



Interacting with Augmented Realities through the Internet of Hospital Things

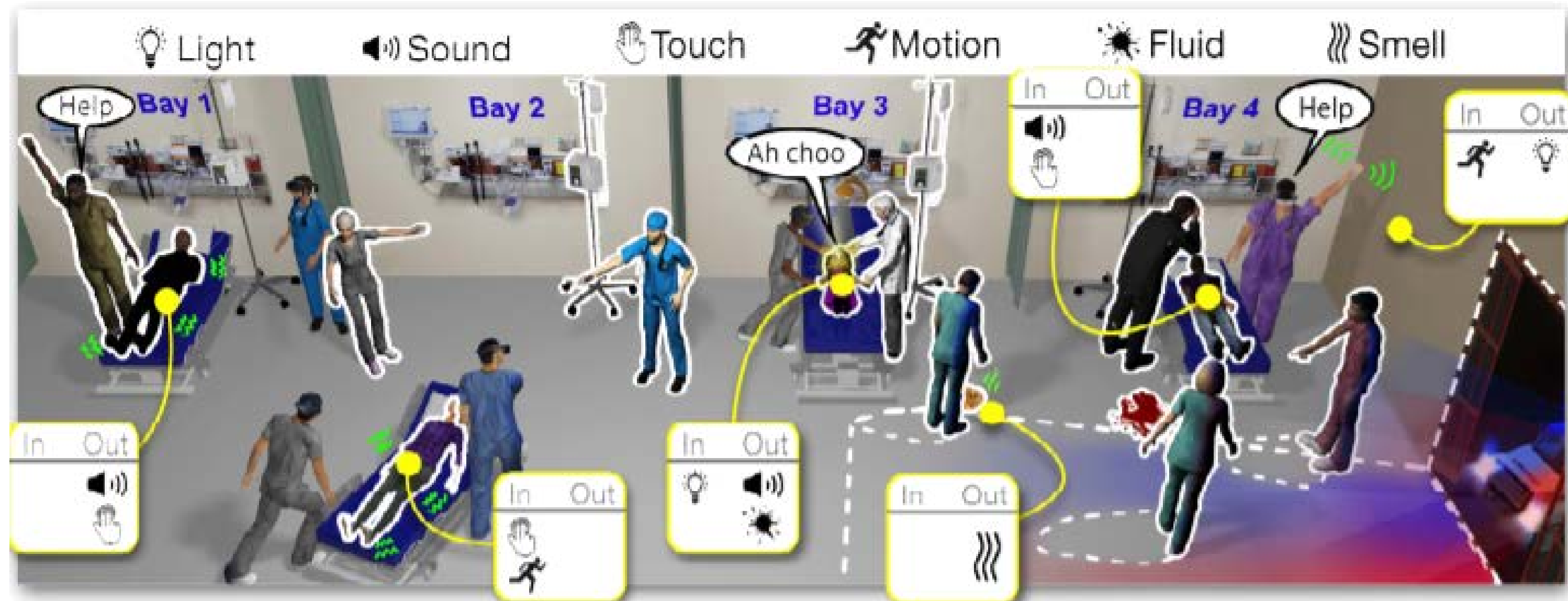
Brandon Belna
Department of Mathematics &
Computer Science
Stetson University
bbelna@stetson.edu

Dr. Greg Welch
Institute for Simulation and
Training
University of Central Florida
welch@ucf.edu

Dr. Gerd Bruder
Institute for Simulation and
Training
University of Central Florida
Gerd.Bruder@ucf.edu



ABSTRACT



The goal of augmented reality over the goal of virtual reality is to blend the physical world with the virtual world, rather than fully immersing a user into thereof. We present a new novel way to enhance augmented realities through Internet of Things devices. The ultimate goal of this is to apply it in professional training sessions, such as for doctors and police.

INTRODUCTION

The ability for IoT devices to interact with augmented reality headsets has many potential applications in simulation, home automation, professional training, and entertainment. In this research we present a way to achieve this via the Microsoft HoloLens. We will also be using an Arduino equipped with an HC-06 Bluetooth shield to act as a simplistic IoT device.



The Microsoft HoloLens, an augmented reality headset.



The HC-06 Bluetooth module used with the Arduino.

We will be using Unity to develop AR environments, and will need to set the following build settings:

- Target .NET 4.6
- Enable Bluetooth capability

CONNECTING DEVICES

Connecting devices requires grabbing a list of available devices from the HoloLens' operating system, singling out the device you wish to connect to, and then opening a StreamSocket on it. This is accomplished by:

- Grabbing a list of devcies through the DeviceInformation class which have RfComm support
- Singling out the device you want by calling Single() on the returned DeviceInformationCollection
- Grab its ID through the RfCommDeviceService class
- Creating a StreamSocket and opening it on the corresponding RfCommDeviceService

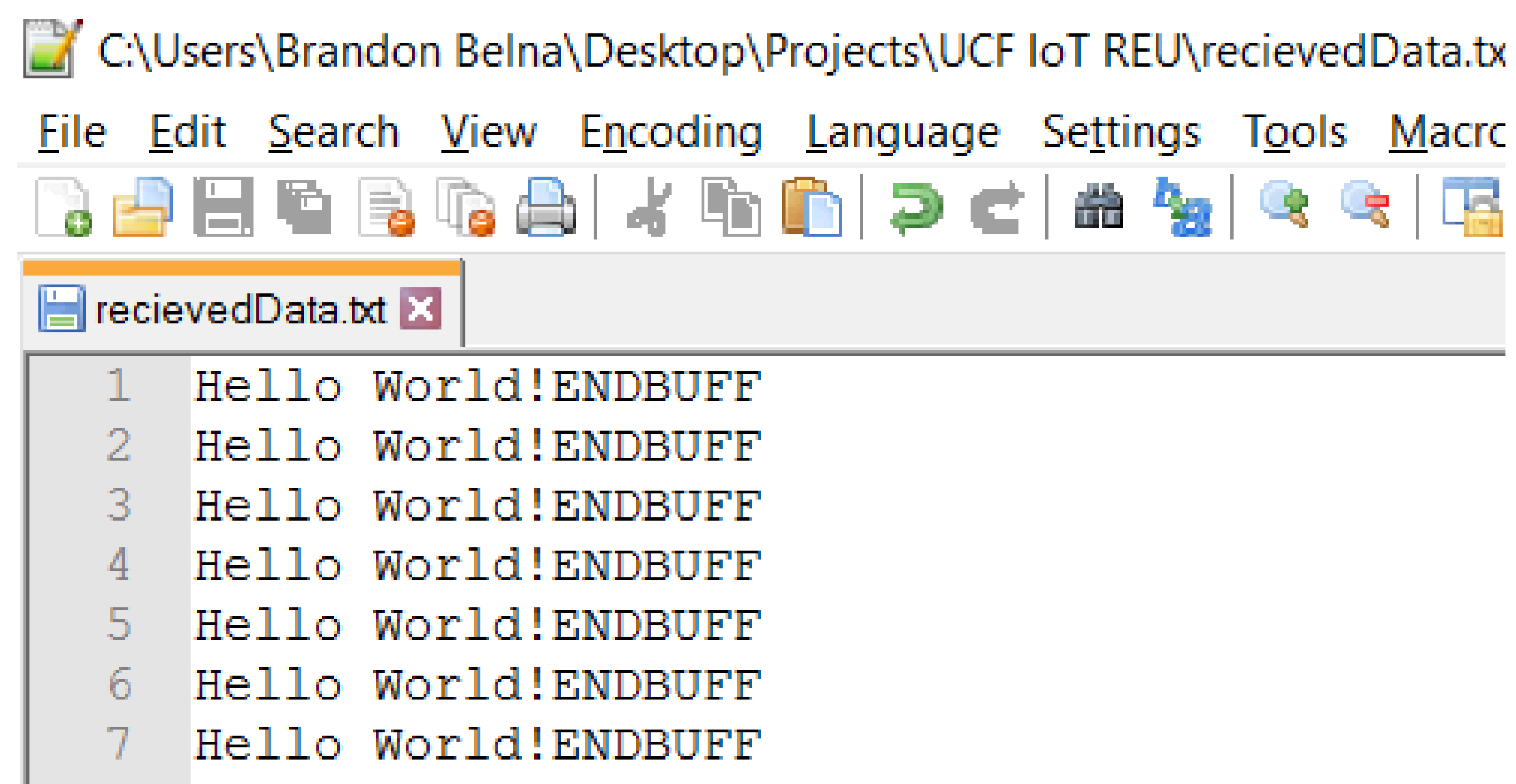
SENDING & RECEIVING DATA

Sending and receiving data is easy once the StreamSocket has been opened. To send data:

- Open a DataWriter on the StreamSocket output stream
- Call WriteString on the DataWriter
- Store the DataWriter task by calling StoreAsync().AsTask() on it

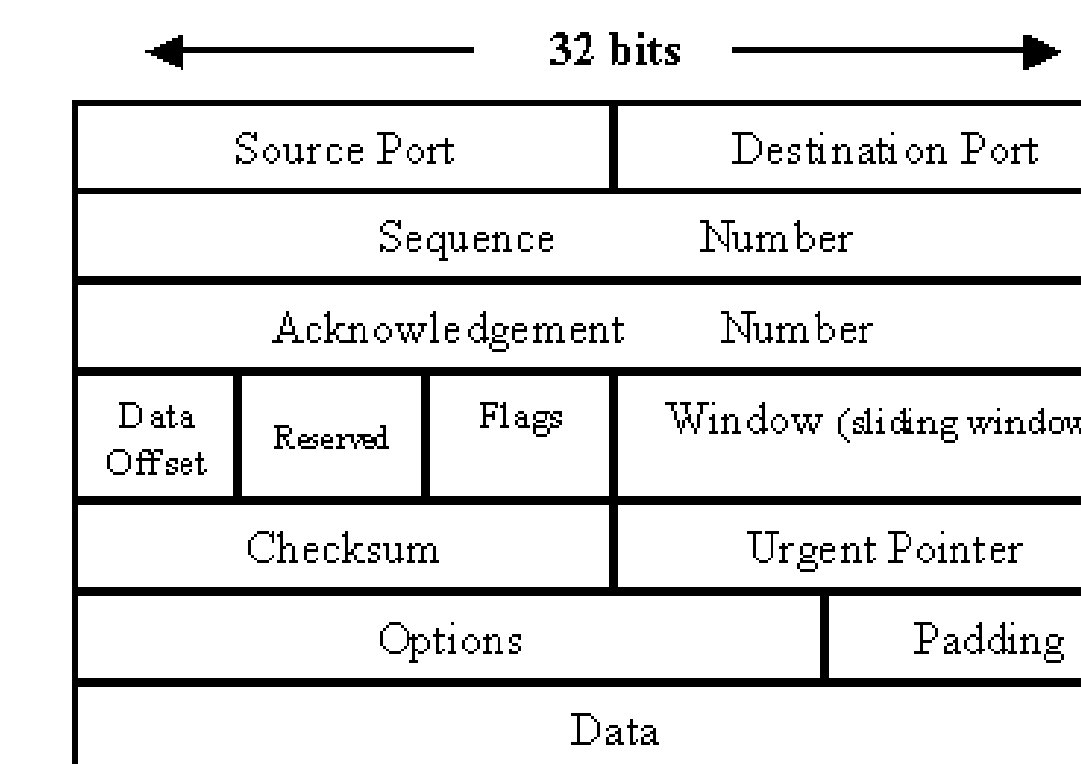
To receive data:

- Open a DataReader on the StreamSocket input stream
- Set the InputStreamOptions to Partial
- Call LoadAsync with the amount of bytes you're expecting



An example of data being received during a demo. ENDBUFF at the end of every line is used to indicate the end of that particular data stream to prevent data transposition.

PACKETING



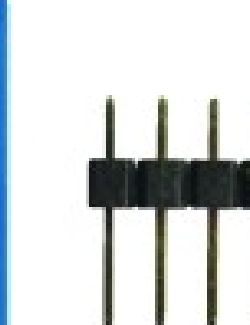
A standard packet used in TCP/IP networking, which can be used as a model for different types of Bluetooth packets.

It is recommended to send data in a TCP/IP like manner with packets. This prevents transposition of data and also allows you to keep track of what data has been received properly and what hasn't. There are several ways of accomplishing this. In this project we took a simple approach which sent data with a timestamp with a start and end data strip. The former allows for the IoT device to receive confirmation of what data was received and when, and the latter prevents transposition.

RESULTS / APPLICATION

The main application of this work at the University of Central Florida's Institute for Simulation and Training is in medical training. Currently in the works is applying this technology to several medical training devices to help enhance the realism of training simulations.

A simple example of this, which is currently in the works, is embedding a small footprint Bluetooth device into a pressure sensor which sends out data of thereof. This can be used in an artificial wound where if pressure is being applied on the wound, in the augmented world bleeding can be stopped.



An Adafruit momentary pressure sensor, which is being used to develop a prototype artificial wound at UCF's IST.

ACKNOWLEDGEMENTS

The support for this work was provided by the National Science Foundation REU program under Award No. 1560302. Any opinions, findings, and conclusions and recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.