

Vissynth Project Proposal

Chang
Evan
Sam
Yui

November 11, 2010

Vissynth is a visual synthesizer, receiving input from a webcam. By changing the input to the camera (e.g. moving around frantically in front of the camera), a user can change the music produced by the synthesizer.

To accomplish this, we will use the OpenCV library to handle talking to the webcam. We intend to have our program save individual images from the camera at a given rate (e.g. 10 Hz), and then have our program operate on the images in the appropriate sequence (first image in time will be read first, followed by second, third, etc.) instead of attempting to handle live input from the camera.

For every image saved, the program will do (roughly) the following:

1. Turn the contrast of the image to 100%, leaving only black and white pixels.
2. Separate the image into a 10x10 grid, each corresponding to a button on the synthesizer (explained below).
3. Count up the number of white and black pixels within each square of the grid.
4. For each square, mark that square ON if there are more white than black pixels, OFF otherwise.

We will then use the ON/OFF sequences as input to the synthesizer in the following manner. Each column of a grid represents a time point (i.e. column 1 = time 1, column 2 = time 2, etc.), such that all of the squares within a given column will be played at the same time. Each row corresponds to a particular frequency, with lower rows representing lower tones and higher rows representing higher tones.

Our group would like to design a program that functions as a visual synthesizer. We intend to take input from a web camera and use this input to drive a musical synthesizer. To achieve this, we will separate the image into a grid, which will roughly correspond to the grid of a synthesizer: an array of buttons,

where columns represent "time steps", every ON button in each column goes off at the same time point, and the rows represent frequencies, with lower rows representing lower tones and higher rows representing higher tones. The program will decide which squares of the grid are ON and which are OFF based on the following process:

1. turn the contrast all the way up on the image
2. count up the number of white and black pixels within each square of the grid
3. for each square, mark that square as ON if there are more white than black pixels, OFF otherwise.