STUDENT COUNTER USING FACE DETECTION ALGORITHMS

A Major Project Report Phase-1

Submitted To

Chhattisgarh Swami Vivekanand Technical University
Bhilai, India

For

The Partial Fulfilment of Degree

of

Bachelor of Technology

in

Computer Science & Engineering

By

Vishal Rathi	**Tushar Dubey**	**Kanha Agrawal**
Roll No.- 303302219121	**Roll No.- 303302219112**	**Roll No.- 303302219050**
En. No.-BH3804	**En. No.-BH3795**	**En. No.-BH3733**
Semester 7th (CSE)	**Semester 7th (CSE)**	**Semester 7th (CSE)**

Under the Guidance of

DR. J.P. PATRA

Professor & Head

Department of Computer Science & Engineering

S.S.I.P.M.T, Raipur

Department of Computer Science & Engineering

Shri Shankaracharya Institute of Professional Management & Technology Raipur (C.G.)

SESSION 2022-23

Department of Computer Science & Engineering
Shri Shankaracharya Institute of Professional Management & Technology Raipur (C.G.)

STUDENT COUNTER USING FACE DETECTION ALGORITHMS

DECLARATION BY THE CANDIDATE

We the undersigned solemnly declare that the report of the Major project Phase 1 report work entitled ***"STUDENT COUNTER USING FACE DETECTION ALGORITHMS"***, is based on our own work carried out during the course of our study under the supervision of **Dr. J.P. Patra** (Professor & Head of CSE Department).

We assert that the statements made, and conclusions drawn are an outcome of the project work. We further declare that to the best of our knowledge and belief that the report does not contain any part of any work submitted for the award of any other degree/diploma/certificate in this University/deemed University of India or any other country.

(Signature of Candidate 1)	(Signature of Candidate 2)	(Signature of Candidate 3)
Name: Vishal Rathi	**Name: Tushar Dubey**	**Name: Kanha Agrawal**
Roll No.:303302219121	Roll No.: 303302219112	Roll No.: 303302219050
En. No.: BH3804	En. No.: BH3795	En. No.: BH3733
Semester: 7th	Semester: 7th	Semester: 7th

CERTIFICATE OF THE SUPERVISOR

This is to certify that the Major project phase-1 on report entitled "**STUDENT COUNTER USING FACE DETECTION ALGORITHMS**" is a record of project work carried out under my guidance and supervision for the partial fulfilment of the award of degree of Bachelor of Technology in the faculty of Computer Science & Engineering of Chhattisgarh Swami Vivekananda Technical University, Bhilai (C.G.) India.

To the best of my knowledge and belief the report

i) Embodies the work of the candidate himself

ii) Has duly been completed

iii) Fulfils the partial requirement of the ordinance relating to the B-Tech. degree of the University

iv) Is up to the desired standard both in respect of contents and language for being referred to the examiners.

(Signature of the Supervisor)
Dr. J.P. Patra
(Professor & Head of CSE Department)
S.S.I.P.M.T, Raipur (C.G.)

Forwarded to
Chhattisgarh Swami Vivekanand Technical University, Bhilai

(Signature of HOD)
Dept. of Computer Science & Engineering
S.S.I.P.M.T, Raipur, C.G

(Signature of the Principal)
Dr. Alok Kumar Jain
S.S.I.P.M.T, Raipur, C.G

CERTIFICATE BY THE EXAMINERS

The Thesis/Project entitled "***STUDENT COUNTER USING FACE DETECTION ALGORITHMS***" Submitted by: (Roll No.: 303302219121, 303302219112, 303302219050. Enrollment No.: BH3804, BH3795, BH3733) has been examined by the undersigned as a part of the examination and is hereby recommended for the award of the degree of Bachelor of Technology of Chhattisgarh Swami Vivekanand Technical University, Bhilai.

Internal Examiner **External Examiner**

Date: **Date:**

ACKNOWLEDGEMENT

We thank the almighty for giving us the courage and perseverance in completing "Student Counter using Face Detection Algorithms". This project itself is the acknowledgment of all those people who have given us their heartfelt cooperation in making this project a grand success. We extend our sincere thanks to **Mr. Nishant Tripathi** Chairman of our college, for providing sufficient infrastructure and a good environment in the College to complete our course.

We are thankful to our principal **Dr. Alok Jain** for providing the necessary infrastructure and labs and also permitting us to carry out this project. With extreme jubilance and deepest gratitude, we would like to thank the Head of the CSE Department **Dr. J.P. Patra** for his constant encouragement.

A Special Thanks to our Project coordinator **Mr. Anand Tamrakar**, Assistant Professor, CSE Department, for his support and valuable suggestions regarding project work. We are greatly indebted to project guide **Dr. J.P. Patra**, HOD, CSE Department, for providing valuable guidance at every stage of this project work. We are profoundly grateful for the unmatched services rendered by him. Our special thanks to all the faculty of Computer Science and Engineering and peers for their valuable advice at every stage of this work.

(Signature of Candidate 1)	(Signature of Candidate 2)	(Signature of Candidate 3)
Name: Vishal Rathi	**Name: Tushar Dubey**	**Name: Kanha Agrawal**
Roll No.: 303302219121	Roll No.:303302219112	Roll No.:3033022190570
En. No.: BH3804	En. No.: BH3795	En. No.: BH3733
Semester:7th	Semester:7th	Semester: 7th

ACKNOWLEDGEMENT –AICTE IDEA Lab

We have taken efforts in this project. However, it would not have been possible without the kind support and help of AICTE-IDEA Lab at SSIPMT, Raipur. We would like to extend our sincere thanks to all the gurus, mentors and support staff of Idea lab.

ABSTRACT

There may be a need to count how many people are in a large room. If a person tries to count, there is a very high possibility of inaccurate data, more over this will take a lot of time. Hence to overcome this hectic work, we have a project that works on the principles of a tiny face detection algorithm. This project explores three phases: Capturing, Processing, and Display. In the first phase, we have an HD camera which is used to capture the image of the room via a microcontroller. The second phase consists of applying the pre-trained deep network to the image to detect tiny faces, and in the third phase of our project which is to display the number of people in a room.

List Of Symbols

Symbols	Names
,	Comma
.	Full Stop
’	Inverted Comma
( )	Parenthesis
:	Colon
-	Hyphen
“ ”	Double inverted comma
[ ]	Big Bracket

List Of Abbreviations

Abbreviations	Definitions
No.	Number
RAM	Random Access Memory
GUI	Graphical user interface
SDLC	Software Development Life Cycle
RAD	Rapid Application Development
SRS	System Requirement Specification
DFD	Data Flow Diagram
ResNet	Residual Networks

List of Figures

Table of Contents

CHAPTER - I
Introduction

1. Introduction About Project

Student Counter helps to count the number of students sitting at that particular instance in the classes. The system will generate the count of students and display the result at the required place. Before this, there wasn't any technology that counted the students in a classroom, hence the process was done manually by any person. He/she has to visit every classroom and has to either count the students sitting in that room or count the students through the attendance sheet. It was a very time-consuming process and required human resources which may lead to a high risk of error and sometimes disturb the running classes. Real-Time Tiny Face Detection is a practical method for dealing with a large number of student's attendance on a daily basis. Many algorithms and techniques have been developed to improve face recognition performance, but our proposed model employs Residual Networks (ResNet) to determine the positive and negative characteristics of the face, all of which are implemented in Python and the TensorFlow and OpenCV library. For user interface purposes, we use the pyQt5 GUI interface. The project deals with finding small objects (particularly faces in our case) in an image, based on scale-specific detectors by using features defined over a single (deep) feature hierarchy: Scale Invariance, Image resolution, and Contextual reasoning. The algorithm is based on "foveal" descriptors, i.e blurring the peripheral image to encode and give just enough information about the context, mimicking human vision. The subject is still an open challenge and we would like to experiment with this approach in different applications. Thus, after presenting the method and the influence of its parameters, we will concentrate on a real-world application which is counting people in a public demonstration. The last part will focus on possible extensions of this work. Our code (using Python and TensorFlow) is available at *https://github.com/vishal6789/Student-Counter*. As humans continue to make technological advances, the need to secure devices that house both our private and official matters is paramount. Some of the traditional methods achieved this feat by using ID cards, passwords, passphrases, and puzzles. However, with the rapid advancement in the field of deep learning and high-performance computing, the usage of human biometrics such as the face, voice, and fingerprints are deemed ubiquitous in modern-day security verification programs.

The widespread use of these human biometrics relates to their uniqueness, making it difficult to replicate them. Similarly, face detection programs allow a quick yet efficient framework for the detection of an individual. Face detection software can be seen in everyday devices like mobile phones and laptops and in physical security devices deployed in offices. Their success in accurately identifying faces is unprecedented. However, to achieve this step, pre-trained models such as FaceNet, ResNets, SesNets, and their variants in use in human face detection are trained on human faces. It is very difficult - if not impossible - to detect very small faces, even for a human user. Thus, we need to establish how best to encode context. For context modelling, the paper makes use of a fixed-size (291px) receptive field. It then defines templates over the “hypercolumn” features (vector of activations of all units for each pixel) extracted from multiple layers of a deep model which are effective ”foveal” descriptors. This technique allows for capturing both high-resolution detail and coarse low-resolution cues across large receptive fields. In order to count people only once, we have to recognize each face and then match them in the next frame. Obviously, we first needed to detect faces in each frame. To do so, we applied the Tiny Faces algorithm for the detection of all the faces, and hence, for each frame, we got a list of predicted detected faces. Then, we have to match them across frames. We created a face embedding for each face to make the matching easier: we wrapped each picture so the faces are always in the same direction. The face alignement is achieved by a pre-trained face landmarks estimation algorithm and affine transformations. To overcome the fact that we do not have many images of each face and in order to have a cheaper (in terms of time and memory) algorithm comparison, we used a pretrained CNN for face embedding (generating 128 basic measurements for each face).

CHAPTER - II

LITERATURE REVIEW & PROBLEM IDENTIFICATION

2.1 Literature Review & Problem Identification

In tiny face detection, the process flow is initiated by capturing images of classes in a frequent interval through a camera by using an IoT device (Raspberry Pi 4 / Nvidia Jetson Nano kit). Then the captured image will be sent to the server for computing the count face of face in the image. This will be done through an algorithm which processes the image captured and identifies the number of faces in the image by analysing from the learned or pre-trained model. Then the result of that Image will be sent to the required places where it will be displayed by GUI on a Screen.

There are times when an authority wants to know how many students are present in a classroom so that they can make any changes or pass any decision. Now, it's a hectic task to go to each and every class physically and count a number of students every time. It will not only disturb the running classes but also will be a not so efficient way. So we wanted to build a tool that can count a number of students in a classroom at any particular time and display the count to the authority seating at that cabin.

The motivation behind this project is to simplify the means of counting students in each classroom which is taken during lectures and by reducing the human power and time. With the tiny face detection system in place, it will be easy to tell the count of students which are actually present in the classrooms.

Computer vision is one of the most successful research areas in the field of machine learning. Major progress occurred in this area within the past two decades. In 2006, Hinton et al proposed a model to train deep neural networks to the desired depth. Then, with the availability of ImageNet, a breakthrough occurred in 2012, with the development of an 8-layer neural network that outperformed all other algorithms in the identification of images in ImageNet. They applied convolutional layers, rectified linear activation, and dropout in their architecture.

Their results paved the way for developing better algorithms in computer vision. Face

detection can be classified as a subset of computer vision. With the application of similar techniques used in computer vision, face detection presents high accuracies in detecting faces. This is important since it enables the detection and authentication of individuals. In this section, we focus our discussion on the deep learning techniques that evolved the work on face detection research. In 2015, FaceNet, , using the GoogleNet-24 framework achieved an accuracy of 99.63% on the LFW database.

Parkhi et al worked on the VGGFace and achieved an accuracy of 98.95% on the LFW database. In this case, the architecture used was the VGGNet16. Most common techniques have relied on ResNet and its variants after Microsoft Research performed an image classification with this architecture with a good performance in 2015. Analogous to our particular case, Cao et al used ResNet-50 architecture to access face detection performance in their work.

Several works have been conducted to detect faces with occlusion. There are two main approaches used in this regard, which are the restoration approach and discard occlusion based approach. The restoration approach tries to restore the occluded parts of the images based on the images in the training. Bachi et al use a 3D face detection system in restoring parts of the face which are occluded. The system registers a 3D input of a person's face using the Iterative Closest Point Algorithm, then the occluded parts of the face are detected, followed by restoration technique using the Principal Component Algorithm(PCA). Similarly, Drira et al [5] use a 3D based statistical approach in detection and estimating the occluded part of the face. THE PCA approach is used in the recovery of occluded parts of the face.

Proposed System:

Our system will capture the image of the classroom using a webcam and then send that to the server and after computing the image, send the desired result to the respected authority.

Typically, this process can be divided into three stages-

1. Image Capturing:

Images of students sitting in a classroom are captured using a webcam which will be

connected by an IoT device (Raspberry Pi 4 or nVidia Jetson Nano kit). Multiple images will be captured by the devices in desired intervals (suppose 2 images in each period of a class). Then the image will be sent to the server for further computation.

Fig:2.1 Captured image of a classroom

2. Server:

A local server is created using Node.Js . Then when it receives an image, it will run the python script that contains the main code of tiny face detection. Once the python file starts executing, it will detect the number of faces present in the image which was received. For the first time, it will create a JSON file containing the count of the students present in each classroom. The JSON file will be updated when the new image is received.

Fig: 2.2 Server

Fig: 2.3 Face Detected

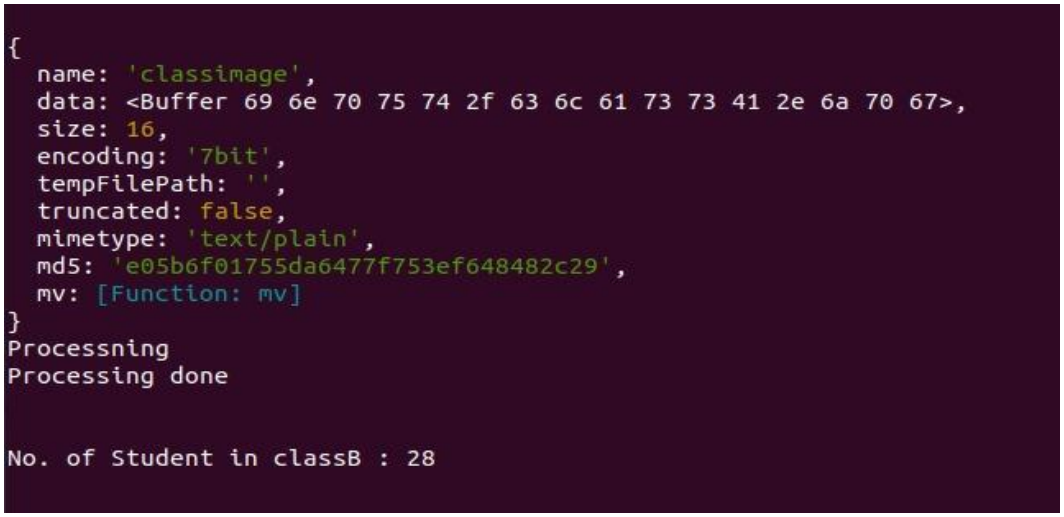


Fig: *2.4 Counting Faces*

3. Display Unit:

Once the data is updated in a JSON file, our display unit (Jetson Nano Nvidia kit) will request the server for the JSON file and after receiving the file. In response to the user's requested query, our system will read the file and perform the desired analysis. There's a choice between showing the count of individual classes or showing the count of students on a particular instance. In order to display the output on the screen, we will use a Graphical User Interface created in Python using pyQt5.

2.2 System Analysis

Systems analysis is the process by which an individual (s) studies a system such that an information system can be analysed, modelled, and a logical alternative can be chosen. Systems analysis projects are initiated for three reasons: problems, opportunities, and directives. The people involved include systems analysts, sponsors, and users. The process by which systems are developed can be described by the systems development life cycle. The tasks, techniques, and tools used by the systems development life cycle can be referred to as a methodology. There are three classifications of methodologies: traditional, information engineering, and object-oriented. CASE tools are automated tools that support specific methodologies.

2.3 Requirement Analysis

Requirements Analysis is the process of defining the expectations of the users for an application that is to be built or modified. It involves all the tasks that are conducted to identify the needs of different stakeholders. Therefore requirements analysis means analyzing, documenting, validating, and managing software or system requirements. High-quality requirements are documented, actionable, measurable, testable, traceable, help to identify business opportunities, and are defined to facilitate system design. Requirement analysis is a significant and essential activity after elicitation. We analyze, refine, and scrutinize the gathered requirements to make consistent and unambiguous requirements. This activity reviews all requirements and may provide a graphical view of the entire system. After the completion of the analysis, it is expected that the understandability of the project may improve significantly. Here, we may also use the interaction with the customer to clarify points of confusion and to understand which requirements are more important than others. This process usually consists of various graphical representations of the functions, data entities, external entities, and the relationships between them. The graphical view may help to find incorrect, inconsistent, missing, and superfluous requirements. Such models include the Data Flow diagram, Entity-Relationship diagram, Data Dictionaries, State-transition diagrams, etc. After modeling the requirements, we will have a better understanding of the system behavior.

The inconsistencies and ambiguities have been identified and corrected. The flow of data amongst various modules has been analyzed.

2.3.1 SOFTWARE REQUIREMENTS

- Python3
- Numpy
- Pandas
- Tensorflow
- MatplotLib

2.3.2 HARDWARE REQUIREMENTS

- 64-bit CPU (Intel /AMD architecture)
- 2 GHz above processor
- 8 GB Ram recommended
- Nvidia Jetson Nano Kit
- Raspberry Pi 4 Kit
- High-Quality Webcam
- Display Monitor

2.4 SDLC MODEL

Software Development life cycle (SDLC) is a spiritual model used in project management that defines the stages include in an information system development project, from an initial feasibility study to the maintenance of the completed application. SDLC is a series of phases that provide a common understanding of the software building process. How the software will be realized and developed from the business understanding and requirements elicitation phase to convert these business ideas and requirements into functions and features until its usage and operation to achieve the business needs. A good software engineer should have enough knowledge on how to choose the SDLC model based on the project context and the business requirements.

Therefore, it may be required to choose the right SDLC model according to the specific concerns and requirements of the project to ensure its success.

There are different software development life cycle models specified and design, which are followed during the software development phase. These models are also called "*Software Development Process Models*" Each process model follows a series of phases unique to its type to ensure success in the step of software development.

2.4.1. Rapid Application Development (RAD)

RAD is a development model that prioritizes rapid prototyping and quick feedback over long drawn-out development and testing cycles. This helps ensure that the final outcome is more quality-focused and is in alignment with the end-users requirements. In our project as per the RAD model, we have to identify and analyze the system functional requirements and on the basis of requirements, the design of the system functionality and the development method is finished with success. In the RAD model, at the very beginning, rapid application development sets itself apart from traditional software development models. It doesn't require you to sit with end users and get a detailed list of specifications; instead, it asks for a broad requirement. The broad nature of the requirements helps you take the time to segment specific requirements at different points of the development cycle.

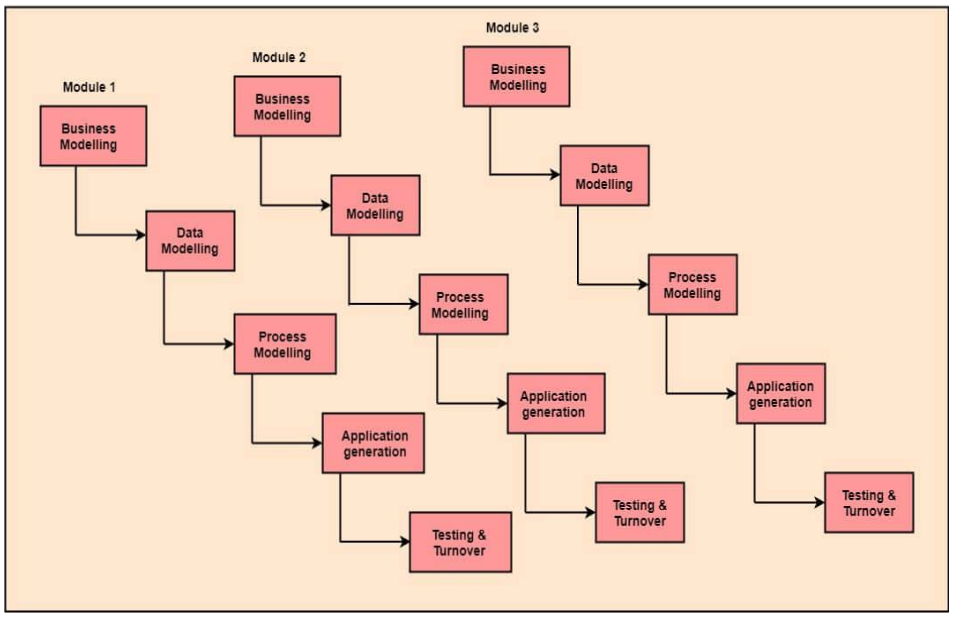


Fig: 2.5 RAD model Diagram

The sequential phases in RAD model are −

- **Business Modelling**: The information flow among business functions is defined by answering questions like what data drives the business process, what data is generated, who generates it, where does the information go, who process it and so on.

- **Data Modelling**: The data collected from business modeling is refined into a set of data objects (entities) that are needed to support the business. The attributes (character of each entity) are identified, and the relation between these data objects (entities) is defined.

- **Process Modelling:** The information object defined in the data modeling phase are transformed to achieve the data flow necessary to implement a business function. Processing descriptions are created for adding, modifying, deleting, or retrieving a data object.

- **Application Generation:** Automated tools are used to facilitate construction of the software; even they use the 4th GL techniques.

- **Testing & Turnover:** Many of the programming components have already been tested since RAD emphasis reuse. This reduces the overall testing time. But the new part must be tested, and all interfaces must be fully exercised.

2.5 Data Flow Diagram (DFD)

DFD is the abbreviation for Data Flow Diagram. The flow of data of a system or a process is represented by DFD. It also gives insight into the inputs and outputs of each entity and the process itself. DFD does not have control flow and no loops or decision rules are present. Specific operations depending on the type of data can be explained by a flowchart. Data Flow Diagram can be represented in several ways. The DFD belongs to structured-analysis modeling tools. Data Flow diagrams are very popular because they help us to visualize the major steps and data involved in software-system processes.

DFD notions and symbols vary according to the methodology model employed. Some organizations have adopted their own conventions, though this is not recommended.

Different DFD notations include:

- Gane and Sarson
- Yourdon and De Marco
- SSADM
- UML (commonly used to map software architecture, but can be used in DFDs)

The Data Flow Diagram has 4 components:

1. **Process -** Input to output transformation in a system takes place because of process function. The symbols of a process are rectangular with rounded corners, oval, rectangle or circle. The process is named a short sentence, in one word or a phrase to express its essence
2. **Data Flow -** Data flow describes the information transferred between different parts of the systems. The arrow symbol is the symbol of data flow. A relatable name should be given to the flow to determine the information which is being moved. Data flow also represents material along with information that is being moved. Material shifts are modeled in systems that are not merely informative. A given flow should only transfer a single type of information. The direction of flow is represented by the arrow which can also be bi-directional.
3. **Warehouse -** The data is stored in the warehouse for later use. Two horizontal lines represent the symbol of the store. The warehouse is simply not restricted to being a data file rather it can be anything like a folder with documents, an optical disc, or a filing cabinet. The data warehouse can be viewed independently of its implementation. When the data flow from the warehouse it is considered data reading and when data flows to the warehouse it is called data entry or data updation.
4. **Terminator -** The Terminator is an external entity that stands outside of the system and communicates with the system. It can be, for example, organizations like banks, groups of people like customers, or different departments of the same organization, which is not a part of the model system and an external entity. Modeled systems also communicate with terminators.

2.5.1 DFD LEVEL-0

It also known as the fundamental system model, or context diagram represents the entire software requirement as a single bubble with input and output data denoted by incoming and outgoing arrows. Then the system is decomposed and described as a DFD with multiple bubbles. Parts of the system represented by each of these bubbles are then decomposed and documented as more and more detailed DFDs. This process may be repeated at as many levels as necessary until the program at hand is well understood. It is essential to preserve the number of inputs and outputs between levels, this concept is called leveling by DeMarco.

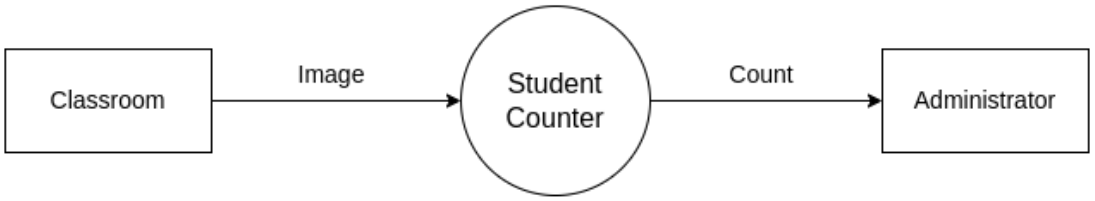


Fig: 2.6 DFD LEVEL-0

2.5.2 DFD LEVEL-1

In 1-level DFD, a context diagram is decomposed into multiple bubbles/processes. In this level, we highlight the main objectives of the system and breakdown the high-level process of 0-level DFD into subprocesses.

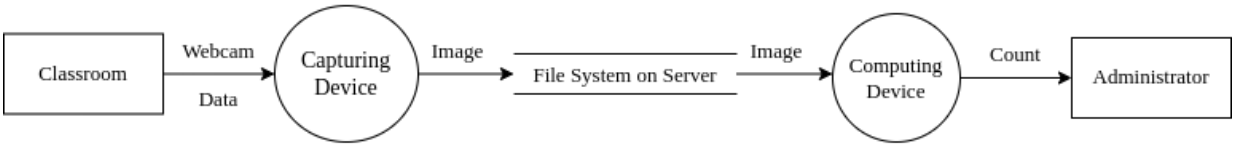


Fig: 2.7 DFD LEVEL-1

CHAPTER - III
SYSTEM ANALYSIS

3.1 Work-Flow diagram

A workflow diagram (also known as a workflow) provides a graphic overview of the business process. Using standardized symbols and shapes, the workflow shows step-by-step how your work is completed from start to finish. It also shows who is responsible for work at what point in the process. Designing a workflow involves first conducting a thorough workflow analysis, which can expose potential weaknesses. Workflow analysis can help you define, standardize and identify critical areas of your process.

Workflows are also useful to help employees understand their roles and the order in which work is completed and to create more unity within different departments. Originating from the manufacturing industry, workflows are now used by a variety of industries—from the government to finance to commerce—and are easier than ever to create.

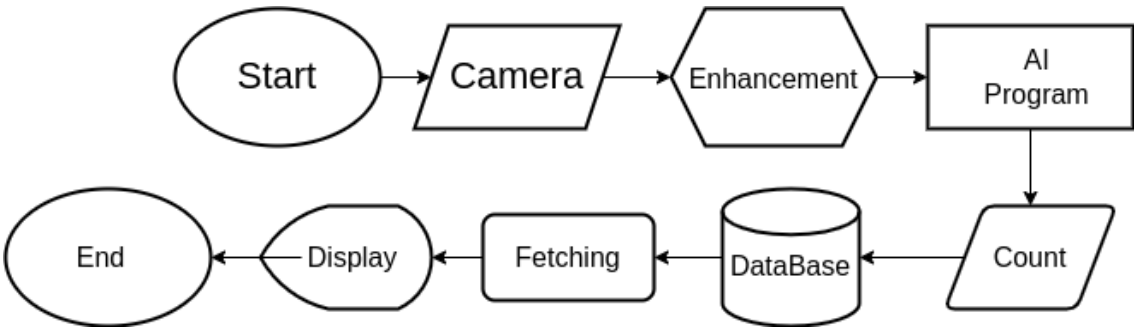


Fig 3.1 WorkFlow Diagram

3.2 Use Case Diagram

A use case diagram is used to represent the dynamic behavior of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and also tells how the user handles a system.

The main purpose of a use case diagram is to portray the dynamic aspect of a system. It accumulates the system's requirement, which includes both internal as well as external influences. It invokes persons, use cases, and several things that invoke the actors and elements accountable for the implementation of use case diagrams. It represents how an entity from the external environment can interact with a part of the system.

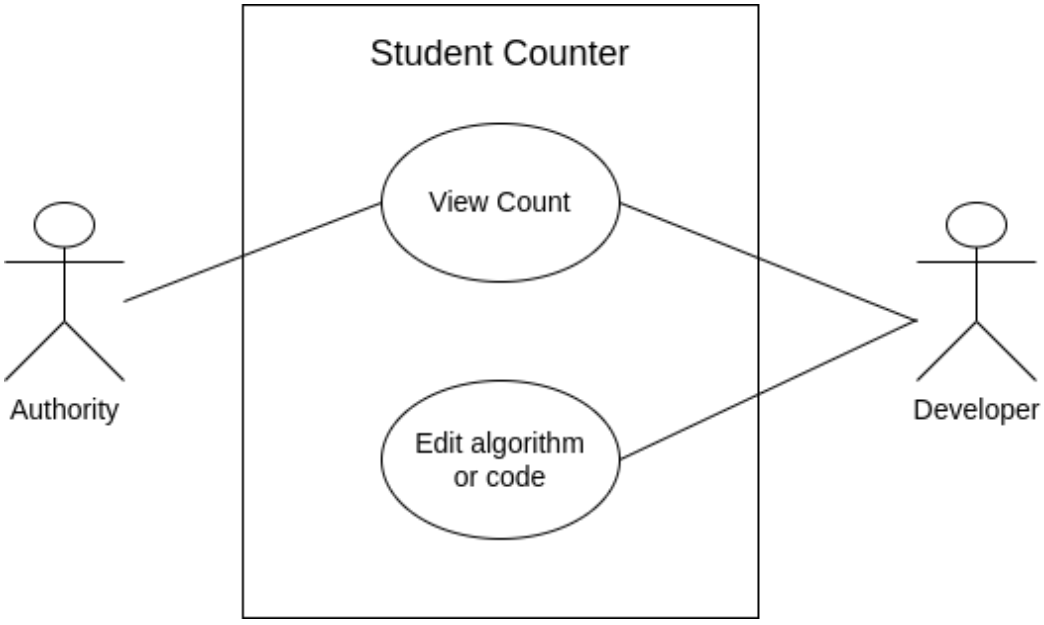


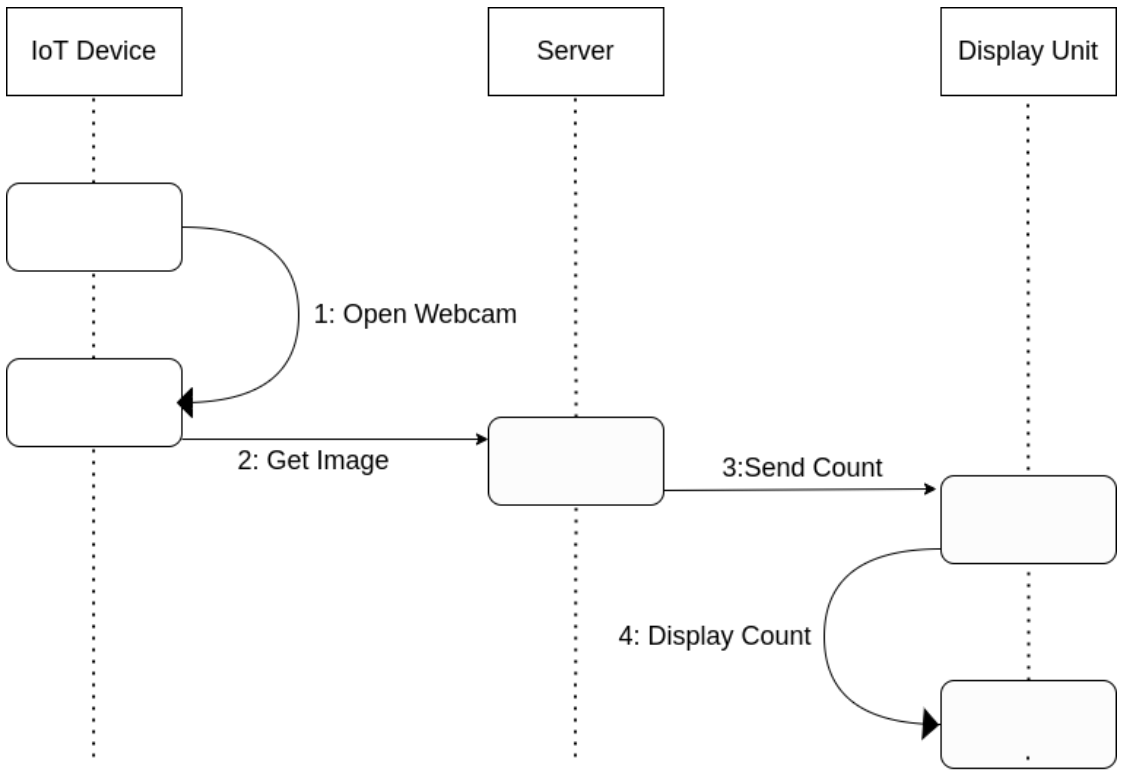
Fig.3.2 USE CASE MODEL

3.3.1 Sequence Diagram

Sequence diagrams are a popular dynamic modeling solution in UML because they specifically focus on lifelines, or the processes and objects that live simultaneously, and the messages exchanged between them to perform a function before the lifeline ends.

Purpose of a Sequence Diagram -

- To model high-level interaction among active objects within a system.
- To model interaction among objects inside a collaboration realizing a use case.
- It either models generic interactions or some certain instances of interaction.



Fig, 3.3.1 Sequence Diagram

3.3.2 Activity diagram

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.
The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc

Purpose of Activity Diagrams

- The basic purposes of activity diagrams is similar to other four diagrams. It captures the dynamic behavior of the system. Other four diagrams are used to show the message flow from one object to another but activity diagram is used to show message flow from one activity to another.

- Activity is a particular operation of the system. Activity diagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques. The only missing thing in the activity diagram is the message part.

- It does not show any message flow from one activity to another. Activity diagram is sometimes considered as the flowchart. Although the diagrams look like a flowchart, they are not. It shows different flows such as parallel, branched, concurrent, and single.

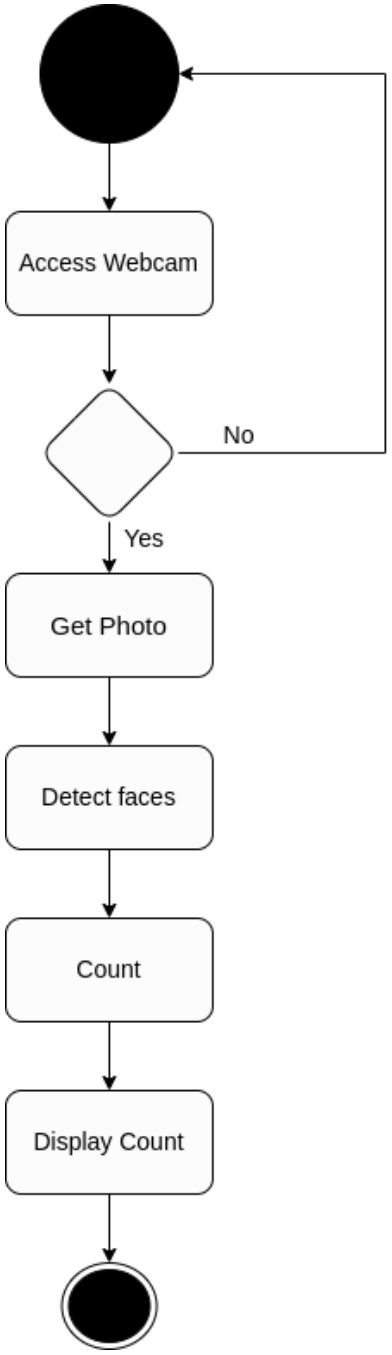


Fig 3.3.2 Activity diagram

The purpose of an activity diagram can be described as −

- Draw the activity flow of a system.
- Describe the sequence from one activity to another.
- Describe the parallel, branched and concurrent flow of the system.

3.4.1 Collaboration Diagram

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). These diagrams can be used to portray the dynamic behavior of a particular use case and define the role of each object.

Collaboration diagrams are created by first identifying the structural elements required to carry out the functionality of an interaction. A model is then built using the relationships between those elements. Several vendors offer software for creating and editing collaboration diagrams.

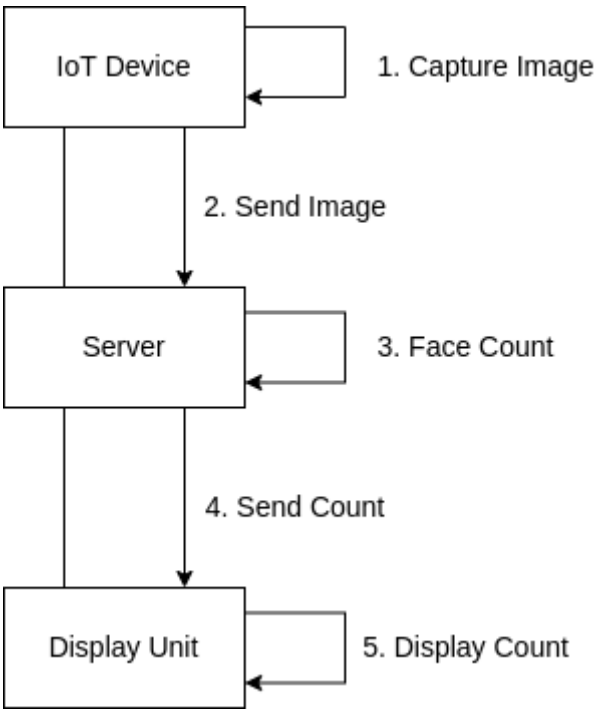


Fig. 3.4.1 Collaboration Diagram

3.4.2 Class Diagram

Class diagram is a static diagram. It represents the static view of an application. The class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. Class diagrams are widely used in the modeling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages. The class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.

The purpose of a class diagram is to model the static view of an application. Class diagrams are the only diagrams that can be directly mapped with object-oriented languages and are thus widely used at the time of construction. UML diagrams like activity diagrams and sequence diagrams can only give the sequence flow of the application, however, the class diagram is a bit different. It is the most popular UML diagram in the coder community.

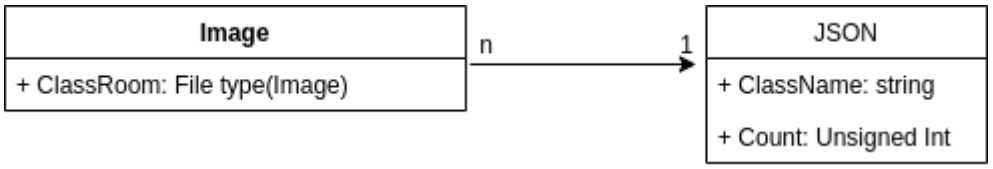


Fig 3.4.2 Class Diagram

CHAPTER - IV
SNAPSHOT

4. Snapshots

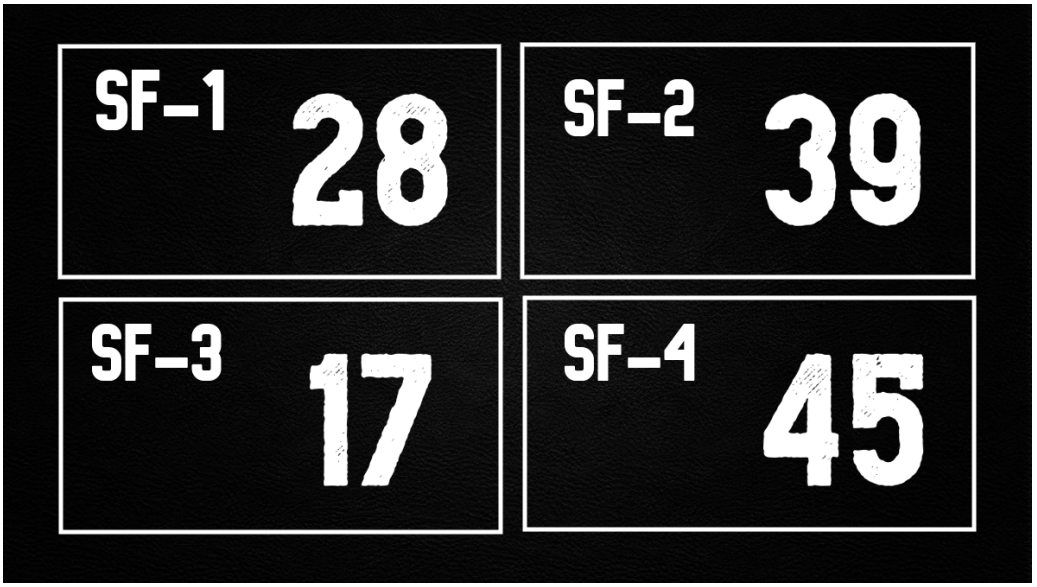


Fig. 4.1 - Display Count GUI

Fig. 4.2 - Student Count Analysis

CHAPTER - V

CONCLUSION

5. Conclusion

This system aims to ease the task of counting the students sitting in each classroom. The proposed system will be able to detect the number of students sitting in a classroom. For Tiny Face Detection, a pre-trained Residual Network is used. It will capture faces via webcam. After capturing, it will send the image for computation (i.e. detecting the number of students sitting in a classroom or lab) and then send the count to authorities and display them via GUI.

The Student Counter System helps in increasing the accuracy and speed ultimately achieving the real-time calculations of students present in a classroom. It not only reduces the chances of errors that can be made by humans but also reduces human interaction and at the same time increases its efficiency. It reduces the time taken to complete the task too.

Face detection is an emerging technology that can provide many benefits. Face detection can save resources and time, and even generate new income streams, for companies that implement it right.

CHAPTER - VI
FUTURE SCOPE

6. Future Scope

The future scope of the project can be integrated with the system with an Android application or website so that authorities can view it from any place and at any time.

The Application will be capable of viewing the counts.
Additionally we can provide a database so that the history of records of students in that classroom can be achieved.

REFERENCES

1. Raspberry Pi 4 For Beginners And Intermediates: A Comprehensive Guide for Beginner and Intermediates to Master the New Raspberry Pi 4 and Set up Innovative Projects by Craig Berg https://singleboardbytes.com/647/install-opencv-raspberry-pi-4.htm

2. NVIDIA Jetson nano official website: https://forums.developer.nvidia.com/t/official-tensorflow-for-jetson-nano

3. Raspberry pi official website: https://www.raspberrypi.org/

4. Dependencies on Jetson Nano https://qengineering.eu/install-tensorflow-2.4.0-on-jetson-nano.html

5. For Research paper reference: https://scholar.google.com/

PAPER PUBLISHED


ICSRESM 2022
3rd International Conference on
Sustainable Research in Engineering Science and Management
Paper ID - 53
CERTIFICATE
This is to certify that
Tushar Dubey
Shri Shankaracharya Institute of Professional Management and Technology Raipur Chhattisgarh, India
has contributed a paper titled
Student Counter Using Iot and Face Detection Algorithm
in 3rd International Conference on Sustainable Research in Engineering Science and Management (ICSRESM-2022) held during December 16, 2022 on Shri Shankaracharya Institute of Professional Management and Technology, Raipur, Chhattisgarh, India We wish the authors all the very best for future endeavors.
Dr. Suman Kumar Swarnkar, Convener (ICSRESM-2022)
Dr. J P Patra, Coordinator (ICSRESM-2022)
Dr. Alok Kumar Jain, Principal, SSIPMT
www.inence.co.in


ICSRESM 2022
3rd International Conference on
Sustainable Research in Engineering Science and Management
Paper ID - 53
CERTIFICATE
This is to certify that
Kanha Agrawal
Shri Shankaracharya Institute of Professional Management and Technology Raipur Chhattisgarh, India
has contributed a paper titled
Student Counter Using Iot and Face Detection Algorithm
in 3rd International Conference on Sustainable Research in Engineering Science and Management (ICSRESM-2022) held during December 16, 2022 on Shri Shankaracharya Institute of Professional Management and Technology, Raipur, Chhattisgarh, India We wish the authors all the very best for future endeavors.
Dr. Suman Kumar Swarnkar, Convener (ICSRESM-2022)
Dr. J P Patra, Coordinator (ICSRESM-2022)
Dr. Alok Kumar Jain, Principal, SSIPMT
www.inence.co.in


ICSRESM 2022
3rd International Conference on
Sustainable Research in Engineering Science and Management
Paper ID - 76
CERTIFICATE
This is to certify that
Vishal Rathi
Shri Shankaracharya Institute of Professional Management and Technology Raipur Chhattisgarh, India
has contributed a paper titled
Land Dispute Resolution Using BlockChain
in 3rd International Conference on Sustainable Research in Engineering Science and Management (ICSRESM-2022) held during December 16, 2022 on Shri Shankaracharya Institute of Professional Management and Technology, Raipur, Chhattisgarh, India We wish the authors all the very best for future endeavors.
Dr. Suman Kumar Swarnkar, Convener (ICSRESM-2022)
Dr. J P Patra, Coordinator (ICSRESM-2022)
Dr. Alok Kumar Jain, Principal, SSIPMT
www.inence.co.in