6. Experiments of controlling the music (VLC player or Youtube):

Description of the Experiment:
After knowing that the system can detect the hand motion and can control the presentation with the hand motion, Team AD added more lines on the python code in order to control the music playing on VLC player or Youtube. The explanation on the changes on the python code is shown in the next section.

Explanation of the Code used in the Experiment:
For controlling the music on VLC player or Youtube, Team AD decided that the system would detect three more types of hand motion to perform more tasks. As a result, the system detects left to right movement, right to left movement, the placement of the hand on the left sensors for a certain period, the placement of the hand on the right sensors for a certain period, and the placement of both hands on both left and right sensors for a certain period.
Team AD decided to increase the volume of music by placing the hand on the right sensor, to decrease the volume of music by placing the hand on the left sensor, and to pause/resume the music by placing both hands on both sensors. If hand is placed on either left or right sensor for a long time, the volume of music is decreased or increased continuously. However, the pause and the resume of the music is not going to happen continuously by placing both hands on both sensors for a long time.
In order to make the system to detect additional hand motion and to execute the tasks associated with the additional hand motion, the code is modified. “space_bar” value is assigned to False initially. “space_bar” value is going to be used to make sure the resume/pause of the music does not happen continuously due to the placement of both hands on both sensors for a long time. In other words, both hands must be removed and placed again in order to pause/resume the music again.
Since all of the additional hand motion involves the placement of the hand on the sensor/sensors for a few seconds, the values appending to the list should be consistently bigger than 0 for some time interval. Three additional movements are checked when the length of the list is greater than or equal to 4. Using the for loop, the first four values of the lists for each sensor are checked.
If all of the first four values of one of the lists or both lists are greater than zero, the values are keep appending to the lists until the system detected the removal of the hand(s) or the length of the list(s) is the maximum length of list. At that point, the lists of both sensors changed back to empty lists.
If all of the first four values of both sensors’ lists are greater than zero and “space_bar” value is False, the space bar is going to be pressed and released using “keyboard.press(Key.space)” and “keyboard.release(Key.space)” since it is the first time that system detected the placement of hands on both sensors. Then, the music is going to be paused or resumed. In order to not press the space bar to pause/resume the music without the removal of hands on both sensors, “space_bar” value is changed to True. If the system detected the removal of the hand on either one of the sensors while “space_bar” value is True, the last value of list(s) is going to be less than or equal to 0. Then, the lists of both sensors become empty lists. Also, “space_bar” value changed back to False.
If all of the first four values of only the left sensor’s list are greater than zero and the last value of the list is greater than zero, the volume is going to be increased by pressing and releasing the up key using “keyboard.press(Key.up)” and “keyboard.release(Key.up).” Until the hand on the left sensor is removed, the last value of the list is going to be greater than 0. Therefore, the volume is going to keep increasing based on the condition. If the hand is removed on left sensor,

the last value of the list is going to be less than or equal to 0. Then, lists of both sensors get emptied.
“keyboard.press(Key.down)” and “keyboard.release(Key.down).” are executed to decrease the volume at the condition of having positive values on the first four positions and the last position of the right sensor’s list only. Since the last value of the list is going to be greater than 0 when the hand is covering the right sensor, the volume is going to keep decreasing until the hand is removed from the right sensor. The last value of the list is going to be less than or equal to 0 when hand is removed from the sensor. At this point, the lists become empty lists.
The code used in the previous experiment on pressing the right key after the detection of the left to right movement and pressing the left key after the detection of the right to left movement is still used. Therefore, the left to right movement of hand would forward the music by few seconds while the right to left movement of hand would rewind the music by few seconds.
Like the previous experiment on testing the control on the presentation, the two light sensors, A/D converter, Raspberry Pi, and VNC viewer are used to check whether the system can control the music on VNC player or Youtube. First, the python code is opened and is started. Then the music is played on either VNC player or Youtube. Before moving the hand, it is important to click the music player or the video portion of the Youtube page to control it. Then, the system is checked by executing all possible hand motions. If the system can execute the specific tasks designated to each hand motion, Team AD concludes that the system works and can control the music.

Result of the experiment:
Demonstration Video shows that the system can detect the hand motion and execute the appropriate tasks. It shows that the system can pause/resume the music, increase/decrease the volume, and forward/rewind the music.