A

Mini Project

On

#### CROWD COUNTING METHOD BASED ON CNN

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

E. SAKETHA (207R1A0576)

G. BHAVITHA (207R1A0578)

B. BHARATH KUMAR (207R1A0566)

Under the Guidance of

**RAHEEM UNNISA**

(Assistant Professor)

##### DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**CMR TECHNICAL CAMPUS UGC AUTONOMOUS**

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi) Recognized Under Section 2(f) & 12(B) of the UGCAct.1956, Kandlakoya (V), Medchal Road, Hyderabad-501401.

**2020-2024**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



#### CERTIFICATE

This is to certify that the project entitled **“CROWD COUNTING METHOD BASED ON CNN”** being submitted by **E.SAKETHA (207R1A0576),G.BHAVITHA (207R1A0578) & B. BHARATH KUMAR (207R1A0566)** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafied work carried out by them under our guidance and supervision during year 2023-24.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

**Raheem Unnisa Dr. A. Raji Reddy**

(Assistant Professor) DIRECTOR

INTERNAL GUIDE

**Dr. K. Srujan Raju EXTERNAL EXAMINER**

HOD

**Submitted for viva voice Examination held on**

##### ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to my guide **Raheem Unnisa,** Associate Professor for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to the Project Review Committee (PRC) **Dr. J. Narasimharao,** **G.Vinesh Shanker, Ms. Shilpa, & Dr. K. Maheswari** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. K. Srujan Raju,** Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr. A. Raji Reddy,** Director for being cooperative throughout the course of this project. We also express our sincere gratitude to Sri. **Ch. Gopal Reddy,** Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

###### E. SAKETHA (207R1A0576)

**G. BHAVITHA (207R1A0578)**

**B. BHARATH KUMAR (207R1A0566)**

##### ABSTRACT

Crowd counting is an important research topic in the field of computer vision. The multi-column convolution neural network (MCNN) has been used in this field and achieved competitive performance. However, when the crowd distribution is uneven, the accuracy of crowd counting based on the MCNN still needs to be improved. In order to adapt to uneven crowd distributions, crowd global density feature is taken into account in this paper. The global density features are extracted and added to the MCNN through the cascaded learning method. Because some detailed features during the down-sampling process will be lost in the MCNN and it will affect the accuracy of the density map, an improved MCNN structure is proposed. In this paper, the max pooling is replaced by max-ave pooling to keep more detailed features and the deconvolutional layers are added to restore the lost details in the down-sampling process. The experimental results in the UCF\_CC\_50 dataset and the ShanghaiTech dataset show that the proposed method has higher accuracy and stability.

|  |  |  |
| --- | --- | --- |
| **FIGURE NO** | **FIGURE NAME** | **PAGE NO** |
| Figure 3.1 | Project Architecture for Crowd  Counting Method Based on CNN | 8 |
| Figure 3.2 | Use Case Diagram for Crowd  Counting Method Based on  CNN | 11 |
| Figure 3.3 | Class Diagram for Crowd  Counting Method Based on  CNN | 12 |
| Figure 3.4 | Sequence diagram for Crowd  Counting Method Based on  CNN | 13 |
| Figure 3.5 | Activity diagram for Crowd  Counting Method Based on  CNN | 14 |

|  |  |  |
| --- | --- | --- |
| **SCREENSHOT NO.** | **SCREENSHOT NAME** | **PAGE NO**. |
| Screenshot 5.1 | People counting from Image | 18 |
| Screenshot 5.2 | Selecting and Uploading ‘1.jpg’file | 18 |
| Screenshot 5.3 | Output as “Total Head : 1” | 19 |
| Screenshot 5.4 | Output as “Total Head : 5” | 19 |
| Screenshot 5.5 | Selecting and Uploading ‘vtest.avi’ file to load video and start human head counting | 20 |
| Screenshot 5.6 | In above frame we got 3 head as total humans are 3 and in below screen we got as 4 | 20 |
| Screenshot 5.7 | Press ‘q’ key on video to stop processing | 21 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

###### ABSTRACT i

LIST OF FIGURES ii

LIST OF SCREENSHOTS iii

1.[INTRODUCTION 1](#_TOC_250030)

1.1 [PROJECT SCOPE 1](#_TOC_250029)

1.2 [PROJECT PURPOSE 1](#_TOC_250028)

1.3 [PROJECT FEATURES 2](#_TOC_250027)

2.[SYSTEM ANALYSIS 3](#_TOC_250026)

2.1 [PROBLEM DEFINITION 3](#_TOC_250025)

2.2 [EXISTING SYSTEM 4](#_TOC_250024)

2.2.1 DISADVANTAGES OF EXISTING SYSTEM 4

2.3 [PROPOSED SYSTEM 4](#_TOC_250023)

2.3.1 ADVANTAGES OF PROPOSED SYSTEM 4

2.4 [FEASIBILITY STUDY 5](#_TOC_250022)

2.4.1 [ECONOMIC FEASIBILITY 5](#_TOC_250021)

2.4.2 [TECHNICAL FEASIBILITY 5](#_TOC_250020)

2.4.3 SOCIAL FEASIBILITY 6

2.5 [HARDWARE & SOFTWARE REQUIREMENTS 6](#_TOC_250019)

2.5.1 [HARDWARE REQUIREMENTS 6](#_TOC_250018)

2.5.2 [SOFTWARE REQUIREMENTS 7](#_TOC_250017)

3.[ARCHITECTURE 8](#_TOC_250016)

3.1 [PROJECT ARCHITECTURE 8](#_TOC_250015)

3.2 [DESCRIPTION 8](#_TOC_250014)

3.3 [USE CASE DIAGRAM 11](#_TOC_250013)

3.4 [CLASS DIAGRAM 12](#_TOC_250012)

3.5 [SEQUENCE DIAGRAM](#_TOC_250011) 13

3.6 [ACTIVITY DIAGRAM 14](#_TOC_250010)

4.IMPLEMENTATION 15

4.1 SAMPLE CODE 15

5.SCREENSHOTS 18

6.[TESTING 22](#_TOC_250009)

6.1 [INTRODUCTION TO TESTING 22](#_TOC_250008)

6.2 [TYPES OF TESTING 22](#_TOC_250007)

6.2.1 UNIT TESTING 22

6.2.2 [INTEGRATION TESTING 23](#_TOC_250006)

6.2.3 [FUNCTIONAL TESTING 23](#_TOC_250005)

6.3 [TEST CASES 24](#_TOC_250004)

6.3.1 [CLASSIFICATION 24](#_TOC_250003)

7.CONCLUSION & FUTURE SCOPE 25

7.1 PROJECT CONCLUSION 25

7.2 [FUTURE SCOPE 25](#_TOC_250002)

8.REFERENCES 27

8.1 [REFERENCES 27](#_TOC_250001)

8.2 [GITHUB LINK 27](#_TOC_250000)

# INTRODUCTION

#### INTRODUCTION

##### 1.1 PROJECT SCOPE

The project scope for crowd counting based on CNNs encompasses the development of a robust system for estimating the number of people or objects within images or video frames. It involves defining the problem, collecting diverse datasets, selecting appropriate CNN architectures, and optimizing models for accuracy. The scope extends to deployment considerations, including real-time processing, scalability, and ethical considerations like privacy and bias mitigation. Depending on the application, the system may need to handle various crowd densities and environmental conditions. The project may also involve user interface development for practical use cases and a feedback loop for continuous improvement. Clear documentation is essential to maintain transparency and facilitate future enhancements, ensuring the successful implementation of crowd counting using CNNs.

##### 1.2 PROJECT PURPOSE

The purpose of a crowd counting project based on CNNs is to develop an intelligent system capable of accurately estimating the number of people or objects within images or video frames. This technology serves a variety of critical applications, including crowd management, urban planning, event security, and retail analytics. By harnessing deep learning through CNNs, the project aims to provide real-time or offline crowd counting solutions that enhance decision-making processes and improve operational efficiency across diverse domains. Ultimately, the purpose is to create a valuable tool for better understanding and managing crowds in various real-world scenarios.

CMRTC 1

##### 

##### 1.3 PROJECT FEATURES

In a crowd counting project based on CNNs, several crucial features define its functionality and effectiveness. At its core, this project harnesses the power of Convolutional Neural Networks, leveraging their ability to extract meaningful features from crowd images or video frames. These features encompass the diversity of datasets, ensuring the system's adaptability to various crowd densities, lighting conditions, and camera angles, thereby enhancing its robustness. Accurate crowd counting is achieved through the application of evaluation metrics like Mean Absolute Error (MAE) and Mean Squared Error (MSE). Depending on the application, the system may be tailored for real-time processing, enabling timely decision-making. Scalability is a fundamental feature, allowing the system to handle crowds of different sizes and densities across various scenarios. Ethical considerations, such as privacy protection and bias mitigation, are integral components of the project, ensuring responsible deployment. Flexible deployment options, including surveillance cameras, event venues, or retail stores, cater to diverse use cases. User-friendly interfaces simplify interaction and interpretation of crowd count results. Continuous monitoring, testing, and feedback loops guarantee ongoing system performance and adaptation. Comprehensive documentation underpins transparency, facilitating future enhancements and the responsible use of

crowd counting technology.

CMRTC 2

## SYSTEM ANALYSIS

##### SYSTEM ANALYSIS

System analysis for crowd counting using CNNs involves a comprehensive approach to develop an effective counting solution. This process encompasses various stages, starting with defining the problem's scope and accuracy requirements. Model selection involves choosing a suitable CNN architecture, while training and fine-tuning optimize model performance. Deployment considerations include real-time processing and scalability for different crowd scenarios.

##### 2.1 PROBLEM DEFINITION

Crowd counting using Convolutional Neural Networks (CNNs) addresses the task of determining the count of individuals or objects within images or video frames. The core objective is to accurately quantify the number of people or objects in crowded scenes, making it applicable to various domains like crowd management, traffic analysis, and event planning. The problem encompasses diverse challenges, including variations in crowd density, lighting conditions, and camera perspectives. To tackle this, CNNs are leveraged for their ability to automatically learn and extract relevant features from the data.

##### 2.2 EXISTING SYSTEM

Crowd counting is used to calculate the total number of people in images or video frames. The crowd counting methods can be divided into three categories: the direct counting method based on target detection, the indirect method based on feature regression and crowd counting based on deep learning. In the relevant researches based on target detection, transform to extract the feature area of the head-like contour and build the SVM classifier to classify the feature area. All of these methods are suitable for the scenes with low density crowd, but the detection accuracy will decrease in the case of high density crowd.

CMRTC 3

###### 2.2.1 DISADVANTAGES OF EXISTING SYSTEM

Following are the disadvantages of existing system:

* + - * Limited accuracy.
      * Data dependency.
      * Difficulty in handling dynamic scenes.
      * Limited generalization.
      * Limited interpretability.

##### 2.3 PROPOSED SYSTEM

With the rapid development of deep learning and big data, crowd counting methods based on deep learning are proposed gradually. It is proposed a cross-scene crowd counting model. It was trained alternately through two learning objectives, density map and global number. However, it is not suitable for the change in the scale of crowd. It is proposed to use the MCNN with three branch networks for crowd counting. Different receptive fields were used in each branch network, and this improved MCNN could adapt to the change in the scale of the crowd. A convolutional neural network with global density feature by using multi-task network cascades (MNCs) is proposed. In order to generate a more comprehensive density map, the max-ave pooling layers are used to keep more features of the image. Meantime, the deconvolutional layers are added to the convolutional neural network in order to restore the lost details in down-sampling process. It will help to improve the accuracy of density map and further improve the accuracy of crowd counting.

###### 2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

* + - * Improved accuracy
      * Reduced data dependency.
      * Ability to handle dynamic scenes.
      * Improved generalization.
      * Improved interpretability.

CMRTC 4

##### 2.4 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and a business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. Three key considerations involved in the feasibility analysis:

* Economic Feasibility
* Technical Feasibility
* Social Feasibility

###### 2.4.1 ECONOMIC FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

###### 2.4.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

CMRTC 5

###### 

**2.4.3 SOCIAL FEASIBILITY**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

##### 2.5 HARDWARE & SOFTWARE REQUIREMENTS

###### 2.5.1 HARDWARE REQUIREMENTS:

Hardware interfaces specify the logical characteristics of each interface between the software product and the hardware components of the system. The following are some hardware requirements.

* Processor : Any Update Processer
* Ram : Minimum 4 GB
  + - * + Hard Disk : Minimum 100 GB

CMRTC 6

##### 

##### 2.5.2 SOFTWARE REQUIREMENTS:

Software Requirements specifies the logical characteristics of each interface and software components of the system. The following are some software requirements.

* + - * + Operating System : Windows
        + Technology : Python 3.7
        + IDE : PyCharm

CMRTC 7

## ARCHITECTURE

##### 3. ARCHITECTURE

##### 3.1 PROJECT ARCHITECTURE

This project architecture shows the procedure followed for classification, starting from input to final prediction.

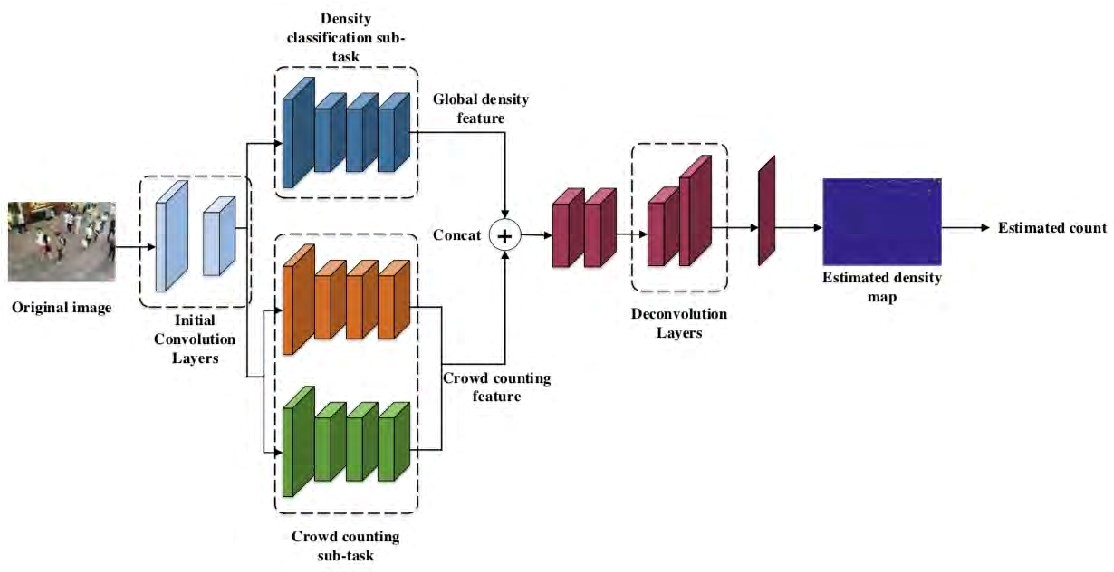


Figure 3.1: Project Architecture of Crowd Counting Method Based on CNN

###### 3.2 DESCRIPTION

Crowd counting methods based on Convolutional Neural Networks (CNN) architecture are used to estimate the number of individuals in a crowded scene from images or videos. Below, I'll describe each layer or sub-task involved in a typical CNN-based crowd counting method:

CMRTC 8

**Original Image:** The input to the network is the original image of the crowded scene. This image contains people or objects whose count needs to be estimated.

**Initial Convolutional Layer:** The original image is passed through a series of convolutional layers. These layers are responsible for extracting low-level features from the image, such as edges, corners, and textures. The depth of these layers increases gradually to capture more complex patterns.

**Density Classification on Sub-task:** In some crowd counting methods, a sub-task involves classifying different regions of the image into high-density and low-density regions. This classification helps in focusing on areas with varying crowd densities.

**Crowd Counting Sub-task:** The core of the network is designed to estimate crowd counts. This part of the network processes the intermediate features from previous layers and produces a count estimation for the entire image or specific regions of interest.

**Global Density Feature:** This feature represents the overall crowd density in the entire image. It is computed from the output of the crowd counting sub-task and provides a global perspective on crowd density.

**Crowd Counting Feature:** This feature encodes localized information about crowd density. It captures variations in crowd density at different spatial locations in the image. These features are often obtained from the intermediate layers of the network.

**Concatenation (Concat):** The global density feature and the crowd counting feature are concatenated or combined in some way to provide a holistic representation of crowd density. This step helps the network fuse information from different scales and perspectives.

CMRTC 9

**Deconvolutional Layers:** After concatenation, deconvolutional layers or transposed convolution layers are used to upsample the feature maps. These layers help recover spatial information lost during downsampling and produce a density map that has the

same resolution as the original image.

**Estimated Density Map**: The output of the deconvolutional layers is an estimated density map. This map assigns a density value to each pixel in the input image, indicating the local crowd density at that location.

**Estimated Count:** The final step is to compute the estimated count from the density map. This can be done by summing the density values in the density map or by employing other post-processing techniques to refine the count estimate.

CMRTC 10

###### 3.3 USE CASE DIAGRAM

In the use case diagram, we have basically one actor who is the user in the trained model.

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures.

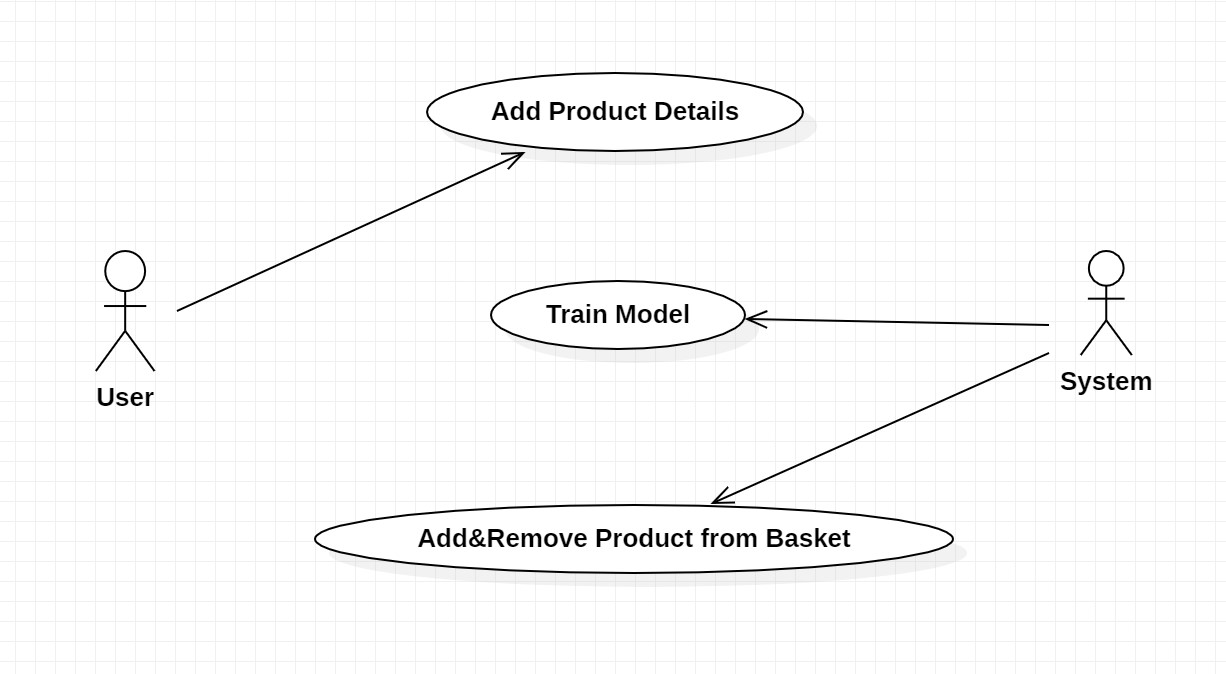


Figure 3.2: Use Case Diagram for Crowd Counting Method Based

on CNN

CMRTC 11

##### 3.4 CLASS DIAGRAM

Class diagram is a type of static structure diagram that describes the structure of a system by showing the system’s classes, their attributes, operations(or methods), and the relationships among objects.

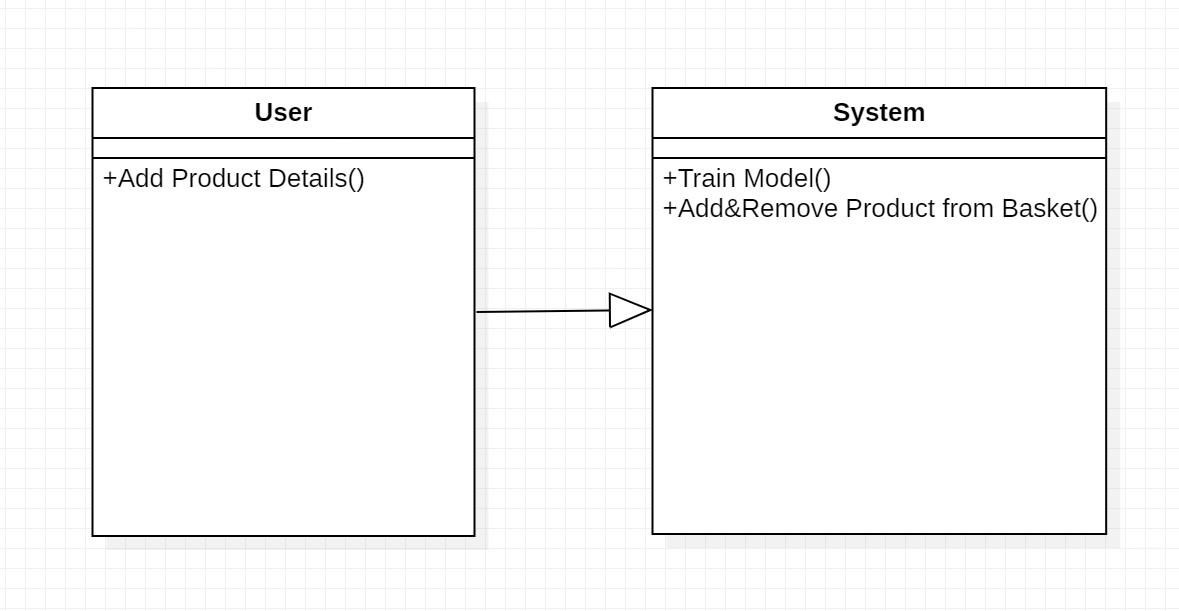


Figure 3.3: Class Diagram for Crowd Counting Method Based

on CNN

CMRTC 12

##### 3.5 SEQUENCE DIAGRAM

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development.

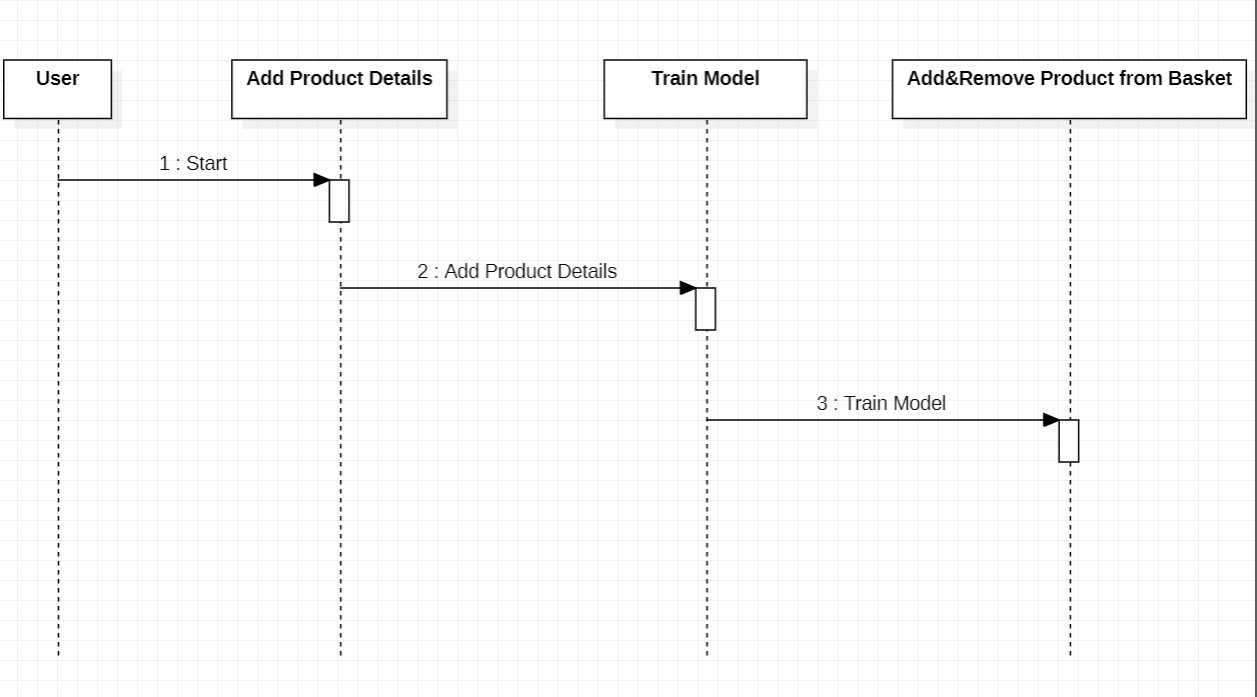


Figure 3.4: Sequence Diagram for Crowd Counting Method Based

on CNN

CMRTC 13

###### 3.6 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. They can also include elements showing the flow of data between activities through one or more data stores.

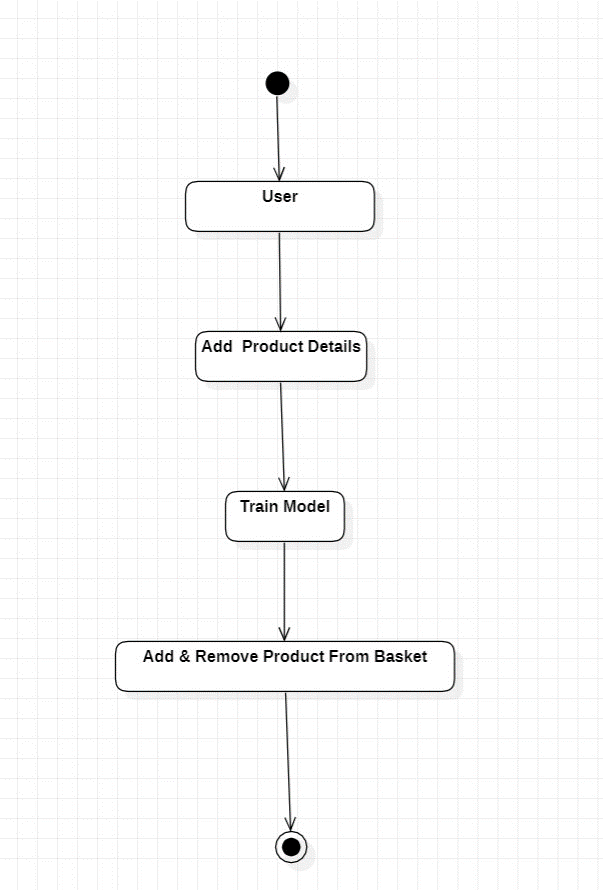


Figure 3.5: Activity Diagram for Crowd Counting Method Based

on CNN

CMRTC 14

## 4. IMPLEMENTATION

##### 4.1 SAMPLE CODE

from tkinter import \*

import tkinter

from tkinter import filedialog

from tkinter.filedialog import askopenfilename

from PIL import Image

import torch

import torchvision.transforms as T

import torchvision

import torch

import numpy as np

import cv2

import os

main = tkinter.Tk()

main.title("Crowd Counting Method Based on Convolutional Neural Network With Global Density Feature")

main.geometry("1200x1200")

#loading FASTER RCNN model to count human head from images and videos

model = torchvision.models.detection.fasterrcnn\_resnet50\_fpn(pretrained=True)

model.eval()

def get\_prediction(img\_path, threshold):

img = Image.open(img\_path)

transform = T.Compose([T.ToTensor()])

img = transform(img)

pred = model([img])

pred\_class = []

for i in list(pred[0]['labels'].numpy()):

pred\_class.append(i)

pred\_boxes = [[(i[0], i[1]), (i[2], i[3])] for i in list(pred[0]['boxes'].detach().numpy())]

pred\_score = list(pred[0]['scores'].detach().numpy())

pred\_t = [pred\_score.index(x) for x in pred\_score if x>threshold][-1]

CMRTC 15

pred\_boxes = pred\_boxes[:pred\_t+1]

pred\_class = pred\_class[:pred\_t+1]

head\_count = 0

for i in range(len(pred\_class)):

if pred\_class[i] == 1:

head\_count += 1

return head\_count

def countFromImages():

global filename

count = 0

filename = filedialog.askopenfilename(initialdir="testImages")

text.insert(END,str(filename)+" loaded\n")

pathlabel.config(text=str(filename)+" loaded")

head\_count = get\_prediction(filename, 0.8)

img = cv2.imread(filename)

img = cv2.cvtColor(img, cv2.COLOR\_BGR2RGB)

cv2.putText(img,"Total Head: "+str(head\_count), (10,50), cv2.FONT\_HERSHEY\_SIMPLEX, 0.7, (0,255,0),thickness=2)

cv2.imshow("output",img)

cv2.waitKey(0)

def countFromVideo():

global filename

global frcnn

filename = filedialog.askopenfilename(initialdir="testVideos")

text.insert(END,str(filename)+" loaded\n")

pathlabel.config(text=str(filename)+" loaded")

video = cv2.VideoCapture(filename)

while(True):

ret, frame = video.read()

print(ret)

if ret == True:

cv2.imwrite("test.jpg",frame)

head\_count = get\_prediction("test.jpg", 0.8)

cv2.putText(frame,"Total Head: "+str(head\_count), (10,50),

CMRTC 16

cv2.FONT\_HERSHEY\_SIMPLEX, 0.7, (0,255,0),thickness=2)

cv2.imshow("output",frame)

if cv2.waitKey(50) & 0xFF == ord('q'):

break

else:

break

video.release()

cv2.destroyAllWindows()

font = ('times', 14, 'bold')

title = Label(main, text='Crowd Counting Method Based on Convolutional Neural Network With Global Density Feature')

title.config(bg='DarkGoldenrod1', fg='black')

title.config(font=font)

title.config(height=3, width=120)

title.place(x=5,y=5)

font1 = ('times', 13, 'bold')

imageButton = Button(main, text="People Counting from Images", command=countFromImages)

imageButton.place(x=50,y=100)

imageButton.config(font=font1)

pathlabel = Label(main)

pathlabel.config(bg='brown', fg='white')

pathlabel.config(font=font1)

pathlabel.place(x=480,y=100)

videoButton = Button(main, text="People Counting from Video", command=countFromVideo)

videoButton.place(x=50,y=150)

videoButton.config(font=font1)

font1 = ('times', 12, 'bold')

text=Text(main,height=10,width=150)

scroll=Scrollbar(text)

text.configure(yscrollcommand=scroll.set)

text.place(x=10,y=400)

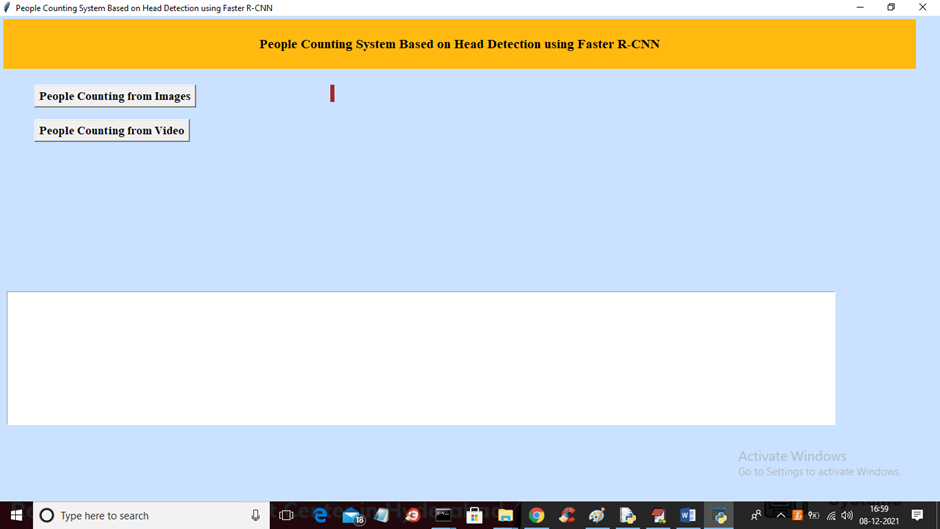
text.config(font=font1)

main.config(bg='LightSteelBlue1')

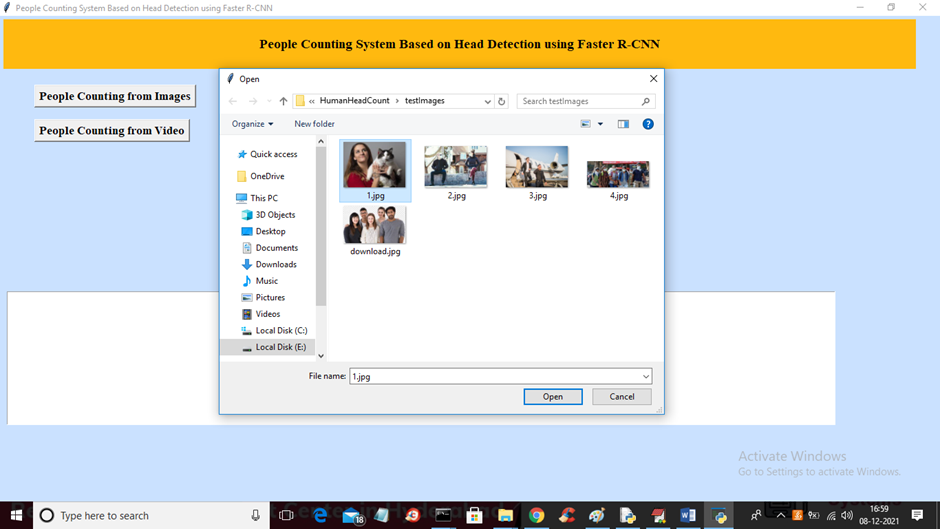
main.mainloop()

CMRTC 17

## 5. SCREENSHOTS

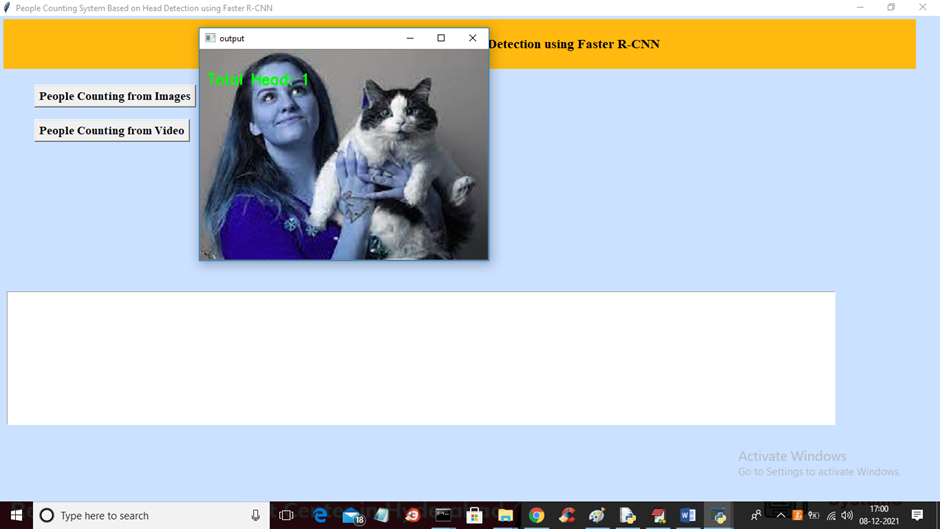
****

Screenshot 5.1: People Counting from Image

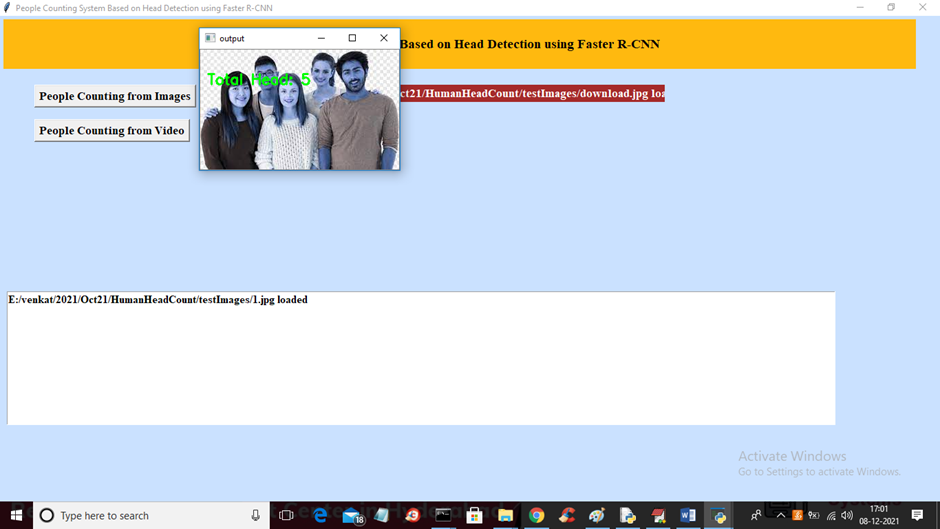


Screenshot 5.2: Selecting and Uploading ‘1.jpg’file

CMRTC 18

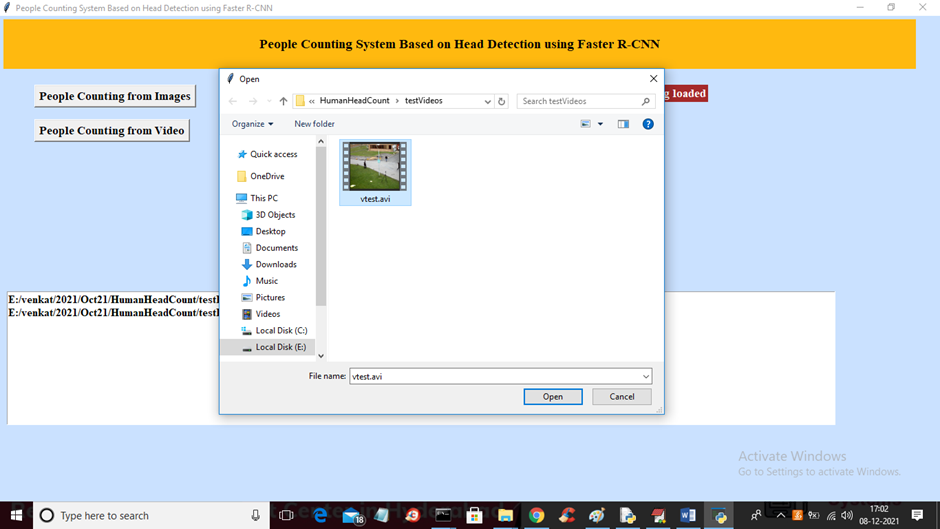


Screenshot 5.3: Output as “Total Head : 1”

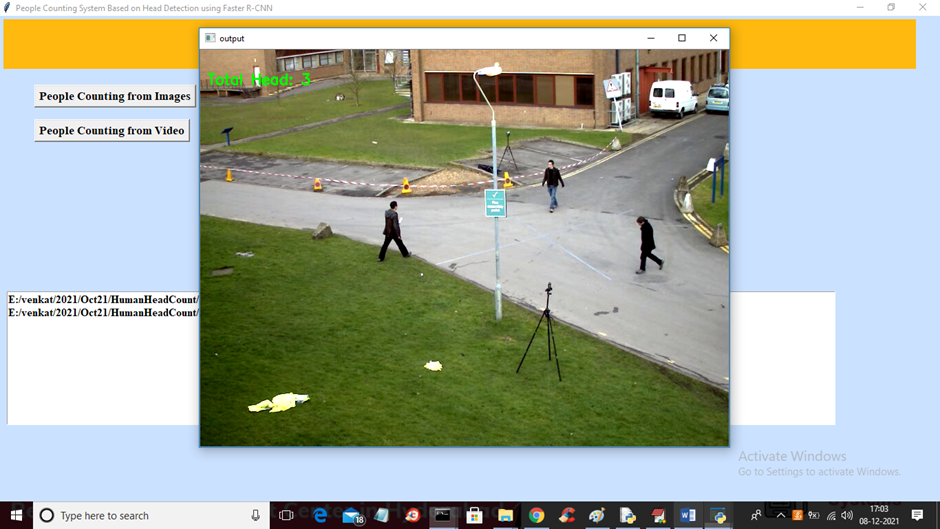


Screenshot 5.4: Output as “Total Head : 5”

CMRTC 19



Screenshot 5.5: Selecting and Uploading ‘vtest.avi’ file to load video and start human head counting



Screenshot 5.6: In above frame we got 3 head as total humans are 3 and in below screen we got as 4

CMRTC 20



Screenshot 5.7: Press ‘q’ key on video to stop processing

CMRTC 21

## 6. TESTING

#### 6. TESTING

##### 6.1 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

##### 6.2 TYPES OF TESTING

###### 6.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .It is done after the completion of an individual unit before integration. This is a structural testing that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

###### CMRTC 22

###### 6.2.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

###### 6.2.3 FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions testedare available as specified by the business and technical requirements, system documentation, and user manuals.Functional testing is centered on the following items:

**Valid Input**  : identified classes of valid input must be accepted.

**Invalid Input**  : identified classes of invalid input must be rejected.

**Functions**  : identified functions must be exercised.

**Output**  : identified classes of application outputs must be exercised.

**Systems**  : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing.

CMRTC 23

##### 6.3 TEST CASES

###### 6.3.1 CLASSIFICATION

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test case ID | Test case name | Purpose | Input | Output |
| 1 | Crowd Counting. | To count People. | The user gives the input in the form of a video or Image. | An output is  no. of people in an Image or video |
| 2 | Crowd Counting. | To count People. | The user walks in a motion towards entry | An output is  no. of people in an Image or video |

CMRTC 24

**7. CONCLUSION**

##### 7. CONCLUSION AND FUTURE SCOPE

##### 7.1 PROJECT CONCLUSION

In this Project, an improved Convolutional Neural Network combined with global density feature is proposed. It is different from existing crowd counting methods. The proposed method focuses on uneven crowd distribution. Moreover, the max-ave pooling and deconvolutional layers are used to generate a more comprehensive density map. The experimental results show that the proposed method achieves competitive performance on different crowd datasets. Due to the high density crowd, some backgrounds will be taken as people by mistakes. It will bring about noise in the estimated density map and influence the counting results. For the future work,we will focus on reducing the noise in the estimated density map and improving the accuracy of counting

##### 7.2 FUTURE SCOPE

* **Improved Accuracy:**

Researchers will likely continue to refine CNN architectures and

training techniques to achieve even higher accuracy in crowd counting. **Real-time Applications:**

Making crowd counting methods faster and suitable for realtime applications, such as crowd management, security monitoring, and event planning, will be a key focus. Optimizing algorithms for real-time processing is crucial.

* **Dataset Diversity:**

Creating and maintaining diverse datasets that represent various crowd

scenarios, including different cultures, demographics, and locations, will be crucial for training robust models.

* **Human-AI Collaboration:**

Developing interfaces that enable crowd counting models to work alongside human operators for improved decision-making and event management.

CMRTC 25

* **Transfer Learning:**

Leveraging transfer learning techniques to adapt pre-trained CNN models to new crowd counting scenarios can save time and computational resources.

CMRTC 26

### 8. BIBLIOGRAPHY

##### 8. BIBLIOGRAPHY

##### 8.1 REFERENCES

1. S. F. Lin, J. Y. Chen, and H. X. Chao, “Estimation of number of people in crowded scenes using perspective transformation,” IEEE Trans. Syst., Man，Cybern. A, Syst., Humans, vol. 31, no.6, pp. 645-654, Nov. 2001.
2. G. Paweł, K. Kowalak, K. Łukasz, and S. Maćkowiak, “Crowd density estimation based on voxel model in multi-view surveillance systems,” in Proc. IEEE Int. Conf. Syst., Signals, Image Process., Sep. 2015, pp. 216- 219.
3. M. Li, Z. Zhang, K. Huang, and T. Tan, “Estimating the number of people in crowded scenes by MID based foreground segmentation and head-shoulder detection,” in Proc. Int. Conf. Pattern Recognit., Dec. 2008, pp. 1-4.
4. T. Zhao, R. Nevatia, and B. Wu, “Segmentation and tracking of multiple humans in crowded environments,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 30, no.7, pp. 1198-1211, Jul. 2008.

.

##### 

##### 8.2 GITHUB LINK

https://github.com/sakethaedula/Crowd-Counting-Method-

CMRTC 27