Multi-Laser Gestural Interface — Solutions for Cost-Effective and Open Source Controllers

Meason Wiley
California Institute of the Arts
24700 McBean Parkway
Valencia CA 91355
jameswiley@alum.calarts.edu

Ajay Kapur
California Institute of the Arts
24700 McBean Parkway
Valencia CA 91355
akapur@calarts.edu

Abstract

This paper describes a cost-effective, modular, open source framework for a laser interface design that is open to community development, interaction and user modification. The following paper highlights ways in which we are implementing the multi-laser gestural interface in musical, visual, and robotic contexts.

Keywords: Lasers, photocell sensor, UltraSound, Open Source controller design, digital gamelan, digital tanpura

1. Introduction

As microcontrollers, sensors and electronic parts get cheaper and smaller, the development of gesture-oriented musical controllers continues to enhance electronic musical performance by providing a clear visual connection between the performer and the audience. By creating a musical interface that focuses on the physicality of the artist, we hope to increase the interactive nature that is so inherent to performance, but that is often lacking in non-academic electronic music. To continue in this tradition, we present a system that is both cost-effective and easy to implement, and that has a multitude of applications.

The history of musical laser interfaces can be traced back to Geoff Rose's invention of the Laser Harp [1] in 1977. Since it's creation, there have been several designs of the Laser Harp, most notably by Bernard Szajner, Yan Terrien, and Phillipe Guerre, who created the first MIDI Laser Harp [2].

Following the development of the Laser Harp, similar interfaces were designed, such as Roland's D-Beam. Leila Hassan's Terminova [3], which uses a Theremin along with lasers and sensors, as well as Ivan Franco's Airstick [4], which uses IR sensors, are both examples of recent advancements in the design and implementation of light-based controllers.

The *Multi Laser Gestural Interface* (MLGI) differs from these in a few ways. For example, most of the Laser Harps are quite large, usually built in a framed structure where the sensors are mounted either directly above the laser, or a mirror is used to reflect the laser light back to a sensor located near the laser module. The Laser Harp also requires a large amount of power to run. The MLGI interface is, by comparison, light weight, portable, and can be powered using a 4.5V power supply (for the laser circuit) and the bus power from the USB port. The MLGI interface works by using the reflected light off of the performers hand, which bounces back to a photocell located directly beside the laser module. This design allows for analog signals to be transferred without the need for a rack design, which, in turn, frees the performer from being confined behind it. The primary difference between the MLGI interface and other gestural controllers is that the design and programming is open source and open to community development. Following Brian Crabtree's [5] (creator of the Monome) method for an environmentally conscious and open source controller design, the MLGI allows anyone who is interested to design and build their own MLGI and contribute through a community-based forum to advance the design in any number of ways.

This paper describes the design of the MLGI in section 2. Section 3 presents a variety of case studies for how the design has been used in musical performance and artistic endeavors.

2. Design

The MLGI interface uses inexpensive 5mW red lasers, photocells, and Dan Overholt's CREATE USB programmable interface (CUI) [6] as the primary components of its circuit. The CUI board has thirteen A/D inputs and 18 general I/O ports, which allows for the ability to switch between both continuous control messages (i.e.: a knob or fader) and on/off messages (i.e.: a button or note on/off). These control messages can be created by breaking the laser beam with the hand (on/off), or by moving the hand up and down within the laser light (continuous), from which light is reflected back onto a photocell housed next to the laser module. The photocell responds to the fluctuation in (or lack of) reflected light, starting at approximately 2 ½ feet above the sensor, which causes an increase or decrease in resistance within the circuit, allowing for the changes in voltage to be registered by the CUI board. The CUI is programmed to send MIDI out via USB, which allows for integration with a number of different musical and visual-based software platforms.

However, we have also implemented OSC using the ChucK platform as a mediator between the CUI and various software that will accept Open Sound Control. During our initial tests we became aware that, depending on the amount of ambient light in a given space, the photocells would start sending signal unintentionally. Because the ambient light in performance spaces varies, we have included a separate photocell with the sole purpose of calibrating the controller in any environment. This ambient signal is used to apply a threshold so that each photocell is always sending out correct values and operates within a useful range.

Another issue we had dealt with the light from one laser being picked up by multiple photocells. We solved this issue by housing the photocells in small rubber domes who's only opening directly faces the sensor's corresponding laser. By doing this, we effectively block out any unwanted light from other laser beams.

We have also experimented with the use of ultrasonic sensors to use in correlation with the photocells, in order to obtain more accurate data for evaluating and processing continuous control messages. Though ultrasonic sensors are more expensive, we recognize the need for accuracy in our data and are currently looking into various ways of implementing them in our design. Details on the design of the MLGI controller are available online [1].

Figure 1. The MLGI interface

3. Case Studies

The modular design of the MLGI has enabled us to find a number of different useful applications. We have implemented the interface for use as a DJ controller, a digital Tanpura, live video controller, and for use with musical robots.

3.1 DJ Control

We have implemented the use of the MLGI for live electronic musical performance, using ChucK to send OSC messages for controlling various parameters on custom-built instruments in Native Instruments Reaktor software.

[1] www.music.calarts.edu/~mtiid/research/interfaces/mlgi

3.2 Digital Tanpura

We built a digital laser-controlled Tanpura, a traditional Indian drone instrument with five strings. This allows a beginning student to perform the Tanpura in an ensemble, without having the skill to fine tune each string with the cumbersome wooden tuning pegs, while still giving the audience a visual reference of strings being plucked.

3.3 Video Control

We have experimented with the MLGI for use with live video manipulation by creating a 3 dimensional virtual space, gathering data of the x and y coordinates for laser position and z from the photo sensor. We worked with video artists to use this data with Isadora and Processing to virtually model real shapes.

3.4 Robotics

We have experimented with the MLGI for use in the Cal Arts Digital Gamelan Ensemble, which has a number of robotic instruments, including a robotic gong. In working with professional Gamelan performers who have no experience using technology in their performance, we were able to provide the ability for the musician to control the robotic gong with the laser interface, while continuing to perform on their own instrument.

4. Acknowledgments

Jeff Lufkin, Dimitri Diakopoulos, Steve Rusch, Bill Alschuler, Owen Vallis, Jordan Hochenbaum, Trimpin.

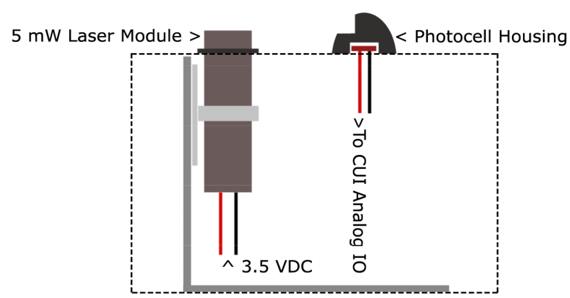

Figure 2. A view of the lasers and photo sensors

References

[1] Laser Spectacles: The Mysterious Laser Harp, [Web Site] 2003, Avail: www.laserspectacles.com/pages/harp.htm

[2] The Laser Harp Pages, [Web Site] 2009, Avail: www.laserharp.org/readarticle.php?articleid=7

[3] Hasan, L., Yu, N. and Paradiso, J., "The Termenova: A Hybrid Free-Gesture Interface", *Proceedings of the 2002 Conference on New Instruments for Musical Expression, Dublin*, May 2002.

[4] Franco, I., "The Airstick: A Free-Gesture Controller Using Infrared Sensing", *Proceedings of the 2005 Conf. on New Instruments for Musical Expression, Vancouver,* May 2005.

[5] Crabtree, Brian - www.monome.org/

[6] Overholt, D., Musical Interaction Design with the CREATE USB Interface: Teaching HCI with CUIs instead of GUIs, *ICMC (New Orleans, LA, 6-11 November 2006).*