

Old Dominion University

ODU Digital Commons

Undergraduate Research Symposium

2022 Undergraduate Research Symposium

Mar 19th, 1:00 PM - 2:00 PM

COVID-19 Classroom Occupancy Detection System

Stephanie Trusty

Old Dominion University

Follow this and additional works at: <https://digitalcommons.odu.edu/undergradsymposium>



Part of the [Other Computer Sciences Commons](#)

Trusty, Stephanie, "COVID-19 Classroom Occupancy Detection System" (2022). *Undergraduate Research Symposium*. 1.

<https://digitalcommons.odu.edu/undergradsymposium/2022/sciences/1>

This Oral Presentation is brought to you for free and open access by the Undergraduate Student Events at ODU Digital Commons. It has been accepted for inclusion in Undergraduate Research Symposium by an authorized administrator of ODU Digital Commons. For more information, please contact digitalcommons@odu.edu.



COVID-19 Classroom Occupancy Detection System

Stephanie Trusty
Summer 2021



Table of Contents

1. Problem Description
 - a. Background
 - b. Project Goals and Equipment
 - c. Python Libraries and Packages
 - d. Limitations and Constraints
2. Solution and Implementation
 - a. Algorithm Flow
 - b. Case Study: A Single Violation
3. Future Considerations
 - a. Recommendations for Improvement
 - b. Benefits of the Recommendations
4. Conclusion



Problem Description

COVID-19 and Social Distancing

What is COVID-19?

- A disease caused by the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2).
- Social distancing is advised to limit the spread of COVID-19/

What is social/physical distancing?

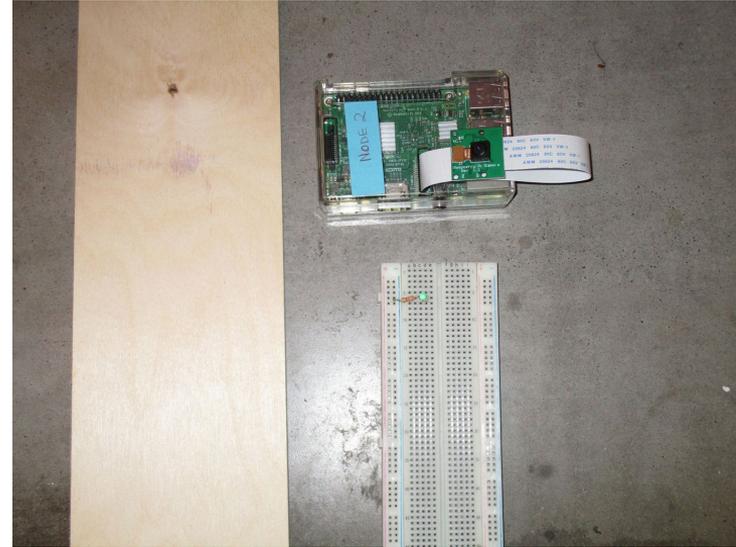
- The practice of maintaining a six-foot distance from individuals that are not of the same household.
- Social distancing is advised in both indoor and outdoor spaces.

Project Goals

- Enforce social distancing protocols within the classroom environment
- Determine if a seat is currently occupied
- Detect if occupied seats are adjacent to each other
- Turn on a green LED if students are properly distanced
- Turn on a red LED if adjacent occupied seats are detected

Project Equipment

- Raspberry Pi 3 Model B
- Raspberry Pi Camera Module v2
- Breadboard
- Green LED
- Red LED
- Male-Female Jumper Wires
- Resistors
- Toy figures



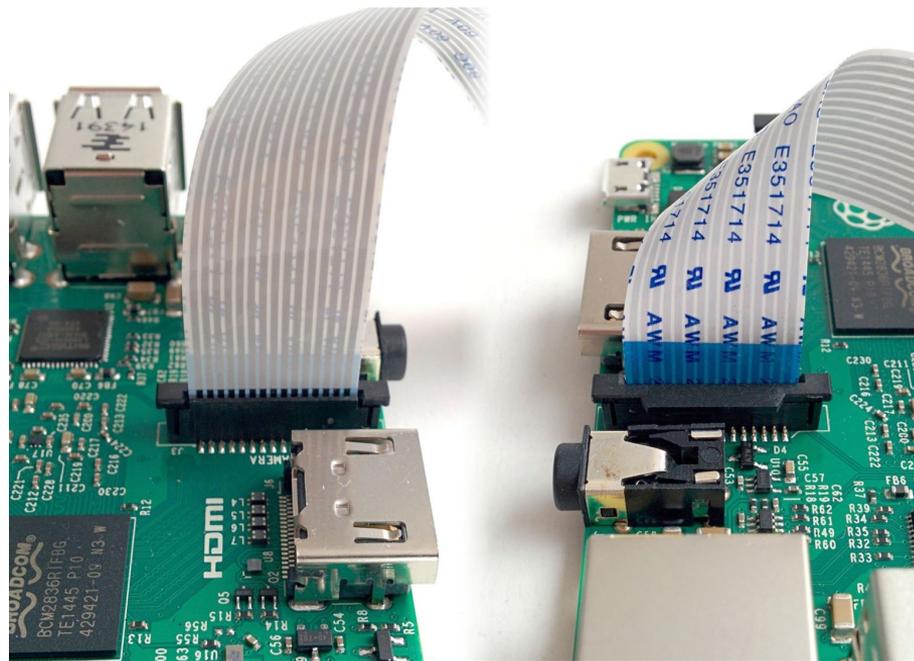
About the Raspberry Pi

- Version: Raspberry Pi 3 Model B
- Single-board computer with bluetooth and wireless LAN connectivity
- Camera Serial Interface port
- 40 general purpose input/output pins
- Micro SD Card port for data storage and OS loading
- HDMI port to connect monitor
- USB ports to connect keyboard and mouse



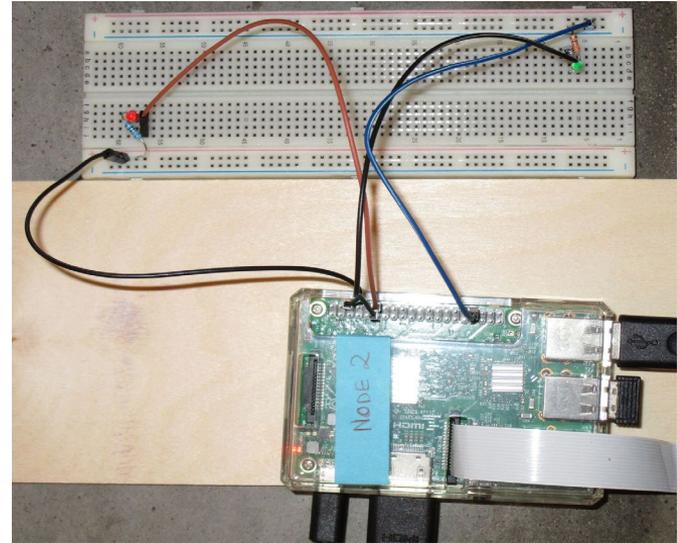
Raspberry Pi Camera Module

- Version: Raspberry Pi Camera Module v2
- Designed for natural light
- Sony IMX219 8-megapixel image sensor
- Captures high-definition videos and still photos
- Connects to CSI port via 15-cm ribbon cable
- Compatible with all Raspberry Pi 1, 2, 3, and 4 models



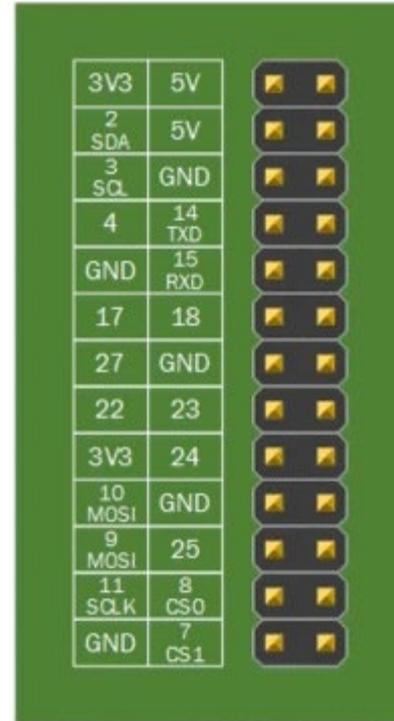
Breadboard Setup

- Two LED circuits are placed on the breadboard
- Male-female jumper wires connect the circuits to the Raspberry Pi's GPIO ports
- For each LED, a series resistor is connected to limit the flow of current through the LED
- The resistor can be placed on the LED's anode side (the positive lead) or its cathode side (the negative lead)



GPIO Pins

- Raspberry Pi 2 Model B has 40 General Purpose Input/Output pins
- GPIO pins allow the computer to connect to and control electronic circuits
- The LED's cathode pin is connected to a GND pin with a male-female jumper wire
- The circuit's positive supply is connected to one of GPIO pins



Python Libraries and Packages

- PiCamera
 - Controlling the Raspberry Pi Camera
- Python Imaging Library
 - Image processing functionality
- Numerical Python (NumPy)
 - Multidimensional array operations

PiCamera

- Control the Raspberry Pi Camera with Python code
- Available for Python 2.7 or above and Python 3.2 or above
- Key Functionality:
 - Start camera preview
 - Stop camera preview
 - Capture image and save to a file

Python Imaging Library (PIL)

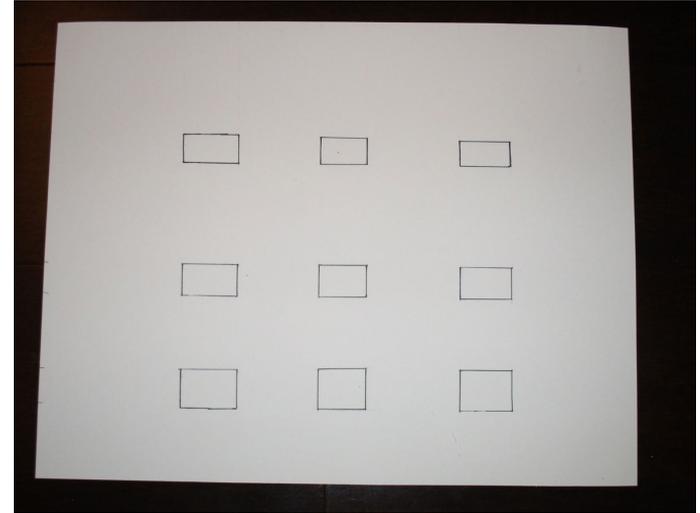
- Provide image processing capabilities to Python interpreter
- Open, close, and manipulate various image file types
- Important PIL Modules:
 - **Image** - represents a PIL image
 - **ImageOps** - provides image processing operations
 - **ImageChops** - provides channel operations (arithmetical image operations)

Numerical Python (NumPy)

- Provides multidimensional array and matrix data structures
- Provides methods to create, manipulate, and perform mathematical operations on arrays
- Used to represent the layout of the classroom
- Convert PIL image to NumPy array
- NumPy array slicing to create smaller multidimensional arrays

Limitations and Constraints

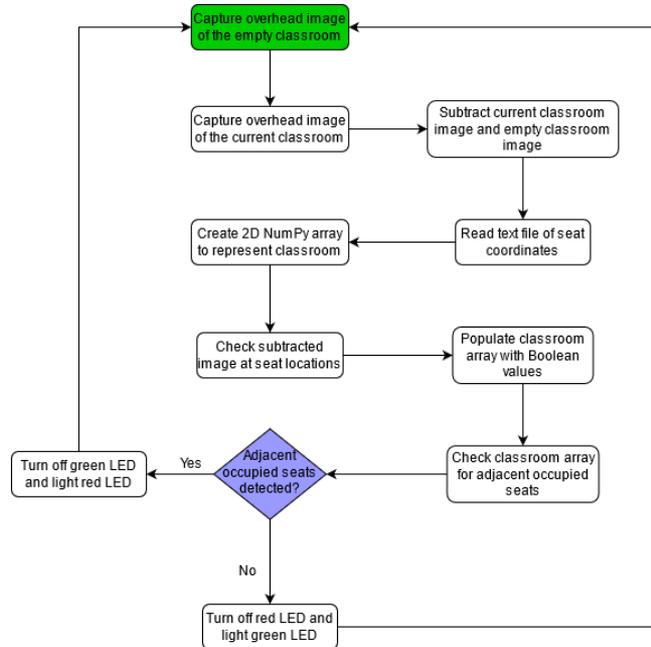
- Absence of a real-world classroom environment
 - Classroom is simulated with simple two-dimensional drawing and toy figures
- System Constraints:
 - Stationary seat locations
 - Designed for classrooms with simple row and column layout
 - Does not account for external factors such as personal belongings

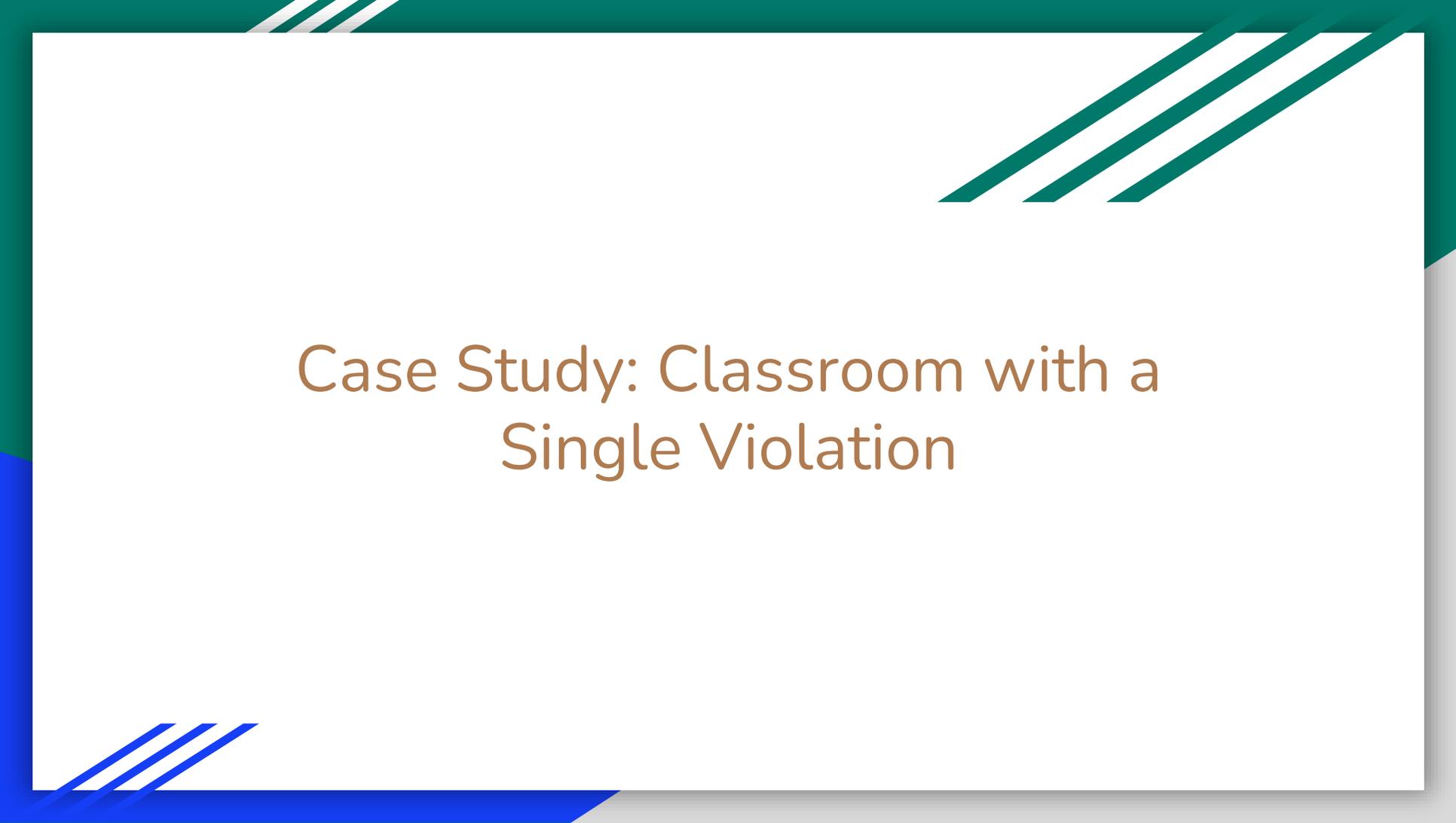




Solution and Implementation

Algorithm Flow

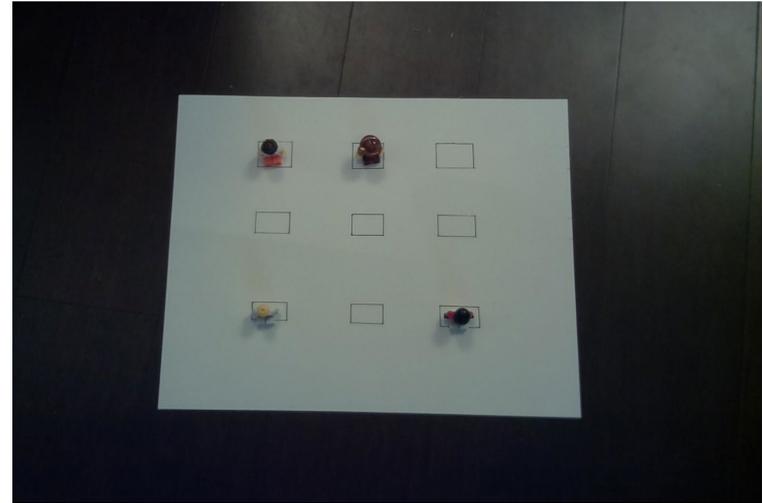




Case Study: Classroom with a Single Violation

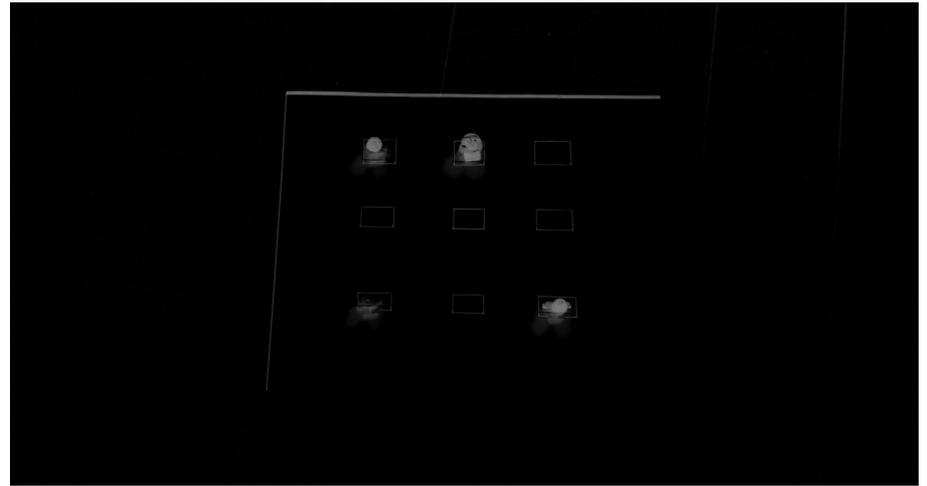
Image Capture and Processing

- Overhead image of the classroom is captured using PiCamera and Raspberry Pi Camera Module
- Open image files with PIL
- Convert images to grayscale with PIL
- Subtract current classroom image and empty classroom image



NumPy Arrays and Occupancy Detection

- Convert subtracted image to NumPy array
- Read seat coordinate text file and establish array dimensions
- Slice array at seat locations and compute sum for each seat
- Determine if sum is within threshold value and set occupancy flag to 1 or 0
- Reshape array to reflect classroom layout

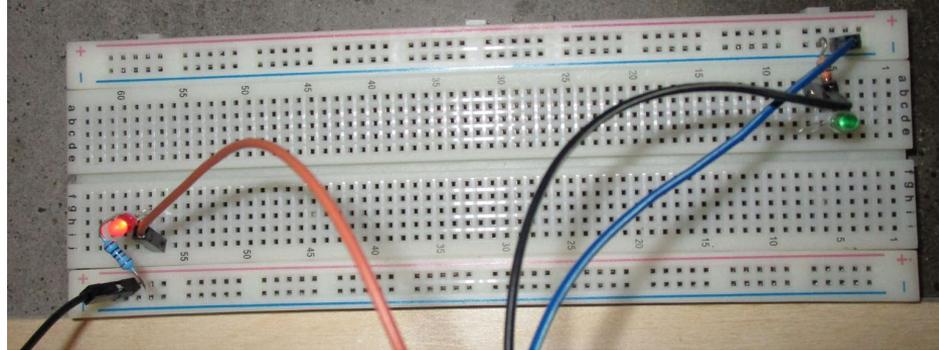


Detecting Adjacency

- Iterate through the 2D array, checking for occupied seat (denoted by a value of 1)
- If seat is occupied, check for an occupied seat in the next row and the next column
- If adjacent occupied seats detected, add seat indices to list of compromised seats
- Remove duplicates from list and retrieve the length of the list
- Turn on red LED if adjacent seats detected (list length is greater than 0)
- Otherwise, green LED is turned on

LED Output

- Red LED is turned on to reflect the seating violation.
- If the green LED was turned on prior, it is turned off before lighting the red LED.



Console Output

- The system displays a message to the console, informing the user of the seating violation.
- The number of compromised students is displayed as well.
- *Note:* the classroom array is only displayed for testing purposes.

```
[[1 1 0]
 [0 0 0]
 [1 0 1]]
Adjacent students detected. Number of compromised students: 2
```




Future Considerations

Recommendations for Improvement

Object Detection

- A computer vision and image processing technology that detects instances of objects in images and videos.
- Identifies which objects may be present in the image and provides information about the location of objects

Image Labeling and Classification

- Computer vision and artificial intelligence technologies that detect and analyze images to identify people, objects, and locations.

Benefits of the Recommendations

- Fewer input files would be required by the end-user
 - Determine seat locations without a text file of seat coordinates
 - Classroom layout can be changed without the need for an updated coordinate file
- The system would be better equipped to handle real-world scenarios
 - Image processing models can be trained to differentiate between students and other objects
 - Reduces the likelihood of false violation reports
 - Addresses the issue of non-stationary seats

Conclusion

- The solution performs successfully under all test case scenarios
- Current implementation is only suitable for specific classroom layouts with completely stationary seats
- Significant limitations make the system unsuitable for real-world applications
- Recommendation: Replace the current image processing method with a trained object detection or image recognition model
 - Eliminates need for seat coordinate input file
 - Allows the system to distinguish between human and non-human objects
 - Automatic seat detection eliminates the need for stationary seats