RUTGERS
School of Engineering

NEW BRUNSWICK

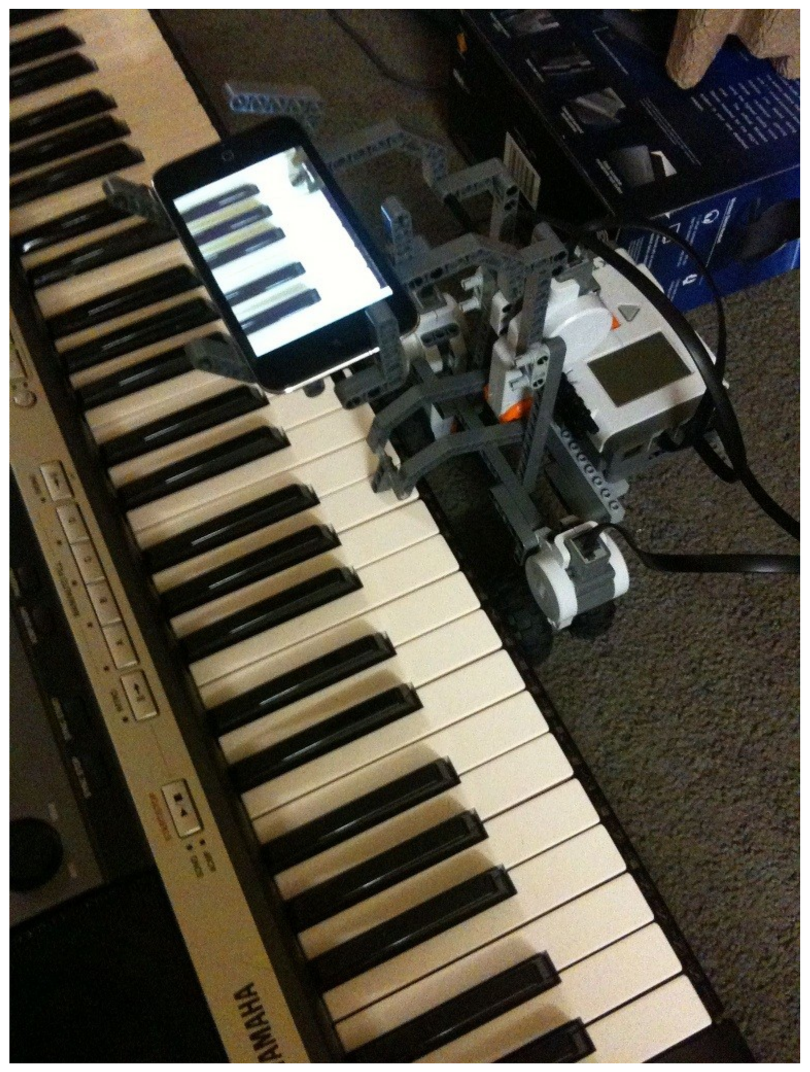
Piano-Playing Lego Mindstorm NXT Robot with iPhone

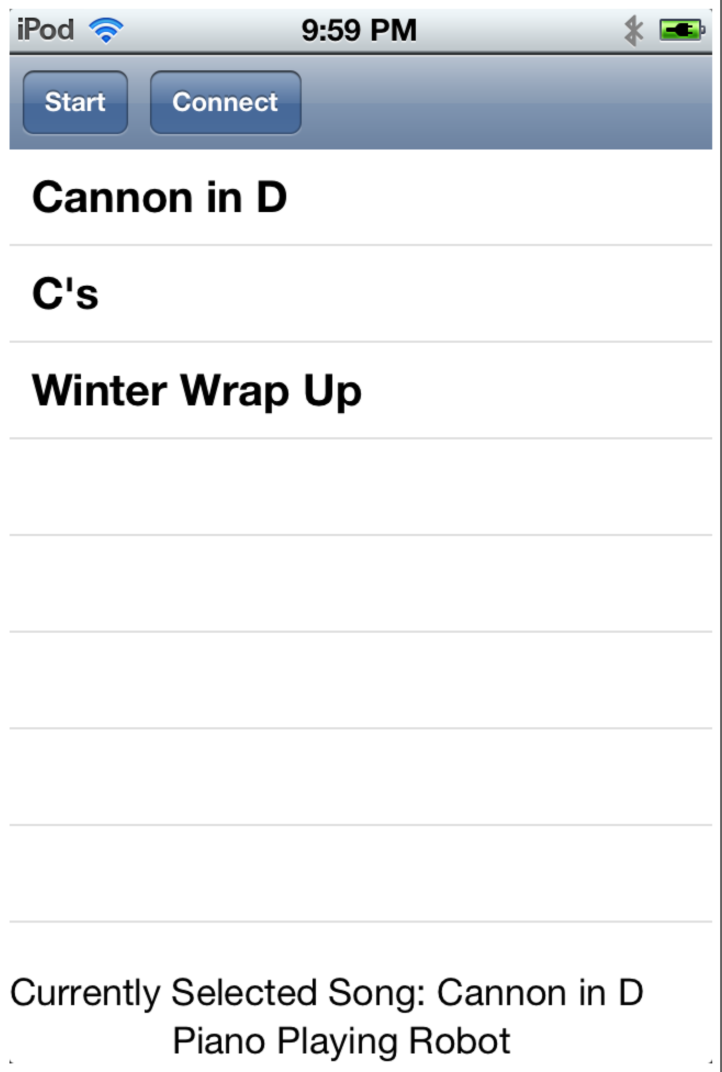
By: Alvin Liao and Ronald Orozco

Faculty Mentor: Professor Kristin J. Dana

ABSTRACT

In the 21st century mobile devices have made an enormous presence around the world where an estimate of two thirds of the world's population uses one. Not only has the number of mobile users increased exponentially, so has the features and computing powers of the mobile device. As mobile devices are ubiquitous and have the capabilities to run complex applications, it is possible to incorporate robots and mobile devices together. The opportunities to develop robotic based mobile applications for the public is both welcoming and encouraging. The vision of this project is to develop a mobile robotic system consisting of the Lego Mindstorm NXT and incorporating computer vision through the Apple iTouch camera.





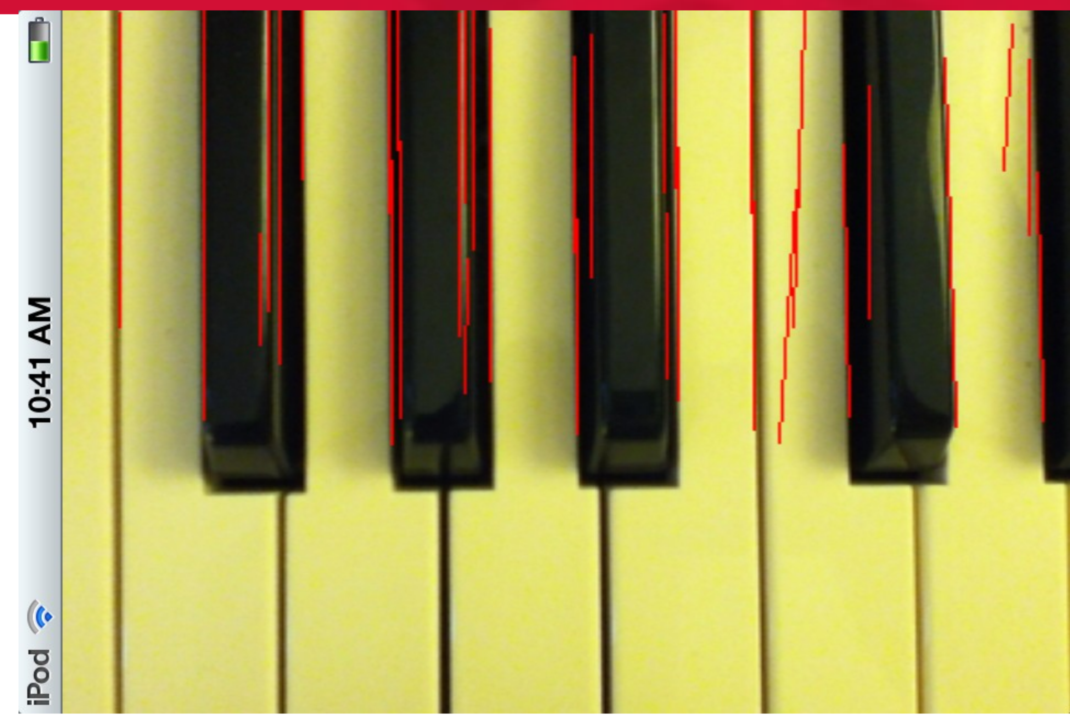
OBJECTIVE

The concept of this project is to develop a piano playing robot using the concepts of computer vision. Using computer vision is not the most practical way to develop a piano playing robot compared to a robot that can play based off the predetermined location of the piano. However by using computer vision the robot will imitate the methods used by humans to play the piano. Therefore them objective of this project is to develop a robot that will simulate a human mind playing a piano. The functions the robot will perform includes distinguishing notes, recognizing piano keys, and playing the piano. In terms of distinguishing notes, the robot will be able to analyze and compare the raw data of notes in order to interpret the data into robot movements. The robot will be able to identify which piano key it is currently in front of and actively search for the next requested piano key. To accomplish these objectives the project will implement Bluetooth and OpenCV.

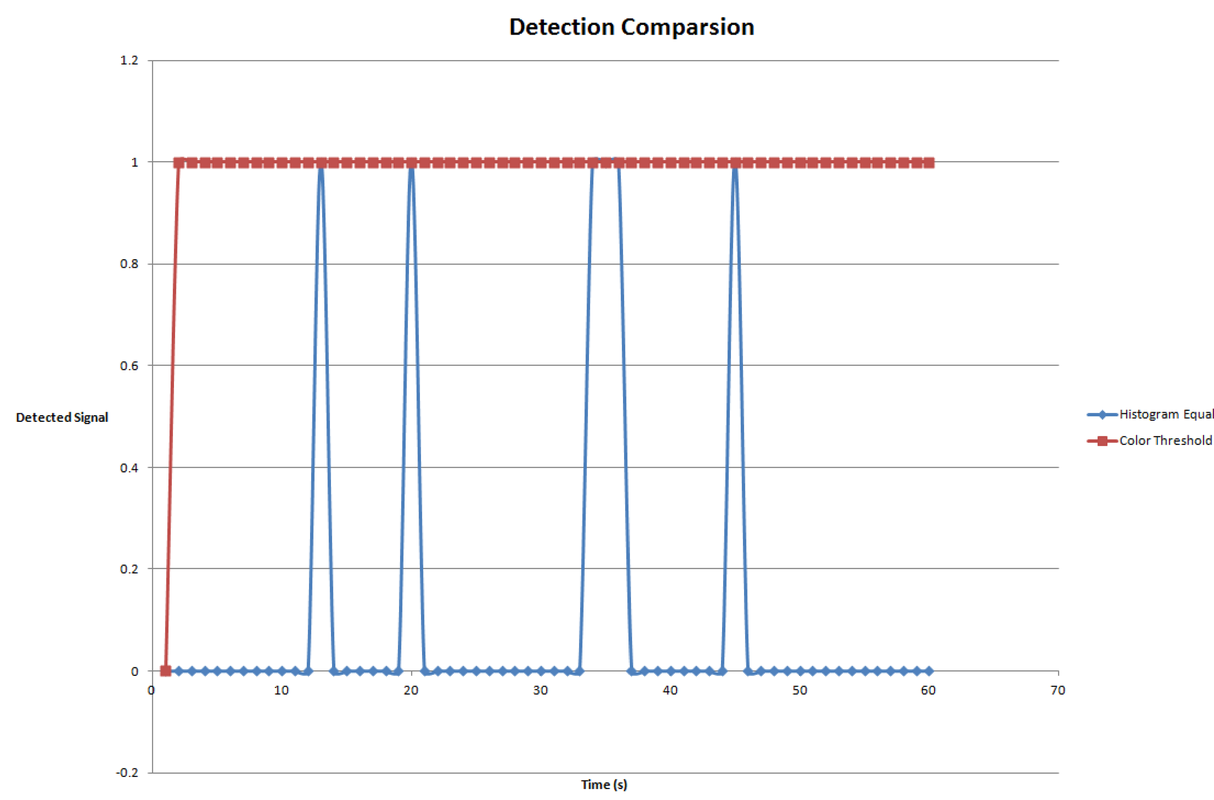
Set Speed (degrees per second)	Correct Trials out of 10	Percent Error %
90	1	10%
75	1	10%
60	2	20%
45	9	90%
30	10	100%

LIMITATIONS OF THRESHOLD

Color Threshold is clearly a great method that can detect piano keys in a fixed environment. While the method is efficient in a fixed environment, Color Threshold has limiting performance with incorporated movement. The first experiment tested the maximum speed the robot can move with acceptable accuracy. The first speed tested was 90 degrees per second. This speed proved to be too fast as the robot was skipping frame in between keys and therefore mixing up octaves. Similar results came with speeds of 75 and 60 degrees per second. At 45 degree per second, the number of frames the camera was reading matched somewhat perfectly with the robot movement. The only error came when the robot pressed the wrong note which was right beside the correct note. At slower speed such as 30 degree per second, there were very little error if not any. Although the robot is able to play somewhat successfully at 45 degrees per second, in reality this is very slow. Even very simple music piece would require the robot to move at least 90 degree per second if it were to reach G3 quarter note from C3. However since the robot is unable to move at fast speed without losing accuracy, the robot is not at the stage where it can be used for practical purposes.



The picture taken is a iTouch camera view of the piano key using the Histogram Equalization and Canny Edge Detection technique. The lines being shown in the picture are created using Hough Line Probability Algorithm. As shown the lines are concentrated around the black key area. There are however other lines scattered in areas that are not black keys and therefore give false data.



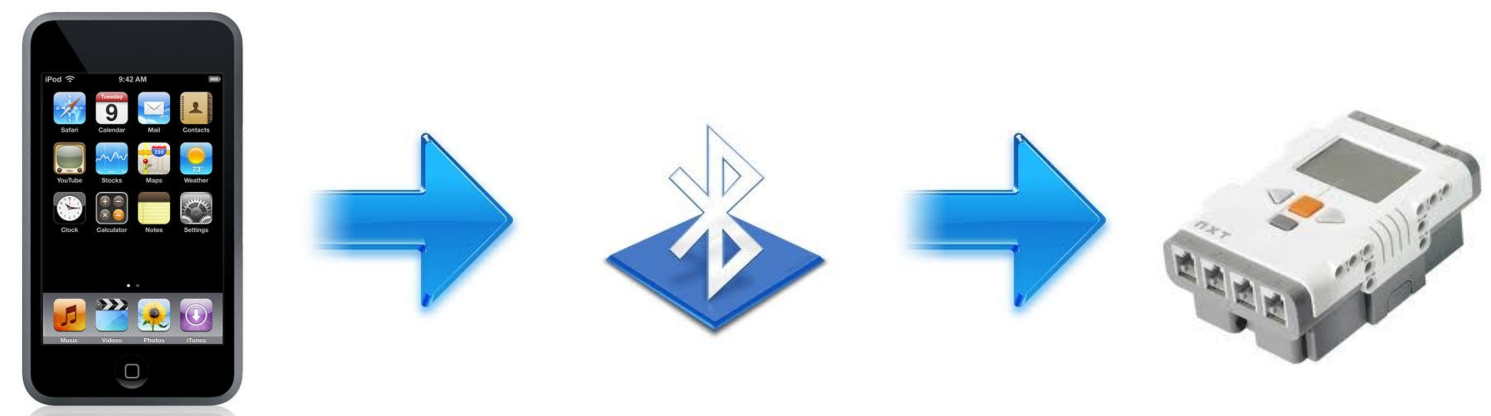
The graph shows a comparison between the two methods used for piano key recognition in the project. The experiment tested the amount of correct detection of the piano key C3 over the course of one minute. The value of 1 signifies a correct detection. A value of 0 signifies a incorrect or unidentifiable detection. The red line represents the Color Threshold method and the blue line represents the Histogram Equalization method.

Trial #	Histogram Time (s)	Color Threshold (s)
1	12.912	0.110
2	10.887	0.090
3	11.150	0.080
4	15.472	0.070
5	12.552	0.080
6	11.966	0.080
7	11.334	0.070
8	13.651	0.070
9	10.984	0.080
10	10.272	0.080
Average	12.118	0.080

The table is a comparison between the two methods used for piano key recognition in the project. The experiment tested the frames per second which would be used to measure the performance. Over a run of 10 trials the Histogram Equalization method had an average of 12.118 seconds per frame and the Color Threshold had an average of 0.08 seconds per frame.

BLUETOOTH CONNECTION

For the Bluetooth portion, the project will be using the RFComm and HCI protocols from the BTStack Library. The Bluetooth connection is initiated by the iTouch while the Lego NXT is in standby mode awaiting a request from the iTouch. The first set of packets sent are part of the HCI protocol which determine the requirements for the connection. The HCI protocol is used to search for the Lego NXT device and establish the initial connection between the iTouch and Lego NXT. During the connection process, the iTouch may be required to enter a PIN number which is set to "1234" by default on the Lego NXT. Afterwards the iTouch will attempt to open a channel which will be used to transfer packets to the Lego NXT. When the connection is set, the iTouch will be able to send packets consisting of bytes to the Lego NXT. The purpose of the RFComm protocol is to send the data packets to the Lego NXT. The direct transfer of packets must use the RFComm protocol as required by the Lego NXT Bluetooth capabilities. The packets sent consist of hexadecimal values. For the sake of simplicity the system is design such that only the iTouch sends packets to the Lego NXT which would receive the packets.



PIANO KEY RECOGNITION

To identify and distinguish the piano keys, the robot takes notice of the placement of the black keys on the piano. This method is similar to the method use by humans to identify piano keys. The feature on the piano is that there are two sets of black keys, one set contains two while the other set contains three. Base on this feature, the robot will divide the image into seven regions of interests. Since the piano is conveniently colored in black and white, the robot tracks for black pixels in the image. The image is converted to grayscale to give the image pixel values relevant to the color intensity of black. If the region is identified to be black, the value assigned is 1. If the region is identified to be white, the value assigned is 0. In the case where the black key may be in between regions, both regions containing the black key will be considered. If both are mostly black, the first region will be assigned the value 1, otherwise 0. The seven regions produce a signal that represents a key and each key has a unique signal assigned to it.

KEY LOCATION TRACKING

The robot is able to make intelligent decision regarding the movement it must make to get to the next note. Aside from recognizing which key is being read, the robot must also track which octave on the piano it is currently looking at. Normally on a 61-Key Keyboard, one would first start at middle C or C3. To simulate the same process a human when use when playing a piano, the robot will assume that it would be starting in the third octave. The robot will keep track of the octaves through two keys, B and C. If the robot is moving right and locates a C, the robot will recognize that it has jumped to the next octave, therefore increasing the octave. If the robot is moving left and locates a B, the robot will recognize the it has move down to the octave below, therefore decreasing the octave.

Piano Key	Signal
C	011101X
D	11101X0
E	1101X01
F	101X011
G	01X0111
A	1X01110
B	X011101