

It might be more straightforward to run this in the Command Window (select, then hit F9)

```
hMyo = Inputs.MyoUdp.getInstance();
hMyo.initialize()
pause(1)

% Ensure data is streaming
myoData = hMyo.getData();
if isempty(hMyo.Orientation)
    error('Setup Failed.  MyoUdp Running?');
end

% Can you create a real time plot of your hand's position in space using
% the real-time orientation matrix from the Myo?  assume the distance from
% Myo to hand is 25 cm.
myoOrigin = [0, 0, 0];
handPosStart = [0.25, 0, 0]; % in meters
figHandle = figure(1);
clf
daspect([1 1 1])
view(-170,15)
StartStopForm([])
while StartStopForm()
```

```
    drawnow;

    % put your code here for grabbing the rotation matrix

    % put your code here for calculating a new handPos based on the
    % rotation matrix and handPosStart

    plot3([0, handPos(1)], [0, handPos(2)], [0, handPos(3)], '.-');
    set(gca, 'XLim', [-0.25 0.25], 'YLim', [-0.25 0.25], 'ZLim', [-0.25 0.25]);
    xlabel('x'); ylabel('y'); zlabel('z');

end
hMyo.close()
cleanup;
```

Part 2: Kinect

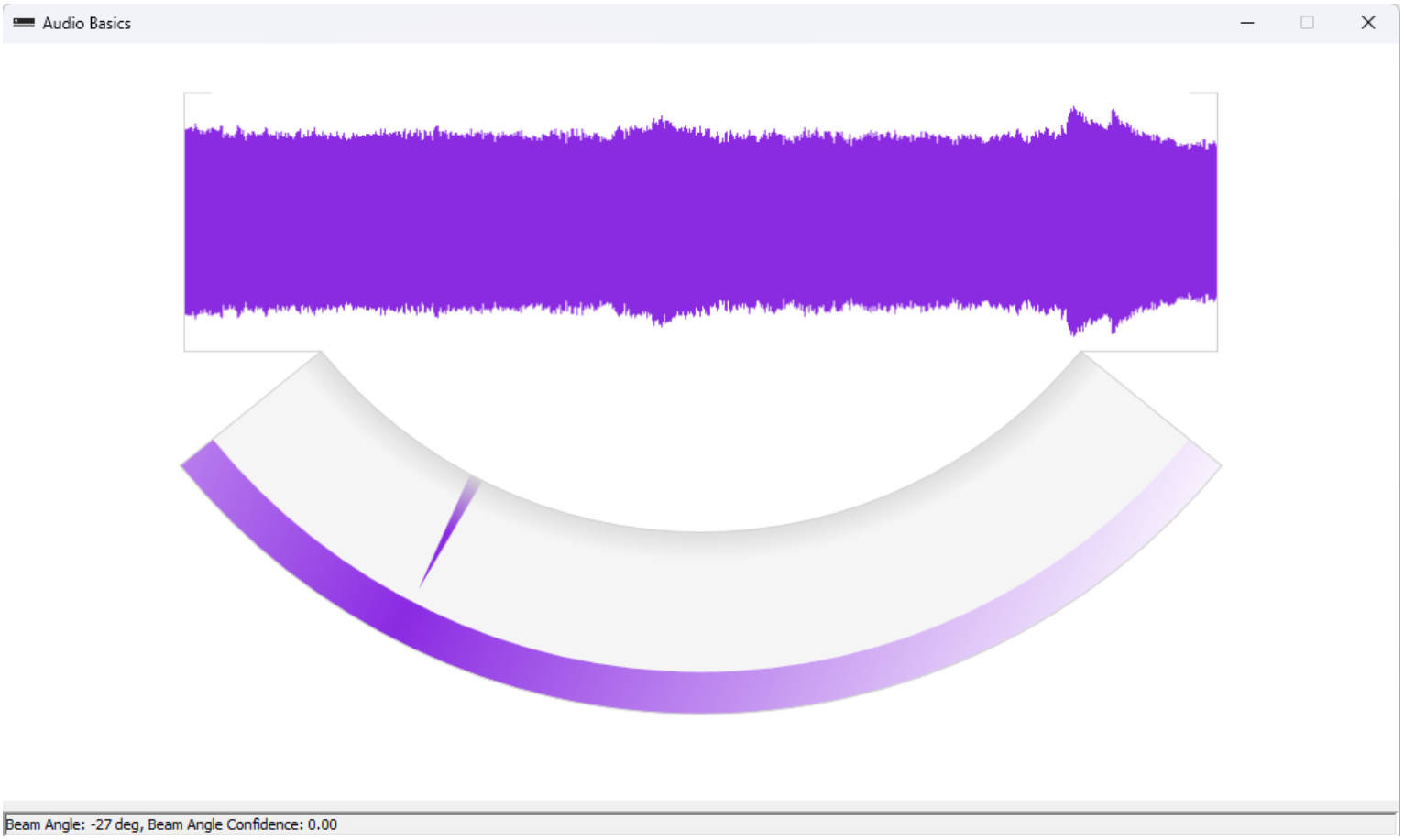
Open "SDK Browser for Kinect" (search in Windows Start Menu)

Run the following examples:

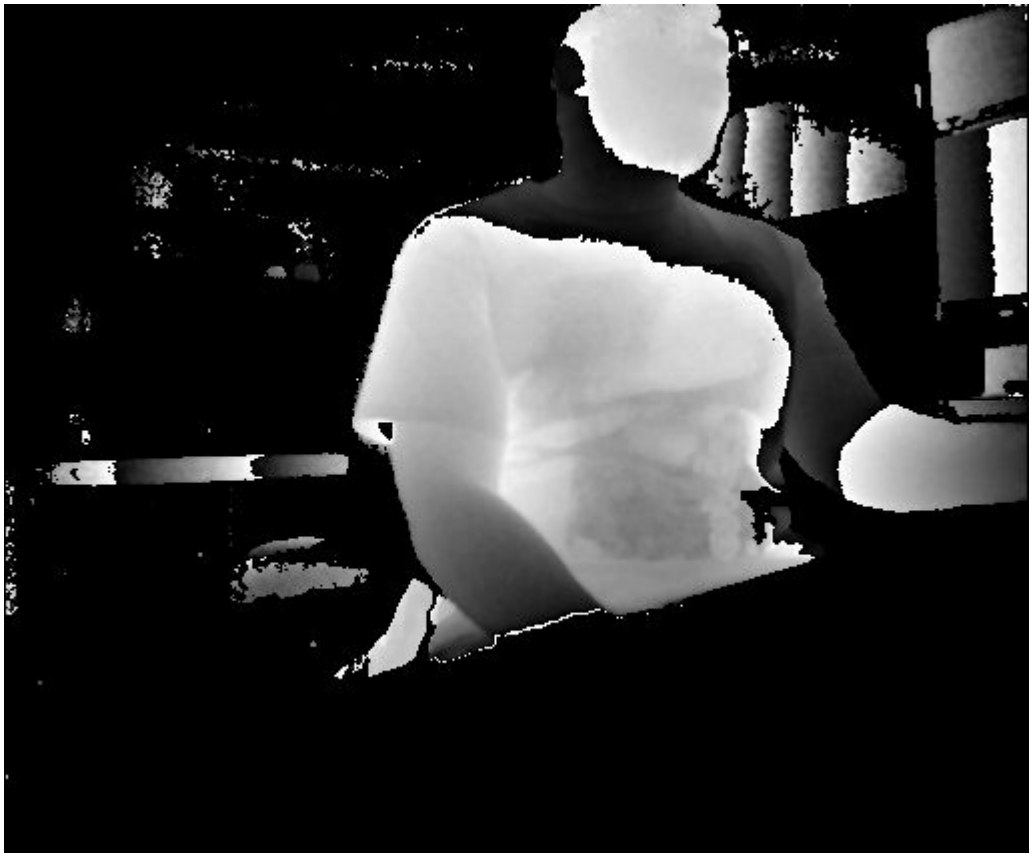
1. Audio Basics
2. Depth Basics
3. Face Tracking Basics
4. Skeletal Basics

Copy and paste window screenshots from each:

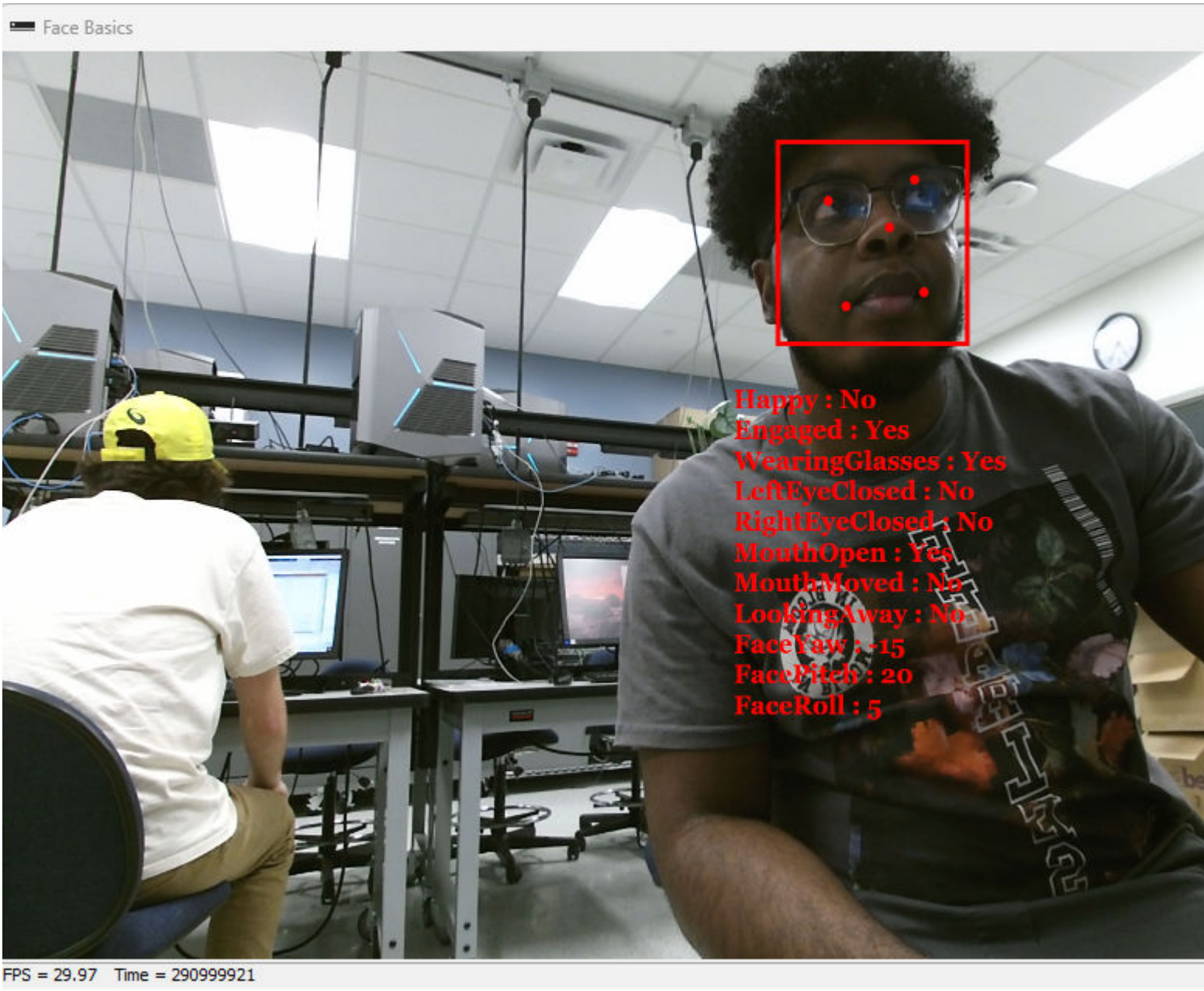
1. **Audio Basic**



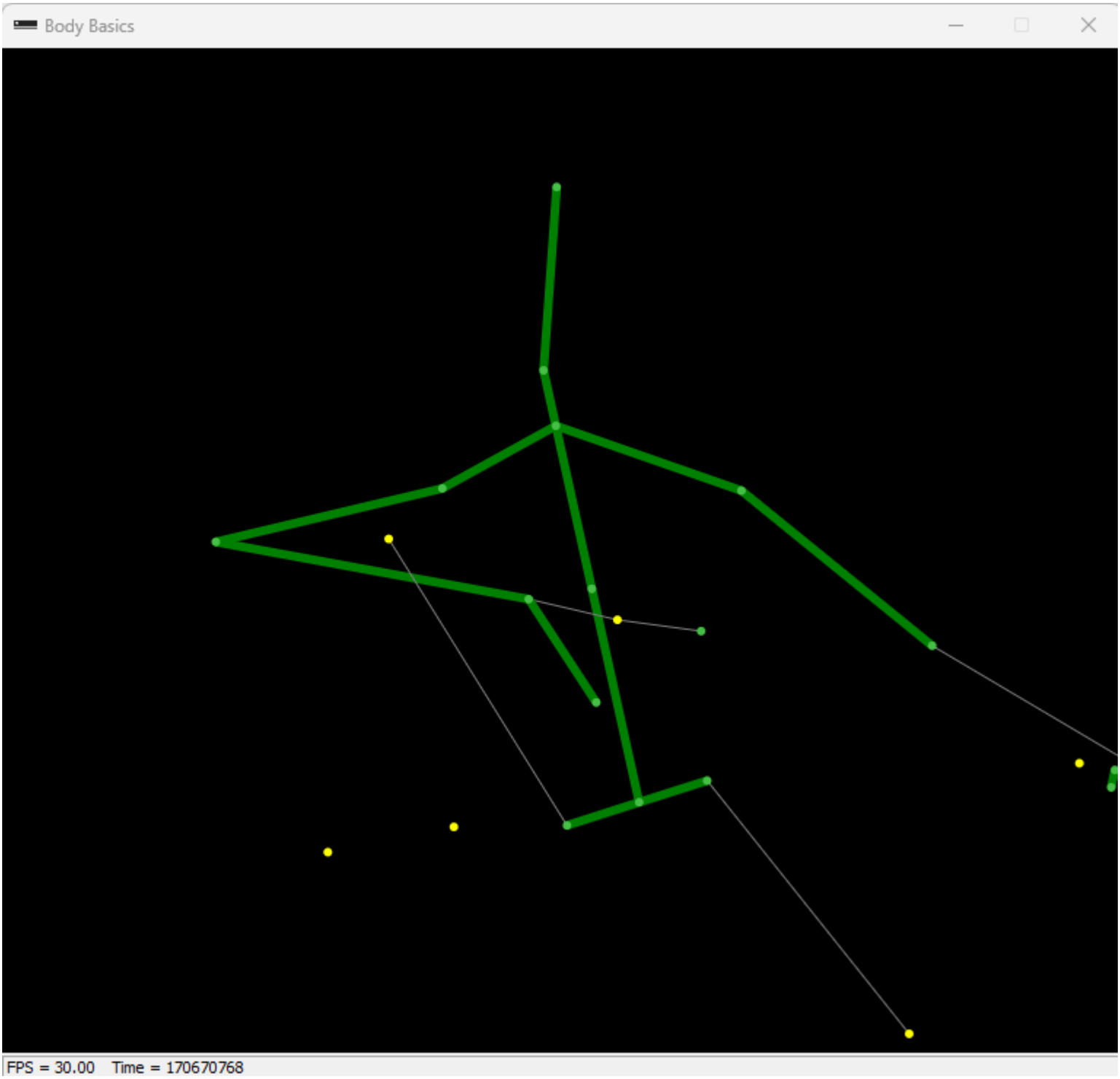
2. Depth Basic



3. Face Tracking Basic



4. Skeletal Basic



Questions:

- What advantage do microphone arrays have over traditional microphones / speech input devices?
  Microphone arrarys allow for directional audio, while a standard microphone is unidirectional. This adds audio depth to audio recorded by the microphone arrary.

- At what range is the face tracking effective? How might this be used as a robot controller? **Face tracking is effective starting at maybe a foot away. Since this tracks the user's eyes and mouth, different controls can be associated with different facial movements. This can be usful in cases where a person's body is not mobal.**

- What kind of motions and postures does the skeletal model capture? When does it break down? **The kinect captures full body, tracking arms, torso, legs and head. Tracking breaks down if the user is too close or if the connect is looking up at the user. Looking up at the user can cause leg tracking issues.**

Kinect sampling from MATLAB

derived from: https://www.mathworks.com/help/imaq/examples/using-the-kinect-r-for-windows-r-from-image-acquisition-toolbox-tm.html

required MATLAB Support Package for Kinect: https://www.mathworks.com/hardware-support/kinect-windows.html

It might be more straightforward to run this in the Command Window (select, then hit F9)

```
utilpath =
'C:\ProgramData\MATLAB\SupportPackages\R2019b\toolbox\imaq\supportpackages\kinectruntime\kinectforwindowsruntimeexamples';
addpath(utilpath);

% The Kinect for Windows Sensor shows up as two separate devices in IMAQHWINFO.
hwInfo = imaqhwinfo('kinect')
```

```
hwInfo = struct with fields:
       AdaptorDllName: 'C:\ProgramData\MATLAB\SupportPackages\R2023a\toolbox\imaq\supportpackages\kinectruntime\adap
    AdaptorDllVersion: '6.7.1 (R2023a)'
          AdaptorName: 'kinect'
            DeviceIDs: {[1]  [2]}
           DeviceInfo: [1×2 struct]
```

```
hwInfo.DeviceInfo(1)
```

```
ans = struct with fields:
              DefaultFormat: 'BGR_1920x1080'
       DeviceFileSupported: 0
                DeviceName: 'Kinect V2 Color Sensor'
                  DeviceID: 1
     VideoInputConstructor: 'videoinput('kinect', 1)'
    VideoDeviceConstructor: 'imaq.VideoDevice('kinect', 1)'
          SupportedFormats: {'BGR_1920x1080'}
```

```
hwInfo.DeviceInfo(2)
```

```
ans = struct with fields:
              DefaultFormat: 'Depth_512x424'
       DeviceFileSupported: 0
                DeviceName: 'Kinect V2 Depth Sensor'
                  DeviceID: 2
```

```
     VideoInputConstructor: 'videoinput('kinect', 2)'
    VideoDeviceConstructor: 'imaq.VideoDevice('kinect', 2)'
          SupportedFormats: {'Depth_512x424'}
```

```
% Create the VIDEOINPUT objects for the two streams
colorVid = videoinput('kinect',1)
```

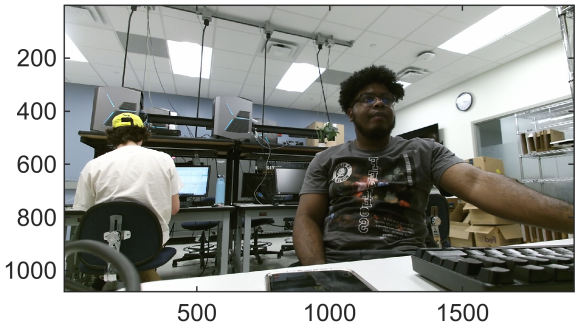
```
Summary of Video Input Object Using 'Kinect V2 Color Sensor'.

   Acquisition Source(s):  Kinect V2 Color Source is available.

  Acquisition Parameters:  'Kinect V2 Color Source' is the current selected source.
                           10 frames per trigger using the selected source.
                           'BGR_1920x1080' video data to be logged upon START.
                           Grabbing first of every 1 frame(s).
                           Log data to 'memory' on trigger.

      Trigger Parameters:  1 'immediate' trigger(s) on START.

                  Status:  Waiting for START.
                           0 frames acquired since starting.
                           0 frames available for GETDATA.
```

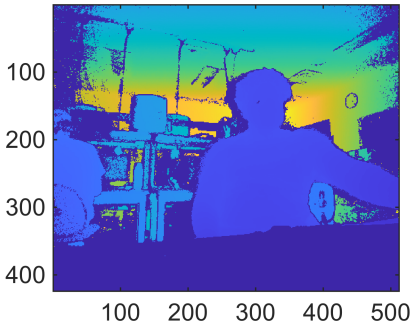
```
depthVid = videoinput('kinect',2)
```

```
Summary of Video Input Object Using 'Kinect V2 Depth Sensor'.

   Acquisition Source(s):  Kinect V2 Depth Source is available.

  Acquisition Parameters:  'Kinect V2 Depth Source' is the current selected source.
                           10 frames per trigger using the selected source.
                           'Depth_512x424' video data to be logged upon START.
                           Grabbing first of every 1 frame(s).
                           Log data to 'memory' on trigger.

      Trigger Parameters:  1 'immediate' trigger(s) on START.

                  Status:  Waiting for START.
                           0 frames acquired since starting.
                           0 frames available for GETDATA.
```

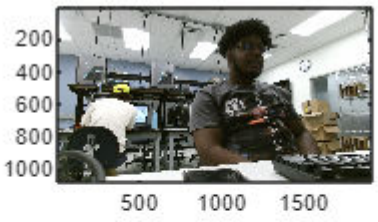
```
% Set the triggering mode to 'manual'
triggerconfig([colorVid depthVid],'manual');
colorVid.FramesPerTrigger = 100;
depthVid.FramesPerTrigger = 100;

% Start the color and depth device. This begins acquisition, but does not
% start logging of acquired data.
start([colorVid depthVid]);

% Trigger the devices to start logging of data.
trigger([colorVid depthVid]);

% Retrieve the acquired data
[colorFrameData,colorTimeData,colorMetaData] = getdata(colorVid);
[depthFrameData,depthTimeData,depthMetaData] = getdata(depthVid);
```

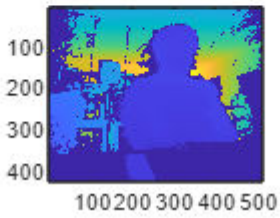
```
% Stop the devices
stop([colorVid depthVid]);

% display one of the frames
figure(1);
subplot(2, 1, 1);
imagesc(colorFrameData(:, :, :, 50));axis equal tight;
subplot(2, 1, 2);
imagesc(depthFrameData(:, :, :, 50)); axis equal tight;
```





Copy and the screenshots of your RGB and depth maps:





Skeletal tracking

Grab a frame including a person's body (i.e., "skeleton" in Kinect terms)

```
depthSrc = getselectedsource(depthVid)
```

```
depthSrc =
   Display Summary for Video Source Object:

      General Settings:
        Parent = [1x1 videoinput]
        Selected = on
        SourceName = Kinect V2 Depth Source
        Tag = [0x0 string]
        Type = videosource

      Device Specific Properties:
        EnableBodyTracking = on
```

```
% Turn on skeletal tracking.
depthSrc.EnableBodyTracking = 'on';

% Acquire 100 frames with tracking turned on.
% Remember to have a person in person in front of the
% Kinect for Windows to see valid tracking data.
colorVid.FramesPerTrigger = 100;
depthVid.FramesPerTrigger = 100;

start([colorVid depthVid]);
trigger([colorVid depthVid]);

% Retrieve the frames and check if any Skeletons are tracked
[frameDataColor] = getdata(colorVid);
[frameDataDepth, timeDataDepth, metaDataDepth] = getdata(depthVid);

% View skeletal data from depth metadata
```

```
metaDataDepth
```

metaDataDepth = 100×1 struct

...

Fields	AbsTime	BodyIndexFrame	BodyTrackingID	ColorJointIndices
1	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
2	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
3	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
4	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
5	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
6	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
7	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
8	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
9	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
10	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
11	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
12	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
13	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
14	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
15	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
16	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
17	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
18	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
19	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
20	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
21	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
22	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
23	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
24	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
25	[2025,3,24,19,50,18...	424×512 double	[720580000000000...	25×2×6 double
26	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
27	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
28	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
29	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
30	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
31	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double

Fields	AbsTime	BodyIndexFrame	BodyTrackingID	ColorJointIndices
32	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
33	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
34	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
35	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
36	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
37	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
38	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
39	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
40	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
41	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
42	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
43	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
44	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
45	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
46	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
47	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
48	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
49	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
50	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
51	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
52	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
53	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
54	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
55	[2025,3,24,19,50,19...	424×512 double	[720580000000000...	25×2×6 double
56	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
57	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
58	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
59	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
60	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
61	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
62	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
63	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
64	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double

Fields	AbsTime	BodyIndexFrame	BodyTrackingID	ColorJointIndices
65	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
66	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
67	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
68	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
69	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
70	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
71	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
72	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
73	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
74	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
75	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
76	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
77	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
78	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
79	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
80	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
81	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
82	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
83	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
84	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
85	[2025,3,24,19,50,20...	424×512 double	[720580000000000...	25×2×6 double
86	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double
87	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double
88	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double
89	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double
90	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double
91	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double
92	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double
93	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double
94	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double
95	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double
96	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double
97	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double

Fields	AbsTime	BodyIndexFrame	BodyTrackingID	ColorJointIndices
98	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double
99	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double
100	[2025,3,24,19,50,21...	424×512 double	[720580000000000...	25×2×6 double

```
% Check for tracked skeletons from depth metadata
anyBodiesTracked = any(metaDataDepth(95).IsBodyTracked ~= 0)
```

```
anyBodiesTracked = logical
   1
```

```
% See which skeletons were tracked.
trackedBodies = find(metaDataDepth(95).IsBodyTracked)
```

```
trackedBodies = 1
```

```
jointCoordinates = metaDataDepth(95).JointPositions(:, :, trackedBodies)
```

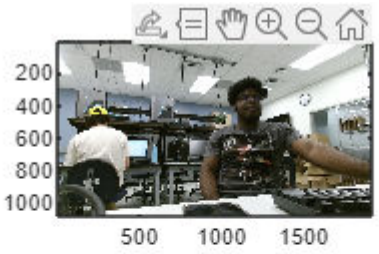
```
jointCoordinates = 25×3
    0.0780   -0.5827    1.3692
    0.0805   -0.2876    1.4040
    0.0816   -0.0014    1.4224
    0.0911    0.1579    1.4083
   -0.1140   -0.0854    1.3964
   -0.3362   -0.0806    1.2848
   -0.4303    0.0710    1.1144
   -0.4509    0.1336    1.0826
    0.2839   -0.0892    1.4164
    0.4828   -0.1047    1.3156
      ⋮
```

```
% Skeleton's joint indices with respect to the color image
jointIndices = metaDataDepth(95).ColorJointIndices(:, :, trackedBodies)
```

```
jointIndices = 25×2
10^3 ×
    1.0629    1.0112
    1.0610    0.7698
    1.0593    0.5481
    1.0670    0.4251
    0.9091    0.6141
    0.7167    0.6168
    0.5872    0.4803
    0.5552    0.4156
    1.2160    0.6148
    1.4016    0.6321
      ⋮
```

```
% Pull out the 95th color frame
image = frameDataColor(:, :, :, 95);
```

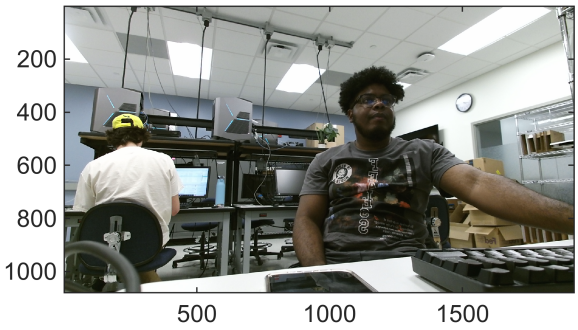
```
% Find number of Skeletons tracked
nBodies = length(trackedBodies);
```

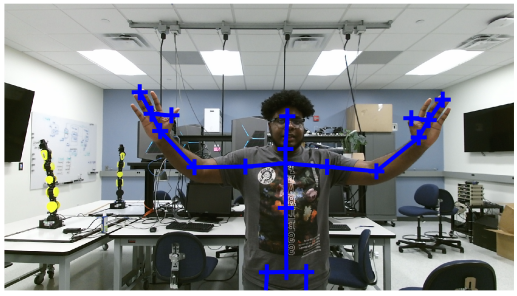


Now plot the skeleton over the RGB image

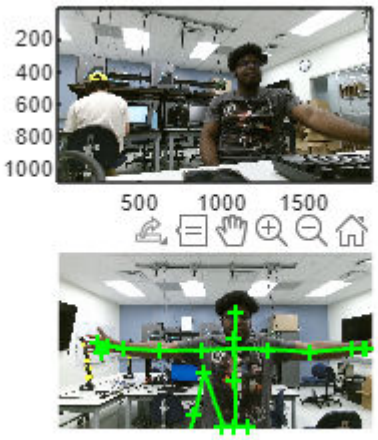
```
% Create skeleton connection map to link the joints.
SkeletonConnectionMap = [ [4 3];  % Neck
                          [3 21]; % Head
                          [21 2]; % Right Leg
                          [2 1];
                          [21 9];
                          [9 10];  % Hip
                          [10 11];
                          [11 12]; % Left Leg
                          [12 24];
                          [12 25];
                          [21 5];  % Spine
                          [5 6];
                          [6 7];   % Left Hand
                          [7 8];
                          [8 22];
                          [8 23];
                          [1 17];
                          [17 18];
                          [18 19];  % Right Hand
                          [19 20];
                          [1 13];
                          [13 14];
                          [14 15];
                          [15 16];
                        ];

% Marker colors for up to 6 bodies.
colors = ['b';'r';'g';'c';'y';'m'];
```

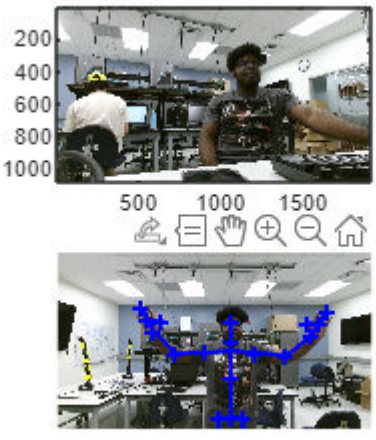
```
% Display the RGB image.
imshow(image);

% Overlay the skeleton on this RGB frame.
for i = 1:24
     for body = 1:nBodies
         X1 = [jointIndices(SkeletonConnectionMap(i,1),1,body)
jointIndices(SkeletonConnectionMap(i,2),1,body)];
         Y1 = [jointIndices(SkeletonConnectionMap(i,1),2,body)
jointIndices(SkeletonConnectionMap(i,2),2,body)];
         line(X1,Y1, 'LineWidth', 1.5, 'LineStyle', '-', 'Marker', '+', 'Color',
colors(body));
     end

    hold on;
 end
 hold off;
```





Copy and paste a screenshot overlaid with the extracted skeleton:





Tasks/Questions:

- Did the skeleton model accurately capture your body geometry? **Yes, however it does seem to get confused if other objects are too close to the user's body (image 1 has a chair close by)**
- How would you estimate the elbow angle from the skeletal model? **The elbow can can most liekly be estimated using the shoulder indicator and forearm indicator from the skeletal model.**

Submitting This Lab

Export this *.mlx file as Lab7_##_<LastName1>_<LastName2>_<LastName3>_<LastName4>.pdf (## should be your computer number) and email to Lauren.Diaz@jhuapl.edu