Lesson Five:

Detecting Poses

Myo™

Lesson Five:
Detecting Poses

Overview

Last lesson, we introduced `onForegroundWindowChange` and how it can be used to detect applications.

This lesson will focus on writing a detailed function. We will look at `pose` and `edge`, two important arguments which decide what the application does (what poses do and when to detect poses). We can use `if` statements to build these parts.

Objective

By the end of this lesson, you will be able to write a script that works for a certain application. You will also have learned how to detect poses, so you can start assigning functions to each pose.

Goals

- Learn how to use `onPoseEdge`
- Explore the concept of `edge`
- Print poses to the Debug Console
- Learn how to detect specific poses

Lesson Length (Time)

60 Minutes

PART 1: POSES

The Myo armband detects five hand poses: Wave Out, Wave In, Fist, Finger Spread, and Double Tap. When and how to use each pose is covered in the UX guidelines[1].

To detect a pose, we can use the second of our predefined callbacks: `onPoseEdge`. The format for this function, shown below, is very similar to `onForegroundWindowChange`.

```
function onPoseEdge(pose, edge)
```

We will cover what `edge` means later. Currently, `pose` can have one of the following values: `waveIn`, `waveOut`, `fist`, `doubleTap`, `fingersSpread`, `rest`, and `unknown`. `rest` is the state where the user is not actively posing, and `unknown` means that the Myo armband can't determine any poses.

PART 2: WHAT IS EDGE?

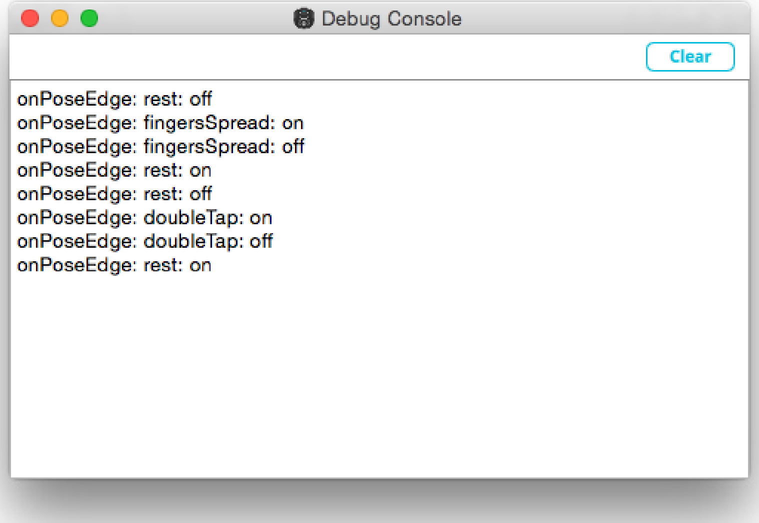
`edge` will be set to `on` when a pose is read, and `off` when the user releases it. This will allow you to start listening for changes in the position of the Myo armband when necessary. You will see this "edge" concept again when we discuss keyboard and mouse button presses.

PART 3: DETECTING POSES ON DEBUG CONSOLE

Printing the pose and edge state to the Myo Debug Console is similar to the last lesson with onForegroundWindowChange. You can add something like this below to your previous code. If you don't want your script to continue outputting the app and title, remove the line of code that contains myo.debug from your onForegroundWindowChange function.

```
function onPoseEdge(pose, edge)
   myo.debug("onPoseEdge: " .. pose .. ": " .. edge)
end
```

Save the file and refresh it in the **Application Manager** (using **Developer Mode**). Try printing poses and the edge state to the console yourself, and compare what happens in the debug log and the pose window (the circle at the bottom of your screen that displays what poses are being made). The debug console should look something like this:



```
onPoseEdge: rest: off
onPoseEdge: fingersSpread: on
onPoseEdge: fingersSpread: off
onPoseEdge: rest: on
onPoseEdge: rest: off
onPoseEdge: doubleTap: on
onPoseEdge: doubleTap: off
onPoseEdge: rest: on
```

Note that you have to unlock the Myo armband with a double tap before any poses come through to your script. Until we come back to advanced locking behaviour, the armband will lock itself automatically after a few seconds.

PART 4: IF STATEMENTS

We are going to implement `onPoseEdge` through several `if` statements. This means you will be performing a specific task for each gesture you care about. In Lua, `if` statements look like this:

```
if <conditional> then
   -- do something
elseif <other conditional> then
   -- do something if something else is true
else
   -- do something else
end
```

An `if` statement evaluates the conditional statement first. If it is true, it executes the code after `then` and nothing else in the rest of the `elseif`/`else` statements. If the original conditional statement is not true, it goes through any `elseif`s in order and does the same thing. If it gets all the way down to an `else`, it will execute that code. For example, you might add something like this in your `onForegroundWindowChange` function to detect your applications in the foreground:

```
if (title == "My Application") then
   return true
elseif (title == “My Other Application”) then
   return true
else
   return false
end
```

This `if` statement only returns `true` if the `title` variable has been assigned the string value “My Game”. If you want to use this in your script, you can find the title of your application by using `onForegroundWindowChange` to output it to the debug console.

Remember to close any `if` statements with `end`.

PART 5: DETECTING SPECIFIC POSES

It is recommended to detect the pose you want in `onPoseEdge`, and then handle what happens for each `pose` in a separate function. Add your logic to your `onPoseEdge` implementation that will detect any gestures through `edge`, and call a separate function for each pose (something generic like `onWaveOut()` for Wave Out).

If you wanted to perform a function for wave out specifically, it would look something like this:

```
function onPoseEdge(pose, edge)
   myo.debug("onPoseEdge: " .. pose .. ": " .. edge)

   if (edge == "on") then
   if (pose == "waveOut") then
                onWaveOut()
         end
   end
end
```

This function first waits for a pose to be detected and then runs the `onWaveOut` function if the pose detected is `waveOut`.

Challenge Activities

The following code is from the previous lesson. You may use it as a template to complete the first question.

```
scriptId = 'com.thalmic.examples.myfirstscript'
scriptTitle = "My First Script"
scriptDetailsUrl = ""

function onForegroundWindowChange(app, title)
myo.debug("onForegroundWindowChange: " .. app .. ", " .. title)
         return true
end
```

1. Write a script using `onForegroundWindowChange`, but use `if` statements so that your script only outputs the `app` and `title` to the debug console for a specific title. For example, if you clicked on your preassigned window the output message would appear as it normally would. However, if you changed your foreground window to any other window, nothing would be output to the debug console.

2. In Part 5 of this lesson, we implemented a specific function for wave out inside of `onPoseEdge`. Implement the other gestures by adding to the if statement, and use default function names such as onWaveOut, onWaveIn, onFist, and onFingersSpread.

Footnotes

[1] https://developer.thalmic.com/ux

Solutions

1.

```
scriptId = 'com.thalmic.examples.myfirstscript'
scriptTitle = "My First Script"
scriptDetailsUrl = ""

function onForegroundWindowChange(app, title)
        if (title == "Replace this text with Application Title") then
                myo.debug("onForegroundWindowChange: " .. app .. ", " .. title)
        end
end
```

2.

```
function onPoseEdge(pose, edge)
        myo.debug("onPoseEdge: " .. pose .. ": " .. edge)

        if (edge == "on") then
                if (pose == "waveOut") then
                        onWaveOut()
                elseif (pose == "waveIn") then
                        onWaveIn()
                elseif (pose == "fist") then
                        onFist()
                elseif (pose == "fingersSpread") then
                        onFingersSpread()
                end
        end
end
```