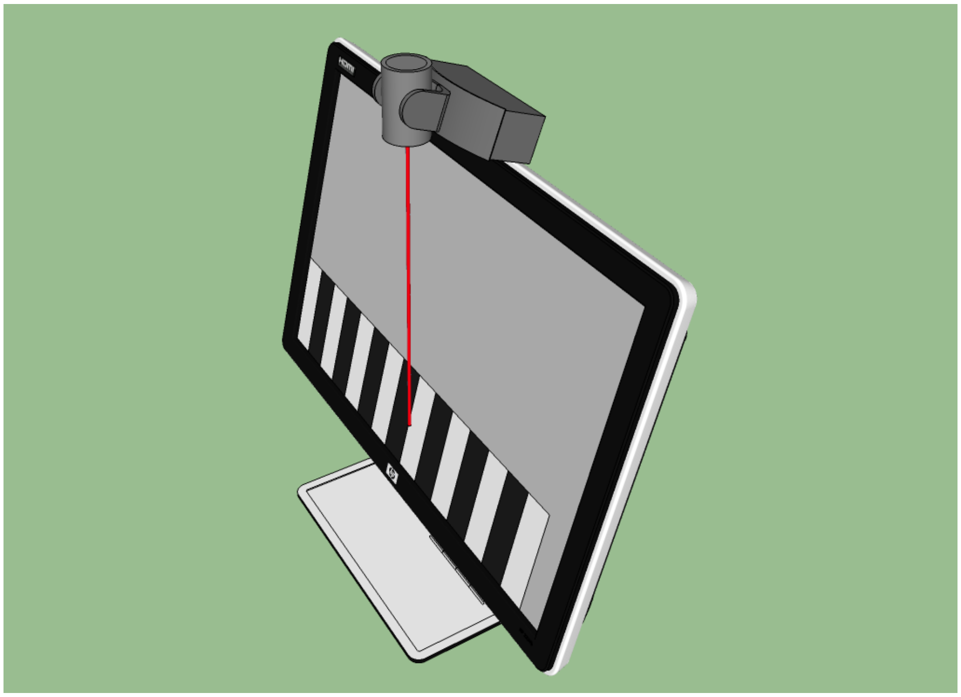
PIANO PLAYER WITH VIRTUAL TOUCH KEYBOARD



2/23/2014

Project Proposal

CSEE W4840 EMBEDDED SYSTEMS

Spring 2014 Prof. Stephen A. Edwards

By Daran Cai (dc2946@columbia.edu)

Linjun Kuang (lk2578@columbia.edu)

Wei Xia (wx2147@columbia.edu)

Wenyuan Zhao (wz2262@columbia.edu)

Piano player with Virtual touch keyboard

PROJECT PROPOSAL

INTRODUCTION

Nowadays, there are many popular computer software and mobile applications for playing virtual instruments, especially piano. These applications are useful for children education and entertainment. However, most of them need relatively expensive peripherals such as a physical keyboard or a sensitive touch screen. For our project, we are going to implement a piano player. Unlike traditional piano player software and applications, we will use a normal screen and a camera to realize a virtual touch screen and a virtual keyboard. We got the basic idea of our project from a project made by Cornell student, but we will do many innovations and improvements in our project. The description of this Cornell project is listed in reference part.

PROJECT FEATURES

The key point of our project is how to use a camera and a normal screen to realize the function of a touch screen. The basic idea is that camera is used to capture the image of fingers and the screen in real time, and then using computer vision algorithm to recognize the relative position of the finger on screen. Using this relative position, we can achieve the function of the touch screen.

As for the user interface, there will be a keyboard on the bottom of the screen. When users press those virtual keys, the corresponding sounds will come out.

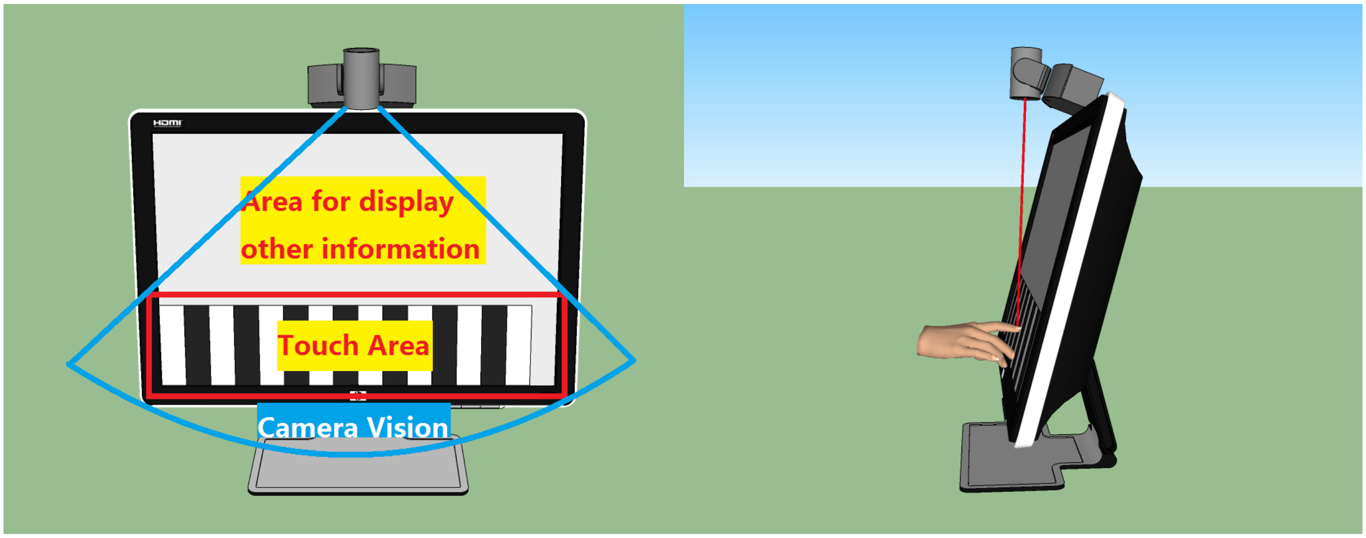
Except for the basic free mode, in which users can play whatever they like, we will provide other modes such as tutorial mode and player mode. If we can realize these modes, we can add more interesting modes.

Some details about the project are listed below.

Building a virtual touch screen system

The position of the camera is very important. If the camera faces the screen directly, then the hand will shield the vision of the camera, so we cannot decide whether the finger is touching the screen, and which key the finger is pressing.

Hence, the camera should be positioned on the top of the screen, as showed in following picture. On the one hand, the keyboard will be displayed at the bottom of the screen, so the camera can catch the full view of the keyboard, just as the first picture. On the other hand, in the second picture, we can see that if the finger is not touching the screen, the camera will catch the whole keyboard, however, if the finger is touching a key, the corresponding area will be shielded, and we will know which key is being pressed using some image processing algorithms.



Sound

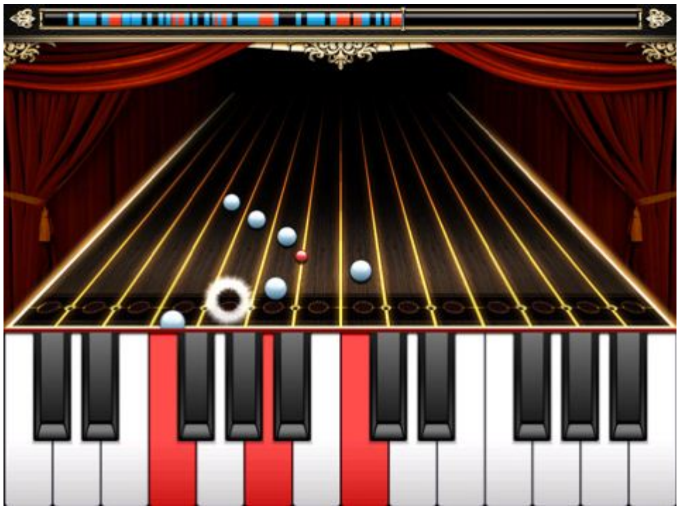
As for the sound part of this project, we figure out two ways to implement piano sound.

1. Using Karplus-Strong string synthesis algorithm to simulate the real piano sound;
2. Store some audio files in memory, and call the corresponding file after recognizing which key is being pressed;

The first method may be faster, while the second one can provide us with more possible sound effects.

<u>We are not sure about how to implement these two methods, and we may need some help from the professor.</u>

User interface



When users press a key, there will be some vision feedback to users, such as changing color of the key or adding shadow effect, making them feel like playing on the real touch screen.

The project will support various modes, and some interesting animation effects will be added to these modes. For example, there will be some falling blocks in player modes, like the picture on the left.

CHALLENGES

1. Since the camera is connected to the FPGA via USB, the control of the camera, and the storage of images might be a problem.
2. After the capture of images, a tough problem is how to process these images. Due to the environment and the condition of light, there must be some noises in the original images. The irregular shapes of fingers will also bring some difficulties. So we need to find a robust algorithm to deal with these problems
3. Since we have little background knowledge in sound part, we need more research to complete this part.
4. Although the key point of our project is not in user interface, we still want to make it fancier and this might add much workload for this project.

REFERENCE

Cornell project link:
http://people.ece.cornell.edu/land/courses/ece5760/FinalProjects/s2013/cl972_rh523/cl972_rh523/index.html