# Abandoned Bag Detection

Submitted in partial fulfillment of the requirements of the degree

**BACHELOR OF ENGINEERING** IN **COMPUTER ENGINEERING**

By

**Junaid Basha 120A1031**

**Indrajit Karmakar 120A1037**

**Mayank Iyer 120A1026**

**Dinesh Prabhakar 120A1016**

Supervisor

**Prof. Namrata Patel**



**Department of Computer Engineering**

**SIES Graduate School of Technology Nerul, Navi Mumbai - 400 706**

**University of Mumbai**

**(AY 2021-22)**

# CERTIFICATE

This is to certify that the Mini Project entitled **“ Abandoned bag detection application ”** is a bonafide work of **Junaid Basha(120A1031), Indrajit Karmakar(120A1037), Mayank Iyer(120A1026), Dinesh Prabhakar(120A1016)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **“Bachelor of Engineering”** in **“Computer Engineering” .**

### (Prof. Namrata Patel)

Supervisor

### ( Dr. Aparna Bannore ) (Dr. Atul Kemkar )

Head of Department Principal

# Mini Project Approval

This Mini Project entitled “Abandoned bag detection application**”** by **Junaid Basha(120A1031), Indrajit Karmakar(120A1037), Mayank Iyer(120A1026), Dinesh Prabhakar(120A1016)** is approved for the degree of **Bachelor of Engineering** in **Computer Engineering.**

## Examiners

**1………………………………………**

(Internal Examiner Name & Sign)

### 2…………………………………………

(External Examiner name & Sign)

Date: Place:

# Contents

### Abstract ii

### Acknowledgments iii

### List of Abbreviations iv

### List of Figures v

### List of Tables vi

### List of Symbols vii

### Introduction 1

* 1. Introduction
  2. Motivation
  3. Problem Statement & Objectives
  4. Organization of the Report

### Literature Survey 11

* 1. Survey of Existing System
  2. Limitation Existing system or research gap
  3. Mini Project Contribution

### Proposed System (e.g. New Approach of Data Summarization) 18

* 1. Introduction
  2. Architecture/ Framework
  3. Algorithm and Process Design
  4. Details of Hardware & Software
  5. Experiment and Results
  6. Conclusion and Future work.

### References 32

**ACKNOWLEDGEMENT**

We would like to express our thanks to the people who have helped us the most throughout our project. We are grateful to our guide Prof. Namrata Patel and coordinator Dr. Rizwana Shaikh for nonstop support for the project.

A special thanks goes to each other who worked together as a team in completing the project, where we all exchanged our own interesting ideas, thoughts and made it possible to complete our project with all accurate information. We also wish to thank our parents for their personal support and attention who inspired me to go my own way.

We would also like to extend our sincere gratitude to our Head of the Department Dr. Aparna Bannore and our principal Dr. Atul Kemkar for their continuous support and encouragement.

We also would like to thank our other faculty members for providing us with all the required resources and references for the project.

**CHAPTER 1**

* 1. **Introduction**

Terrorist attacks have claimed the lives of thousands of innocent people over the last few decades. With the rise of sophisticated methods of terrorism, it is the need of the hour for special technological advancements for terrorist bombing prevention. Traditional methods of prevention like the use of manpower and manual surveillance are no longer enough. Advancements in technological video surveillances in public places is needed. Visual surveillance has attracted more and more researchers in the past because of its tremendous application prospects. But only setting up surveillance cameras isn’t enough if the events being surveyed can’t be analyzed in real time, a feat impossible for humans in many cases. In most of the cases of this kind, the video that had been recorder has to be analyzed for the sake of restructuring that particular even after said event has occurred. We need a system for smart surveillance that works in real time. We have worked for the sake of development of a system that will make the video surveillance more versatile and reliable. In general, an object that has been left unattended can be called abandoned. Detection of abandoned objects is very important in public places like airports, shopping malls, railway stations, bus stops etc. As surveillance and CCTV cameras become increasingly popular, this is the right time for us to come up with soft wares to make them more efficient and useful for mankind. A very big complication in the detection of abandoned objects is the classification of an identified object as being abandoned. Our software detects if an object has been abandoned by the distance it has with the person who left it and runs a code to alert the authorities about it. In our project, the OpenCV tool is used using the PyCharm IDE. When a bag is left in the area under surveillance, the system keeps a lookout for the position of the bag and the person whose presence in the scene defines the status of the bag. If the system notices that a bag has been abandoned, it will immediately notify the security personnel.

* 1. **Motivation**

Every year, a lot of money is spent on surveillance using human beings and putting them in potentially dangerous situations for the sake of protection of innocent lives. This many a times leads to the loss of indispensable lives of human beings. At the end of the day, human beings came up with computers and computer applications for making their lives easier and generally safer. We are playing our part in this noble human endeavor of saving lives through this application. This has been the motivation behind our project all along.

* 1. **Problem Statement and Objectives**

**Problem Statement:**

It is highly inefficient to still be relying largely on manual manpower for surveillance and prevention of danger. We spend a lot of time, energy and sometimes even lives for the prevention of and protection from terrorist bombings. But if/when the attack happens, many a times it is too late and many lives will have been lost by then. It costs lives and crores or rupees in property damage and fears among the people. We need a trustworthy system that isn’t limited like the capabilities of human beings. Our system aims to take human beings out of the surveillance equation. The application will survey for abandoned bags and notify the concerned authorities about it.

**Objectives:**

1.To make it easier for the detection of bags that can potentially contain harmful weapons.

2.To make it more convenient for security personnel to work efficiently to save lives.

3.To prevent the occurrences of large-scale property damage due to terrorist attacks.

* 1. **Organization of Project**

In chapter 1, we have given a brief introduction of our project and the circumstances under which it is supposed to help. Then we have made an effort to explain the motive behind doing this project and what, problems it is supposed to fix. Then we have defined the problem statement and the objectives i.e., describing in detail what the problem is and what and how we hope to fix it. Chapter 2 deals with the literature survey part of the report. It includes first, a survey of the existing system and some research papers that support our theory. Then we make clear the limitations of the existing system i.e., why we decided to do the project. Then we give a brief introduction on what we aim to do with the project and our contribution to society with it. Then comes chapter 3, where we explain the project in as much detail as possible without the reading getting too tedious. We start with the working and general flow of the program along with some screenshots of the implementation. Then we end with the conclusion and scope for future.

**Chapter 2**

**Literature survey**

**2.1 Survey of existing system**

In terrorist attack detection using purely human manpower, the loss of human lives in case of uneventful occurrences are inevitable. Human beings do not have the ability to perform surveillance 24/7 while a computer can. Great care has to be taken by the human being against various parameters to make sure that the bag is abandoned. Our application fixes these shortcomings and gives a comparatively better solution.

According to research paper [1] In this study we investigated the amount of risk in terms of the range of possible productivity that can be expected in a portfolio of software development. Working on the large database with 4106 projects from across the world, this study revealed that inefficient development teams can spend as many as 13 times of the effort taken by proficient teams for the development. This study further examined the effects of project size and team size on the variances in software development. The results showed that to reduce the risks of cost overruns, a small project is better than a large project when the team size is chosen in advance, and a small team is preferable to a large team for a project of a certain fixed size.

According to research paper [2] This paper identifies debugging as tedious, time consuming, and very expensive. It notes that despite all efforts and the scientific progress made, modern software still contains bugs that not only cause pure inconveniences, but also have a negative impact on the economy. Thus, we still need techniques that help us debug software systems. Both debugging techniques of reproducing the failure and finding the defect can be a tough and risky challenge. Automated debugging aims to ease the search for failure causes. Most software development companies spend a huge number of resources in testing and debugging. A lot more research need to be conducted to fully automate the debugging process thereby reducing software production cost, time and improve quality.

According to research paper [3] Techniques to predict the performance and functional behavior of applications are important in the design of many tools. When these tools take into account application input and configuration parameters, they are able to achieve high levels of predictive accuracy but face the challenge that they can only be trained on a small fraction of the overall space of inputs and configurations. Since this means that in most production runs the models will need to make predictions about configurations that are well outside the space on which they were trained, it is necessary for such models to quantify their prediction errors across the entirety of this space. In this paper, we present a systematic approach to quantify such prediction errors and demonstrate the utility of our approach via a practical use case of an anomaly detector. This detector calibrates its alarm thresholds based on the error estimates provided by our technique. Our experimental evaluations confirm that this tool achieves a low false positive rate while maintaining a high detection accuracy compared to a fixed threshold-based anomaly detector that do not use our prediction error characterization-based technique.

According to research paper [4] a study of job failures and performance issues in a large university-wide community cluster (Conte) was presented. To the best of our knowledge, this is the first failure analysis of a production community cluster based on real failure data. We found that, due to the wide diversity of research domains and user experience levels, community clusters have an interesting mix of user errors and application errors. More specifically, we found that the largest failure categories were those related to inaccurate estimation of job runtime and memory issues. Even though these two failure types appear independent, we found strong correlation between memory issues and jobs exceeding wall time. We also presented two lightweight techniques for improving cluster reliability: (1) a fine-grained classification technique for mapping jobs to various application groups, and (2) statistically predicting potential buggy libraries based on job failure data. We also presented some of the affected users with suggestions for alleviating these issues. Our future work would involve a thorough analysis of the collected syslogs and resource usage data (such as network usage and Lustre IO metrics) to find more fine-grained reasons for failures and performance issues.

According to research paper [5] Discovering and fixing software bugs is a difficult maintenance task, and a considerable amount of effort is devoted by software developers on this issue. In the world of soft ware one cannot get rid of the bugs, fixes, patches etc. each of them have a severity and priority associated to it. There is not yet any formal relation between these components as both of these either depends on the developer and tester or on customer and project manager to be decided on. On one hand, the priority of a component depends on the cost and the efforts associated with it. While on the other, the severity depends on the efforts required to accomplish a particular task. This research paper proposes a formula that can draw a relationship among severity and priority.

According to research paper [6] This paper analyzes the vulnerabilities of six web applications using their past 655 security fixes as the field data. Results show that only a small subset of 12 generic software faults is responsible for all the security problems (XSS and SQL injection). We found considerable differences by comparing the distribution of the fault types of our results with studies of common software faults. We also detected that one of the Missing Function Call fault types (MFC extended) is responsible for 76% of all the security problems analyzed. The fault types are thoroughly detailed providing enough information for the definition of vulnerability fault models that can be used by researchers of realistic attack injectors. For future work we intend to analyze the exploit code used to attack web applications. With these results it will be possible to build a realistic attack injector for web applications. Other studies could be done, following the same methodology presented here but aimed at vulnerabilities in operating systems and their applications, searching for a common pattern

**2.2 Limitations of existing system**

In this type of system, it becomes difficult to track the live distance between the owner and the bag. If the distance between them is overlooked, and only stationary items are tracked, there is a risk of creating nuisance for unsuspecting innocent people. In such a case whatever money has been spent on the application will be practically useless

One will have to go on the site with all sorts of protective gear every time a stationary bag and human being is detected. It will be more inefficient and dangerous if there is absolutely no use of sophisticated technology that does not get tired and we only depend on human manpower for the purpose. In order to have a safer and smoother detection process, we need an application that works real time and also detects moving objects

|  |  |  |  |
| --- | --- | --- | --- |
| **Paper Title** | **Journal** | **Problem Addressed** | **Gaps** |
| Detecting abandoned baggage items in a public space. | Kevin Smith, Pedro Quelhas, and Daniel Gatica-Perez IDIAP Research Institute & Ecole Polytechnique F ´ ed´ erale de Lausanne (EPFL) ´ Switzerland | A two-tiered solution to the left luggage problem, wherein a detection process uses the output of an RJMCMC tracker to find abandoned pieces of luggage. | Provides average tracking for a single camera view. Performs worse when there are more camera views. |
| Real-time detection of abandoned bags using CNN | The Federal State Unitary Enterprise ”State Research Institute of Aviation System”, Viktorenko street, 7, Moscow, Russia | Proposal for approach of abandoned bag detection based on Gaussian mixture model. | The proposed system works on high end devices with powerful GPUs. Not on general computers and microcomputers. |
| Detecting Suspicious Background Changes in Video Surveillance of Busy Scenes | D. Gibbins, G.N. Newsam and M.J. Brooks Centre for Sensor Signal and Information Processing Signal Processing Research Institute Technology Park, Adelaide, SA 5095, Australia | A technique has been outlined for solving the problem of background change detection | It is still a work in progress. The algorithm is inefficient and still being refined. |
| Smart Human Security Framework Using Internet of Things, Cloud and Fog Computing | Vivek Kumar Sehgal, Anubhav Patrick, Ashutosh Soni and Lucky Rajput. | Due to population explosion, it has becoming increasingly challenging for civil  authorities to provide security cover to citizens. In order to tackle this enormous  challenge, we have proposed a physical security framework that employs the con-  cept of IoT, fog computing and cloud.  Due to population explosion, it has becoming increasingly challenging for civil  authorities to provide security cover to citizens. In order to tackle this enormous  challenge, we have proposed a physical security framework that employs the con-  cept of IoT, fog computing and cloud.  Due to population explosion, it has becoming increasingly challenging for civil authorities to provide security cover to citizens. In order to tackle this enormous challenge, we have proposed a physical security framework that employs the concept of IoT, fog computing and cloud. | No solution has been provided for detection of terrorist attacks. |
| Research on Abandoned and Removed Objects Detection Based on Embedded System | Bin Hu, Yiyue Li, Zizhang Chen, Gang Xiong, Fenghua Zhu | This paper proposes a robust and accurate algorithm to detect abandoned objects. | It can only detect stationary objects, so cannot measure the live distance between the human and the object. |

**2.3 Mini project contributions**

So, our mini project aims to cover some of the shortcomings of the currently present system such as we track the live position of the human with respect to the bag. It will be very simple and easy to use the application. We just have to launch it and it will automatically use the already trained model and then detect the images pixel by pixel to find the particular models in real life. It will measure the distance and as soon as the distance between the human being and the object increases to a certain point, it will immediately alert the authorities.

**Chapter 3**

**Proposed System**

**3.1 Abandoned bag detection application**

Surveillance and prevention of terrorist attacks is a dangerous and life-threatening task to accomplish. Most of the time, the video footage is analyzed after the unfortunate event has occurred. This is because the huma body can only accomplish so much. We have our limitations. The same is not true for technologies. Their purpose is to make our lives easier and safer. We have created this application for real time detection of abandoned bags in public places. The application doesn’t need to rest like us humans do and keep surveillance 24/7.

**3.2 Architecture/Framework**

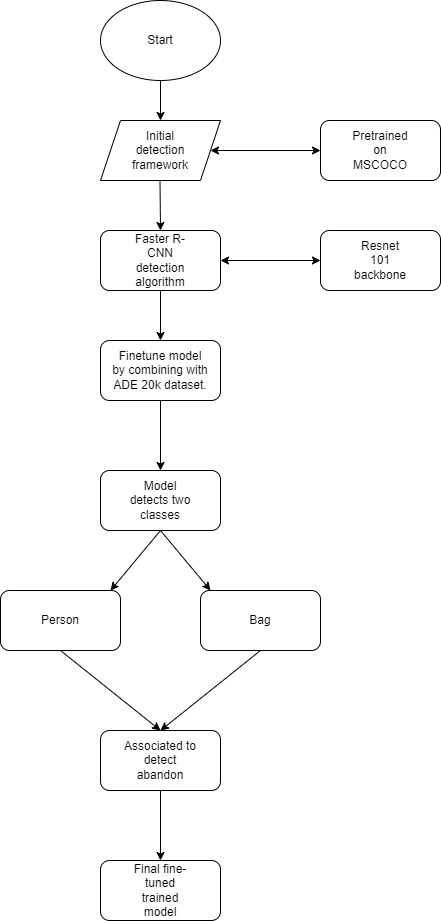


Fig 3.2.1 Flowchart

The application is made with the goal of detecting abandoned objects that might contain weapons of destruction, removing them and catching the perpetrators as easy as possible. The application does most of the work and hence makes it easier for the human beings to work comparatively more safely. The application uses models trained in coco (common objects in context) for this purpose. It is trained for human being and bag models.

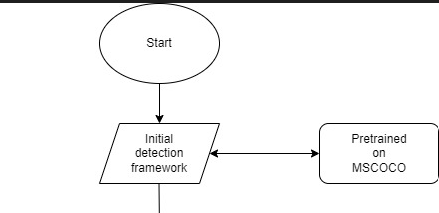


Fig 3.2.2 Detection using trained coco models

Once the model trained on coco has been identified, fast R- CNN (Region-based Convolutional Neural Network) algorithm is used to detect the object. Instead of feeding the region proposals to the CNN, we feed the input image to the CNN to generate a convolutional feature map. From the convolutional feature map, we identify the region of proposals and warp them into squares and by using a Roi pooling layer we reshape them into a fixed size so that it can be fed into a fully connected layer. From the Roi feature vector, we use a SoftMax layer to predict the class of the proposed region and the offset values for the bounding box.

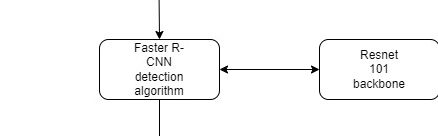


Fig 3.2.3 Detection using R-CNN

The model is then more finetuned and made more clearer for the purpose of identification by combining it with the ADE 20k dataset. This particular dataset contains more than 20 thousand scene centric images with pixel-level objects and objects parts labels.

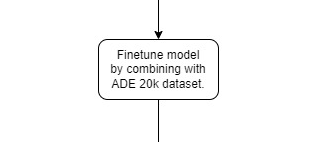


Fig 3.2.4 Finetuning using ADE 20K dataset

Using the final model, the application needs to detect 2 classes. One is of a human and the other of a bag. Then it needs to associate them together to understand the situation. Then it will measure the distance between the bag and the said human. If the human creates a big distance between him and the bag or if the huma completely disappears from the view after only letting the bag stay in the view, the application will notify the personnel in whatever way the application has been programmed to notify them.

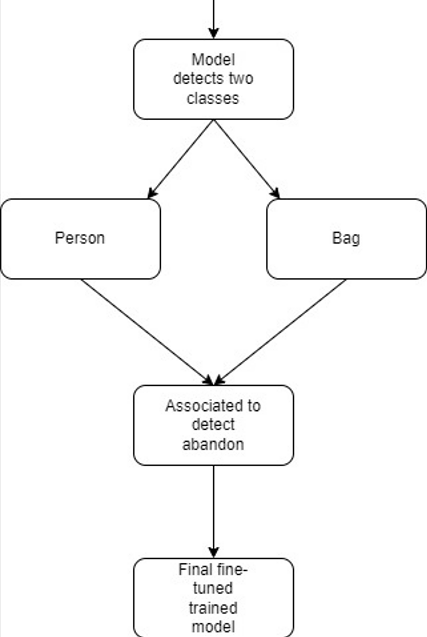


Fig 3.2.4 Detection and notification

**3.3 Algorithm and Process Design**

According to the flowchart, as soon as the application is launched, it loads up the initial framework and gets the model from coco. It then starts analyzing the live video footage in real time using fast R-CNN algorithm. It then uses the ADE 20K dataset for more finetuning of the already detected model. Once that is done, it works towards associating different models together. It tries to associate the bag and human class together. Once that has been done, it will then measure the distance between these 2 entities. If the human being goes away by a certain distance, the application will alert the concerned authorities.

* 1. **Details of Hardware and Software**

Hardware:

1)Laptop/PC with at least a Pentium processor

2)Laptop/PC with Windows 10/11.

Software:

1)Python

2)Pycharm IDE

3)OpenCV library

4)PyTorch and Torchvision library

5)Git Bash for basic shell scripting

* 1. **Experiment and Results**

**3.6 Conclusion and future work**

This project proposed an application that can essentially replace humans in terrorism attacks prevention to a large degree. It will help preserve the indispensable human life. A method for efficiently detecting abandoned bags that might contain dangerous weapons is provided. Ways for error free surveillance that works 24/7 without getting tired is provided. It will measure the distance between the person and the bag. If the distance increases up to a certain point, then the application will notify the concerned authorities.

**References**

[1] Comstock, C., Jiang, Z., & Naudé, P. (2008, August). Risk analysis in software development. In Proceedings of the 8th conference on applied informatics and communications, AI’C08, Stevens Point, Wisconsin.

[2] Wambugu, G. M., & Mwiti, K. (2017). Automatic Debugging Approaches: A literature Review.

[3] Mitra, S., Bronevetsky, G., Javagal, S., & Bagchi, S. (2015, October). Dealing with the unknown: Resilience to prediction errors. In 2015 international conference on parallel architecture and compilation (PACT) (pp. 331-342).IEEE.

[4]Mitra, S., Javagal, S., Maji, A. K., Gamblin, T., Moody, A., Harrell, S., & Bagchi, S. (2016, October). A study of failures in community clusters: The case of conte. In 2016 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW) (pp. 189-196). IEEE.

[5]Soner, S., Jain, A., Tripathi, A., & Litoriya, R. (2010, June). A novel approach to calculate the severity and priority of bugs in software projects. In 2010 2nd International conference on education technology and computer(Vol. 2,pp. V2-50). IEEE.

[6]Fonseca, J., & Vieira, M. (2008, June). Mapping software faults with web security vulnerabilities. In 2008 IEEE international conference on dependable systems and networks With FTCS and DCC (DSN) (pp. 257-266). IEEE.

|  |
| --- |
|  |
|  |
|  |  |

**Web References**

1. Python Documentation

https://docs.python.org/3/

1. PyTorch and TorchVision Documentation

https://pytorch.org/docs/stable/index.html

1. OpenCV Documentation

https://docs.opencv.org/4.x/

1. Cuda Toolkit Documentation

https://docs.nvidia.com/cuda/

1. Detectron2 Documentation

https://detectron2.readthedocs.io/en/latest/

1. Object Detection using Faster R-CNN Documentation

https://learn.microsoft.com/en-us/cognitive-toolkit/object-detection-using-faster-r-cnn