# TASK Glove

Zachary R. Trybus
*Cedarville University*, zachtrybus@cedarville.edu

Macy E. Schieber
*Cedarville University*, macyschieber@cedarville.edu

Bailey M. Weiss
*Cedarville University*, bweiss@cedarville.edu

Max A. Ross
*Cedarville University*, maxross@cedarville.edu

# TASK Glove: Hand Gesture Detection and Tracking

Bailey Weiss, Macy Schieber, Max Ross, Zach Trybus

## Abstract

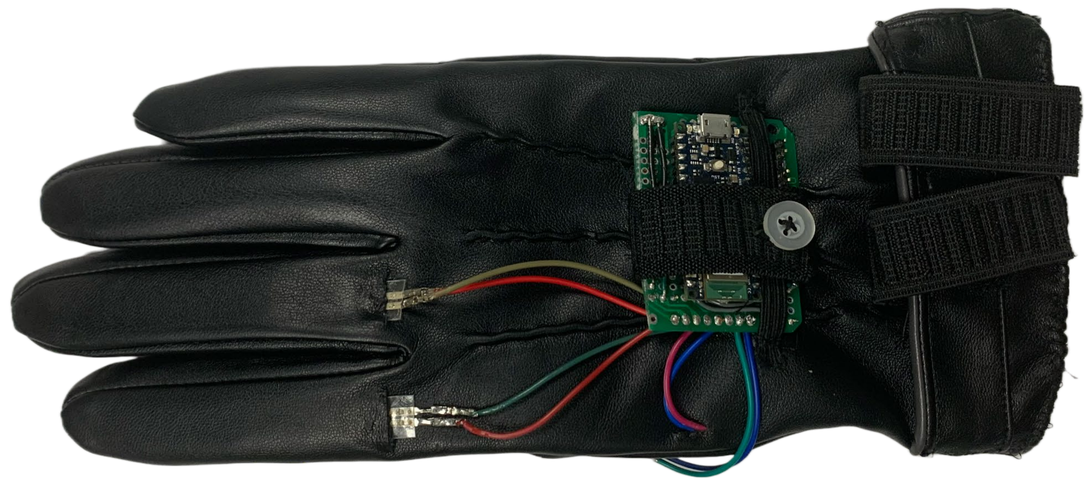
The purpose of this project is to design a glove that can interact with many different surfaces as if they were touch screens. The finished product will be useful for a host of applications. In an academic context, it will allow professors to scroll and click on projector screens during lectures and presentations. Further applications include any context in which the use of a computer may be enhanced by touch screen capabilities.

The glove includes: an Arduino Nano, three flex sensors, a pressure sensor, battery, and IR emitter. The flex sensors generate resistance data for hand gesture detection that is communicated via Bluetooth low energy to a computer. A camera detects the IR emitter's light to determine hand location. The camera capture undergoes a filtering process for this purpose. An application will provide a simple UI with calibration instructions. It will also translate and execute gesture commands.

## Objectives

To achieve the functionality that we are hoping for, we must meet the following requirements:

- The Arduino must be able to recognize gestures with the glove.
- The Arduino must send the gesture to the connected computer.
- The computer program must take the sent gestures and do the corresponding action.
- The computer must be able to track the location of the hand using the attached camera.



## Software

### Gesture Recognition

Using the sensors we attached to the glove, we see the positions of the user's fingers. We start by scanning the current position of the glove ten times per second. With this data, we use a finite state machine to jump between gestures. Using Bluetooth low-energy (BLE), we send an integer that represents the gesture to the app on the computer. To avoid falsely reading gestures, we only send a gesture to the computer if it is polled three sequential times.

### Bluetooth Low-Energy

The Bluetooth Low Energy communication protocol is ideal for this application because of its low power consumption in comparison to other short-range radio communication protocols. Moreover, a high data bandwidth, considered BLE's largest weakness, is not a requirement for this project. For these reasons we chose BLE.

### Hand Tracking

Our tracking process detects the brightest object in an IR channel, which is the IR light on the TASK glove. The camera also detects the edge of the monitor screen to know if the IR light is within frame. If so, the mouse must move. In order to scale hand-movements to mouse-movements on the computer monitor, software detects the actual size of the monitor as well as the size of the monitor from the camera view.

### App Commands and UI

Integers received over BLE are passed through an asynchronous function that executes commands on the computer. We utilize methods within the PyAutoGUI library to accomplish tasks such as clicking and mimicking hotkey commands. The UI is a simple window designed with Tkinter. It lists the available Bluetooth devices for user selection and offers buttons to show the video feed, draw on the screen, and calibrate the IR tracking.

## Hardware

### Camera

Our project requires the employment of a camera to track the IR light on the user's finger. Our software is compatible with a variety of cameras, provided the camera has a designated IR channel for tracking. Cameras with IR channels can be rather cheap, which plays a part in both the practicality and versatility of the TASK glove. Camera setup is rather easy. If the whole screen of the monitor or projector is in the viewing frame, then the TASK glove is ready to use.

### Circuit

The design of our circuit board is quite simple. We have two 3.7V 500mAh batteries connected in series which will power the Arduino Nano as well as the sensors on the glove. Each sensor is built into a resistor divider scheme that limits the range of sensor values to get more accurate readings.

### Arduino

We chose the Arduino Nano 33 BLE microcontroller for this project. It is compact and power efficient. The foremost reason we chose this model is its integrated BLE antenna, which is necessary for communication with the user's computer. The antenna integration precludes the necessity of using an external antenna that would consume space and complicate the power budget.

### Sensors

We built two types of sensors into the glove: flex sensors and a singular button. Three flex sensors run along the thumb, index, and middle fingers. These measure finger bend. The button lies on the pad of the thumb. We primarily use the button to mimic clicking a traditional mouse, yet it also indicates other gestures, such as zooming.

## Results and Significance

After extensively testing each crucial component of both hardware and software, we made the necessary changes to successfully recognize and execute our target commands while also tracking the user's hand position in relation to the screen. Using the glove, users may execute the following commands: scroll, left click, right click, zoom in, zoom out, open draw tool, close draw tool.

- Flex sensors are subject to inaccuracies due to their sensitivity, yet they can effectively detect bend when their resistance values are harnessed within a range
- IR tracking an object with accuracy requires a combination of distortion correction and filtering, yet it produces a smooth, reliable track
- The TASK glove proves that inexpensive tracking methods effectively accomplish the work of current pricey market competitors