

Detection of Hidden Spy Cams in the Public/Private Areas using IoT based Mobile Application

Sarikonda Meghana¹, Syed Abdur Rauf Magrabi²

¹Student, Department of Computer Science and Engineering, Sphoorthy Engineering College, Hyderabad, India

²Assistant Professor, Dept. of Electrical and Electronics Engg., Sphoorthy Engg. College, Hyderabad, India

Abstract: In this paper, we've have developed an IoT based Android mobile application that is used for probing the hidden spy cameras for the recording the person's activity in the cluttered environmental areas like trail rooms of shopping malls, the knob or bolt of a room door in hotel rooms. This application works only if there is an IR sensor that has been already built at the time of mobile making. Certain considerations or actions have been taken out through which we were able to provide significant results which went successful.

Keywords: IR Sensor, GPS tracker, GSM module, spy cameras, Android mobile, MIT App Inventor-2.

1. Introduction

With the advancement in the digital technology, the usage of mobile phones has so become vast that are allowed to use in the private areas or restricted venues like offices, conference halls, malls, lodges etc only for the purpose of their important and personal use. But even in-terms that , there are some places where they don't allow the people to get their mobile phones inside that area stating with an argument that "this is a private place and please submit your mobiles in the counter to avoid disturbance". There are some cunning natures in the society who're living only with a purpose "take advantage over the people and make it a business". Women life plays a major role in that when it comes to the privacy and security. In some areas, there are cameras placed in that area and no doubt if there is a min spy camcorder then everything will be detected. Camcorders or cameras at hidden places are the one that is very paramount these days. The data get exploited in the public that may threaten their life and also leading to their death.

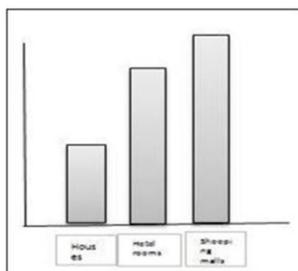


Fig. 1. Shows the analysis of placed cameras in different cluttered areas

Few actions were taken into the considerations but not as many. Hence during this thought process, we've designed and developed an IoT based android application where the sensor senses, detects and tries to locate the camera. Few tests were taken out using teh case that made the app work successfully.

2. Block diagram

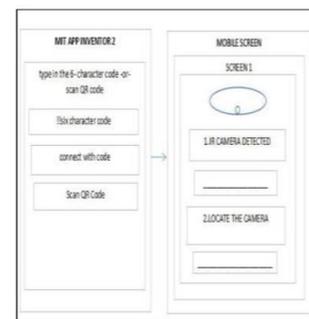


Fig. 2. Block diagram of the SPY CAM DETECTOR app

The above schematic block represents the mobile IoT application. On the mobile screen, there is a magnetometer scanning button and down to that there is a detection of IR Scanning button. The purpose of magnetometer scanning is that it detects only normal camera from where we're checking in through and there is one more button detect IR Camera that detects the hidden soy camera. Next to that, there is one more button "Locate the Camera" which detects the camera location through GPS Tracker and GSM Module. This app can built in two ways (1) by ANDROID STUDIO 3.1 with kotlin 1.2.50 or (2) by MIT App Inventor 2(taking help of MI2 AI companion). We've built our Product through the support of MIT App Inventor 2 as it is more compatible and it doesn't require any coding part and little time to build the app. Here it is shown how to style a subsection and sub sub-section also

3. Software design

A. Hardware components required for software development

Type the web address refer [2] in the address search bar of the Internet Explorer or other internet supporting web servers

like Google Chrome or Mozilla Firefox or Opera browser. The above website redirects to your Google Mail i.e. Gmail where we need to sign up and accept the policies or terms and conditions. Next it opens a tab where it displays few options like Projects, Connect, Build, Help, and My Projects in the header bar. Click on the Project button, it displays the above options.

Table1

Shows the basic components required for building the application

No.	Components
1	Laptop/Desktop connected to a Wi-Fi or Internet.
2	Android Mobile(Installed with MIT AI2 Companion)

Table 2

Shows the options of the project bar



Click on ‘Start New Project’ it displays where we need to give the title to the Project as “SpyCamDetector” and spaces in between the name should not be given while naming the app.



Fig. 3. Key store

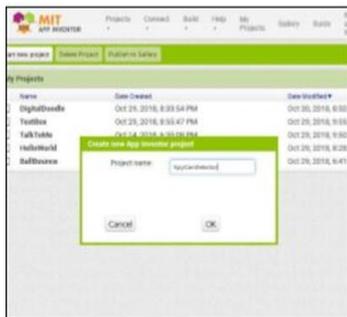


Fig. 4. Shows the Creation of Spy Cam Detector App

It directs to the page where a palette is placed on the left side of the tab. The palette consists of UI, Layout, Media, Drawing and Animation, Maps, Sensors, Social, Storage, Connectivity, Experimental and Extensions. In this project we will be making

use of UI (For Buttons, Text box), Sensors (For location tracking and Near field), Media (For Text to Speech and Camera). Click on the UI and click on the Button box and drag it to the Screen. On Right side of that window tab, there are two tabs “SCREEN”, ”BUTTON”, ”TEXT TO SPEECH”, ”NEAR FIELD” and “LOCATION SHARING”. Click on the Screen option and go to the last right tab to change the title of the Screen and Name it as Spy Cam Detector and Click on the Button field to change the name of the button and here we use four buttons. Note that we can different color variants to change the background color of the screen and even we can make use of Canvas button for making the app more compatible and canvas play a role where components like Buttons, Textbox can be dragged to the center. Change the Button1 name as -->’ Detect the Camera by Radiation Meter’. Next Change the Button2 name as -->’Detect the Location’. If the cameras are detected two more buttons are added for giving the person a voice message that a camera is detected which are invisible components.

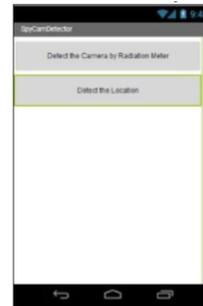


Fig. 5. Shows the designing screen of the app

Here the TextToSpeech, Camera, LocationSensor1 and NearField are invisible components.

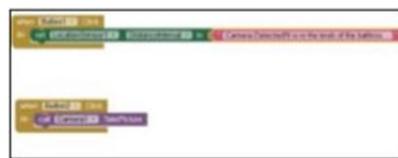


Fig. 6. Shows the block viewer of the app



Fig. 7. Shows the compilation part of the app

Fig. 5. Now connect the app to the mobile. On the header bar of the web page, besides PROJECT there is are two more options Connect or Build. In Connect option, we can connect

our app to our mobile through MIT AI2 Companion installed in our mobiles through AI Companion, USB or Emulator. Here I have connected my mobile through AI companion and it gave the QR code where we've scanned it and connected to the app to my mobile but it doesn't install the app. Hence I have moved for other option called Build where I can install this app by connecting my mobile to this website screen through a simple QR Code through a simple command "App(provide QR code for .apk).

After the compilation, it provided the result with 0 errors and the QR code is displayed on the screen and the QR code provides a link where we need to install the app.



Fig. 8. Shows the website link to install the app from MIT AI2 Companion app

After clicking on the button 'Scan QR Code' , it provides us the website link where we should install the app. Click on the button 'Connect with Code' the application gets installed and the screen of the app will be displayed same as shown in the Fig. 3.

4. Tests and result

Click on the Button 'Detect the Camera by Radiation Meter' and once if the app is detected it produces the following results as shown in the figures below.

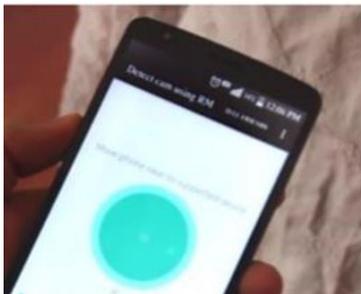


Fig. 9. Shows the detection of the camera

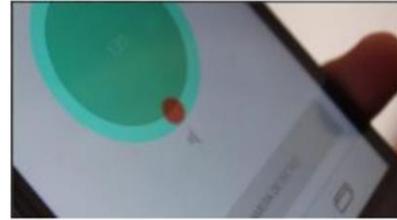


Fig. 10. Shows after the camera is detected in the location

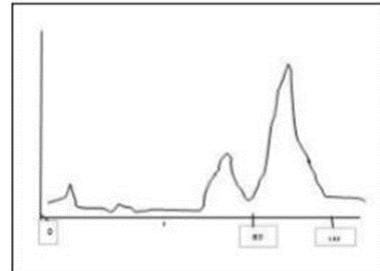


Fig. 11. Shows the graph for detecting the camera if the magnetometer speed expels beyond the range (80-120)



Fig. 11. Shows the placed camera through Mobile

In the Fig. 8. The detected camera location shows the suspected camera and the camera is spotted through beep and visual red spot when the magnetic activity is similar to the camera.

5. Conclusion

This paper surveys as an approach for detecting the cameras in the cluttered environments and can also be detected when lights turned off using flashes (an old practical method) for getting prevented victim from the Scandal issues. This application provides an assurance for maintaining the security to the top level extent in our lives.

References

- [1] Tian Liu, Ziyu Liu, Jun Huang, Rui Tan,Zhen Tan: Detecting Wireless Spy Cameras Via Stimulating and Probing.
- [2] <http://ai2.appinventor.mit.edu/>
- [3] K. Parvateesam, G. A. Arun Kumar: Presence of Active Mobile Phones and Hidden Camera Detection, International Journal of Science, Engineering and Technology Research (IJSETR), Volume 6, Issue4, April 2017.
- [4] Murtaza Taj, Andrea Cavallaro: Multi-Camera Scene Analysis using an Object-Centric Continuous Distribution Hidden Markov Model, 2007 IEEE International Conference on Image Processing.
- [5] Chien-Cheng Lee ,Yi-Fang L: Computer Vision Techniques for Hidden Conditional Random Field-Based Mandarin Phonetic Symbols I Recognition, 2011 International Conference on Broadband and Wireless Computing , Communication and Applications.

- [6] Timothy Callemeyn, Wiebe Van Ranst, Toon Goedemé: The autonomous hidden camera crew, 2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA).
- [7] Brent Lagesse , Kevin Wu, Jaynie Shorb, Zealous Zhu- Detecting Spies in IoT Systems using Cyber-Physical Correlation, 2018 IEEE 28th Canadian Conference on Electrical and Computer Engineering (CCECE).
- [8] Brent Lagesse, Kevin Wu, Jaynie Shorb, Zealous Zhu- Automated Hidden Sensor Detection in Sensor-Rich Spaces, 2018 IEEE 28th Canadian Conference on Electrical and Computer Engineering (CCECE).