DEFENCE INSTITUTE OF ADVANCED TECHNOLOGY

(DEEMED UNIVERSITY)
(An Autonomous Organization, DRDO, GOI)
Girinagar, Pune – 4110025

School Of Computer Engineering & Mathematical Sciences

M.Tech Dissertation-1 2022-23

Thesis Evaluation Phase-I presentation

Title:- TEXT DETECTION AND EXTRACTION FROM VIDEO USING DL TECHNIQUES

Under the guidance of :-

Name of DIAT Guide:- Dr. Odelu Ojjela
Asso.Professor

Name of DRDO Guide:- Dr. J V SatyaNarayana
Scientist G, RCI, DRDO

By:-
Name:- **RAMKRISHANA MARRI**
Reg No : 21-27-21

CONTENTS

INTRODUCTION

- With advanced technology, smart devices and high- speed internet Textual content appearing in semantic retrieval of videos, live stream videos, as well as YouTube videos**.**
- The current work provides a comprehensive framework for detecting and recognizing textual content within video frames.
- YouTube is widely used for news, scientific, health, political and educational videos. Videos have turned out to be a great source of information. The text in the video contains a huge amount of information and data, but could not be edited. If this text is converted to editable form, it will be useful information for us and it will be stored efficiently and it will be easier to access it next time.
- The proposed system's operation is simple and user-friendly. if someone gets a video from a website or YouTube from which they want to extract the text. The proposed system take video as input and divides the Video into individual frames and performs text extraction and detection on each frame. Each frame's identified text is saved in a text file.

Highlights

- The given input video is converted into video frames
- Different filters are applied on each frame before text extraction
- Tesseract-OCR was used to extract texts from each video frame

Requirements:

- OpenCV
- Python
- Tesseract-OCR

OBJECTIVE

- The main objective of this project is to use one of the advancements, which is using Deep Learning model with the help of LSTM and CNN architecture which will be used to extract text from a video frame.
- With the help of Python-Tesseract, convert the video into video frames to extract textual content from a video.

LITERATURE SURVEY

- Baseem Bouaziz, Tarek Zlitni, Walid Mahdi [2] explained automatic video text extraction. It performs content based video indexing. This method can detects only static superimposed text.
- Lifang Gu [5] explained text detection in MPEG (Moving Picture Experts Group) video frames. It reduces spatial and temporal data redundancies. This method gives accurate results in MPEG videos.
- Anubhav Kumar, Neeta Awasthi [6], have proposed a method to localize the text data in both image and video files is proposed. It is easy to recognize and extract text from images, but difficult to do so in case of a playing video. Once the system locates the text files on the multimedia file, it is easier to extract them. The drawback of this system is that it takes a very long time in processing long videos.

Problem Statement And System Overview

Problem Statement

There are two types of text occurring in a video

- Natural text
- Superimposed text

Problem Statement And System Overview

Natural Text:

- Natural text is the text which occurs in the video when it is being recorded. These texts are part of scene where video is recorded.
- Example: House number, Car plate number, Name Plate, etc.

Problem Statement And System Overview

Superimposed text:

Superimposed text is the text which is not part of video when it is recorded but is superimposed to give extra information about that particular scene.

Example:

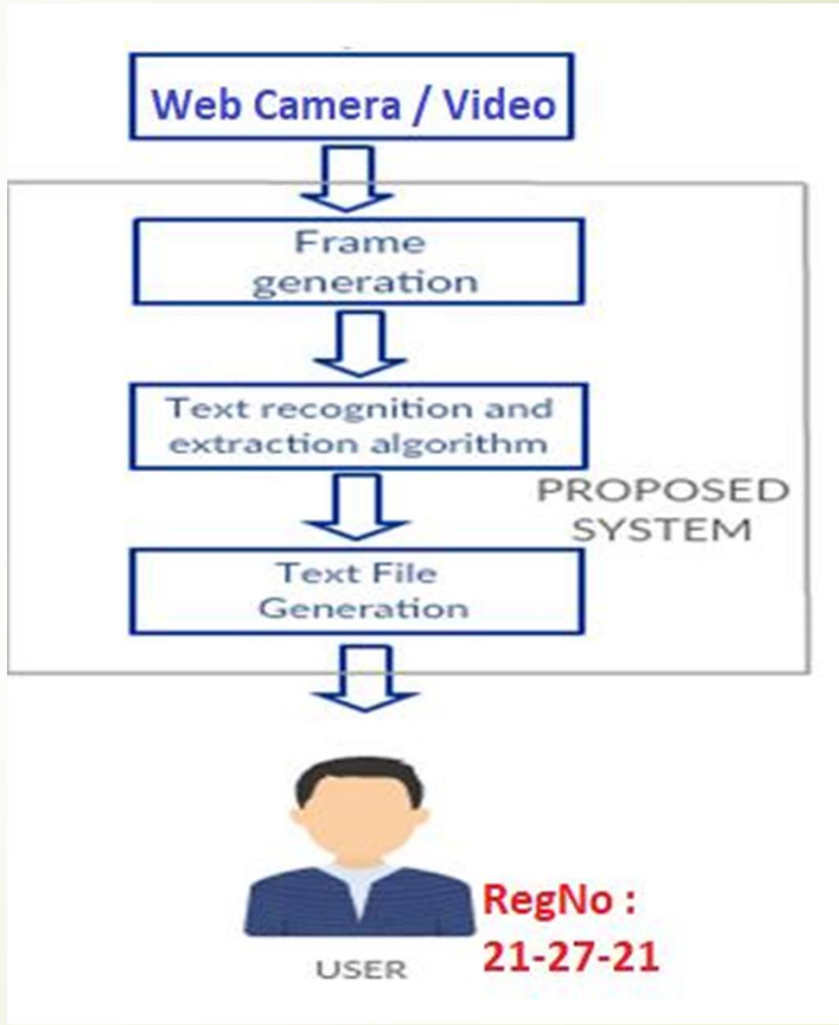
Text occurring in News Video

Problem Statement And System Overview

SYSTEM OVERVIEW

The proposed system has three main components:

- Frame Generation
- Text Recognition and Extraction
- Text File Generation



Problem Statement And System Overview

- ***Frame Generation:*** In this step, the video is converted into frames. Frames are the images of a particular time of a video. These frames can be saved in any image format.

  The people will have two options while converting video to frame. The first is convert entire video and second is converting a selected portion of video.

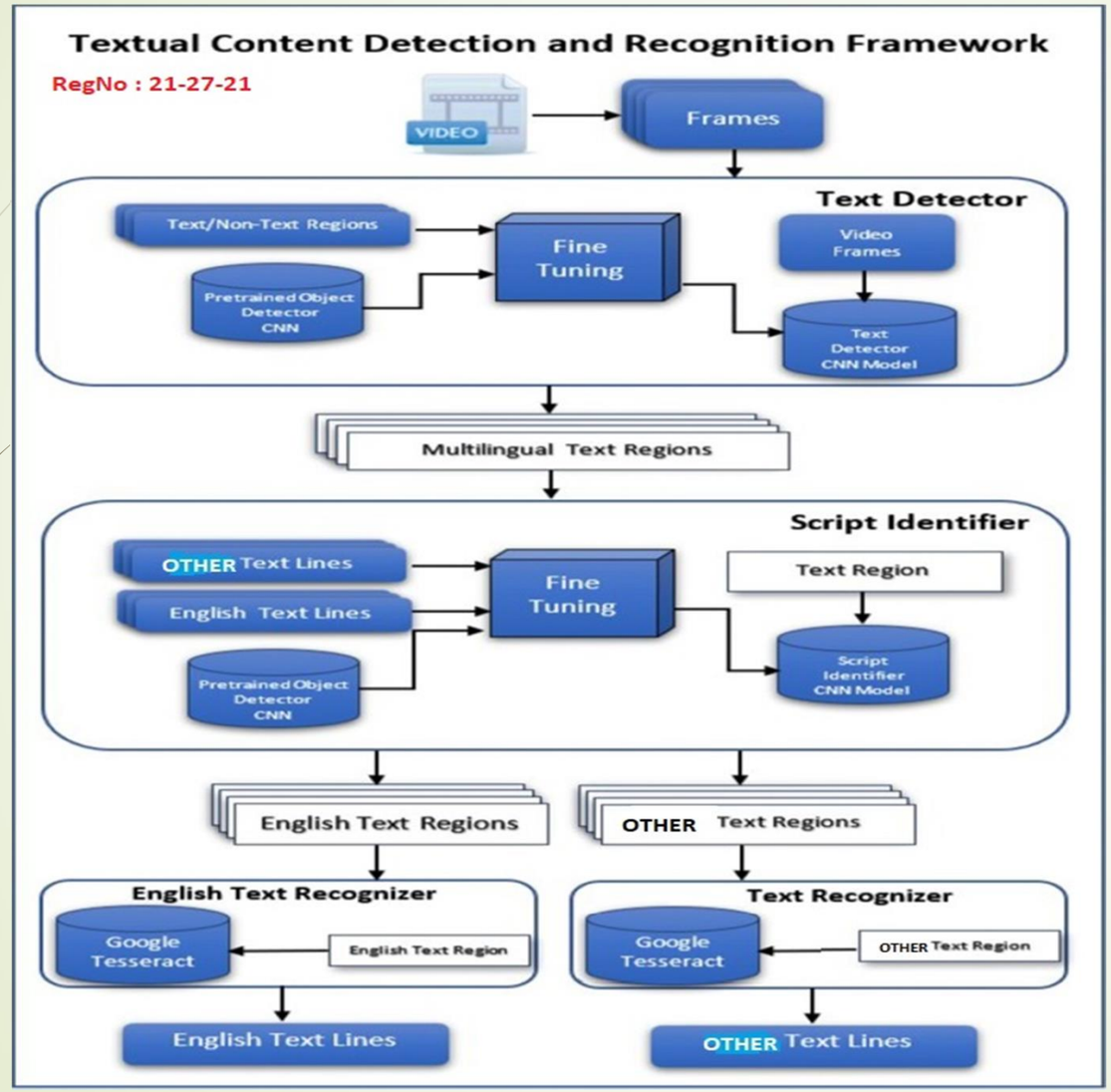
- ***Text Recognition and Extraction:*** This step is applied on every frame. In this step the text region is detected using Tesseract-OCR. The detected text regions are then refined to increase the efficiency of extracting text. The efficiency of detecting text depends on font color, text size, background color and resolution of the video.

- ***Text File Generation:*** The extracted text is stored in a text file. For every frame the generated text is appended to the previous text in the text file and stored. At the end of extracting text from all the images the path of the output file will be given to people. Size of the text file is very less as compared to the size of the video. This saves memory and also makes quicker access to information possible.

METHODOLOGY

- The suggested framework is described in detail in this section and illustrated in the following slide in the flowchart. The three key components comprise the entire system: the text detector, script identifier, and text recognizer.
- The first module, the text detector, must locate and recognize every piece of text in a frame. Since text can appear in many scripts (within the same frame), the text portions that have been found are provided to the script identification module, which splits the text lines according to the script (English and other Indian Languages being the two scripts considered in the present study). The text is then forwarded to the respective recognition engines of each writer, where they convert the images of the text lines into strings that can be used in various situations.

METHODOLOGY



The flowchart illustrates the framework for detecting and recognizing textual content within video frames.

METHODOLOGY

Text Detection

- The proposed framework's first step is the identification of potential text sections in the retrieved video frames. Modern object detectors based on convolutional neural networks (CNN) have been used to recognize textual information in a given frame

Determining Bounding Boxes

- The individual characters will initially be combined into a single connected component before the bounding box of the text region is computed. To eliminate any outliers, this can be done by morphologically closing and then opening.

Text Extraction

- The Tesseract OCR package, which includes an optical character recognition (OCR) engine called **libtesseract** and a command-line tool called Tesseract, is used to implement text extraction. Long Short-Term Memory (LSTM) based OCR engine, which concentrates on line identification and also identifies character pattern, is a new neural network included in Tesseract. The building blocks of recurrent neural networks are the LSTM network. An optical character recognition (OCR) tool in Python called the Python-Tesseract is used to extract text.

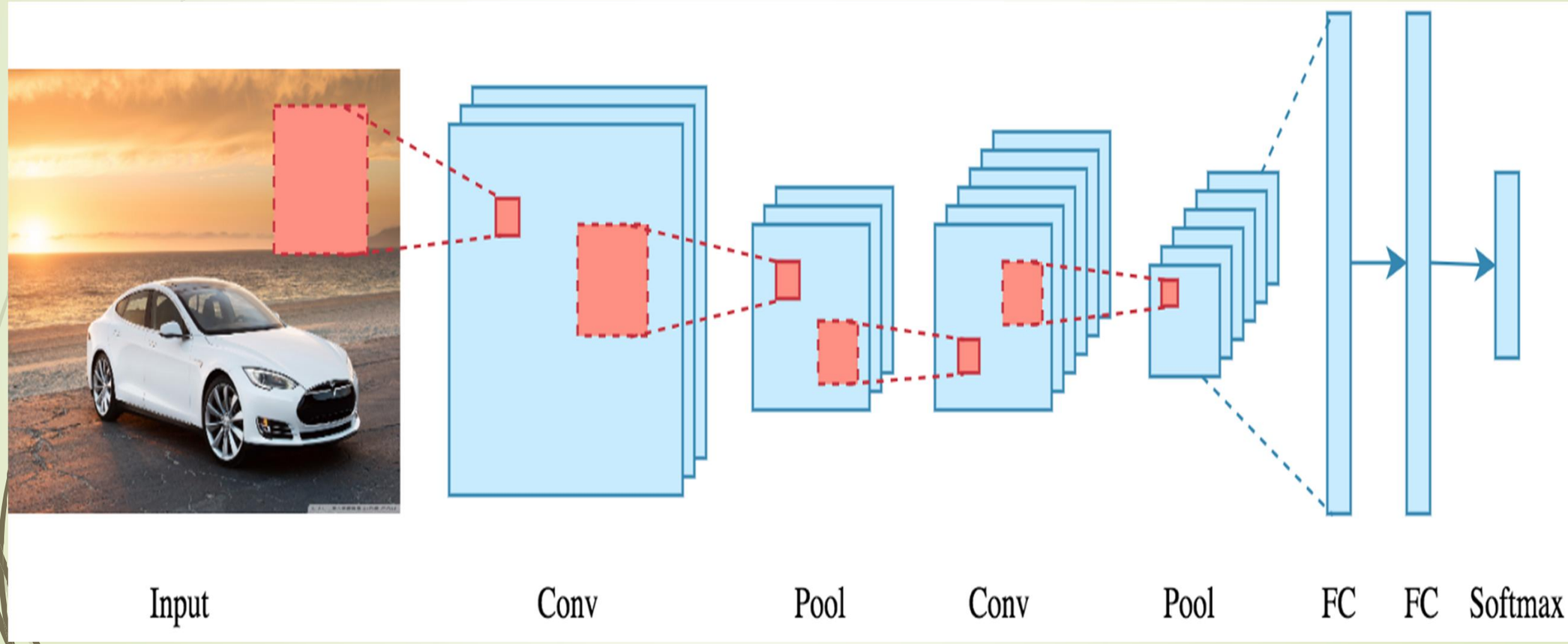
DESCRIPTION OF DEEP LEARNING MODEL USED

Convolutional Neural Network (CNN)

- CNN is a simple deep learning based algorithm that converts the input data to useful representation to do complex image classification. It usually consists of convolutional layers, pooling layers and fully connected layers (known as dense layers) as depicted in Fig. in Next Slide. Convolutional layers learn the feature representation of input image by assigning appropriate weights and biases to each neuron connecting with some neurons of preceding layer.
- The resultant is passed to activation function such as Rectified Linear Unit (ReLU). Pooling layer plays role to reduce features dimensionality. In the end, a fully connected layer combines all the learned features of previous layers to formulate the prediction in output layer (classification layer) with help of probabilities computed by softmax layer

DESCRIPTION OF DEEP LEARNING MODEL USED

Convolutional Neural Network (CNN)



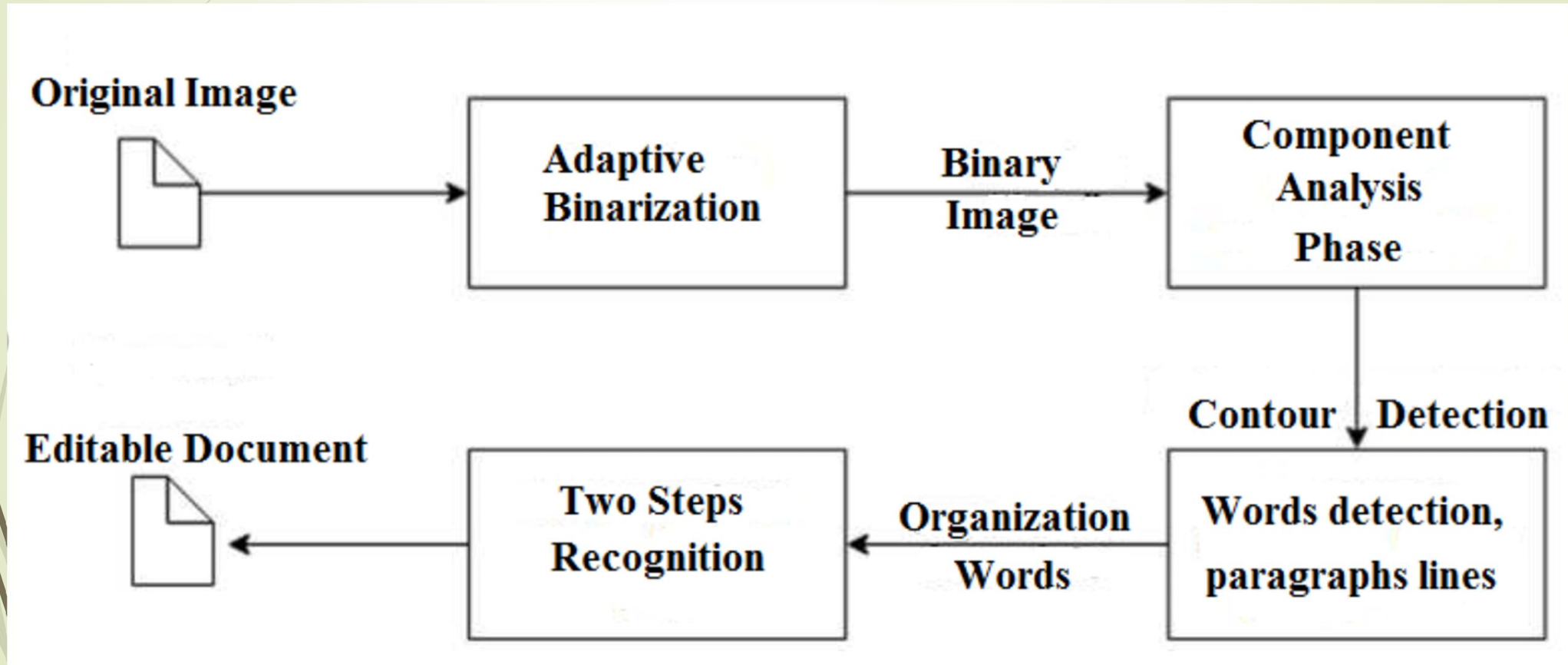
DESCRIPTION OF DEEP LEARNING MODEL USED

Long short-term memory (LSTM)

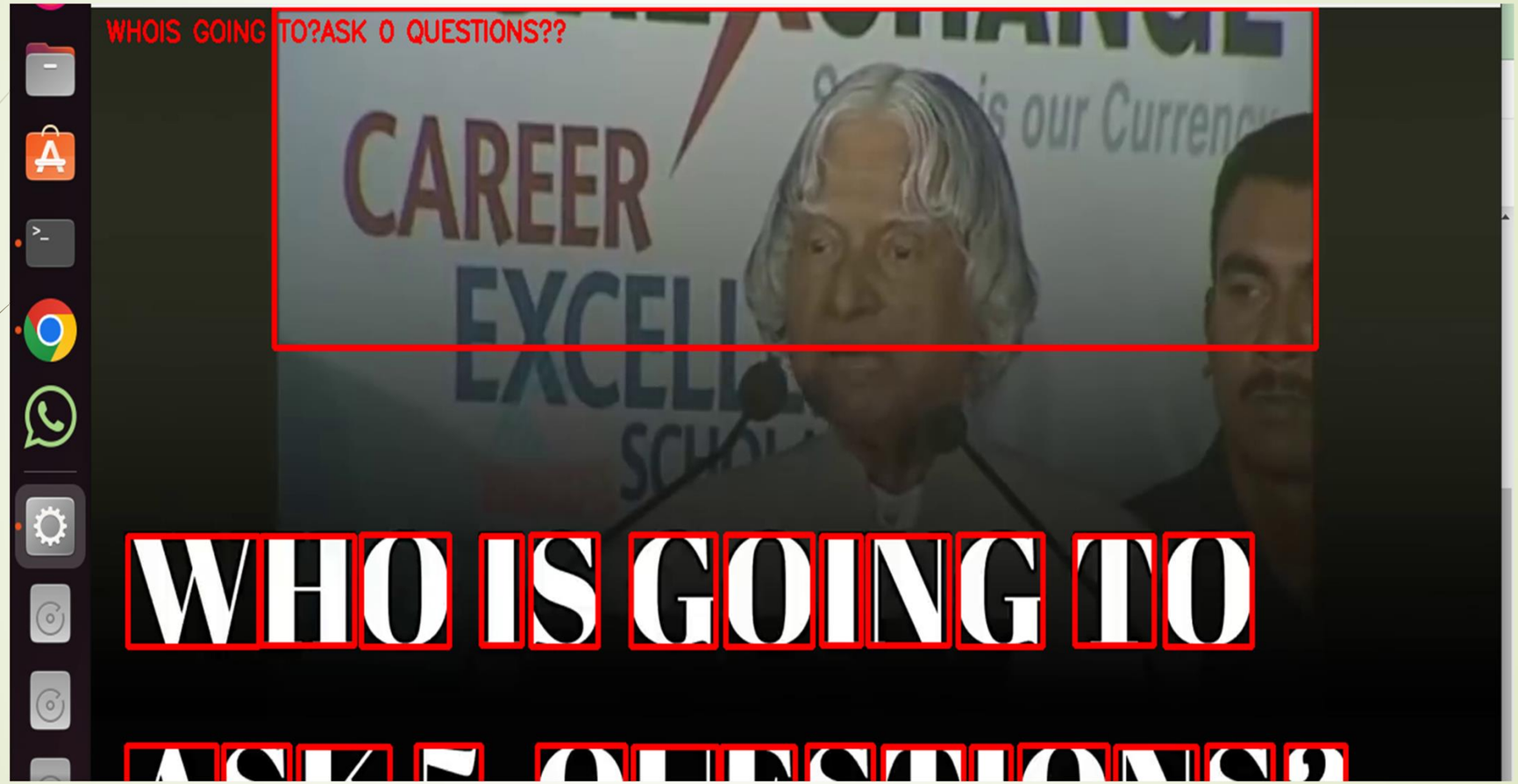
- LSTMs are great at learning sequences but slow down a lot when the number of states is too large. There are empirical results that suggest it is better to ask an LSTM to learn a long sequence than a short sequence of many classes. Tesseract developed from OCRopus model in Python which was a fork of a LSMT in C++, called CLSTM. CLSTM is an implementation of the LSTM recurrent neural network model in C++, using the Eigen library for numerical computations.
- Modernization of the Tesseract tool was an effort on code cleaning and adding a new LSTM model. The input image is processed in boxes (rectangle) line by line feeding into the LSTM model and giving output. In the image below we can visualize how it works

DESCRIPTION OF DEEP LEARNING MODEL USED

Long short-term memory (LSTM)

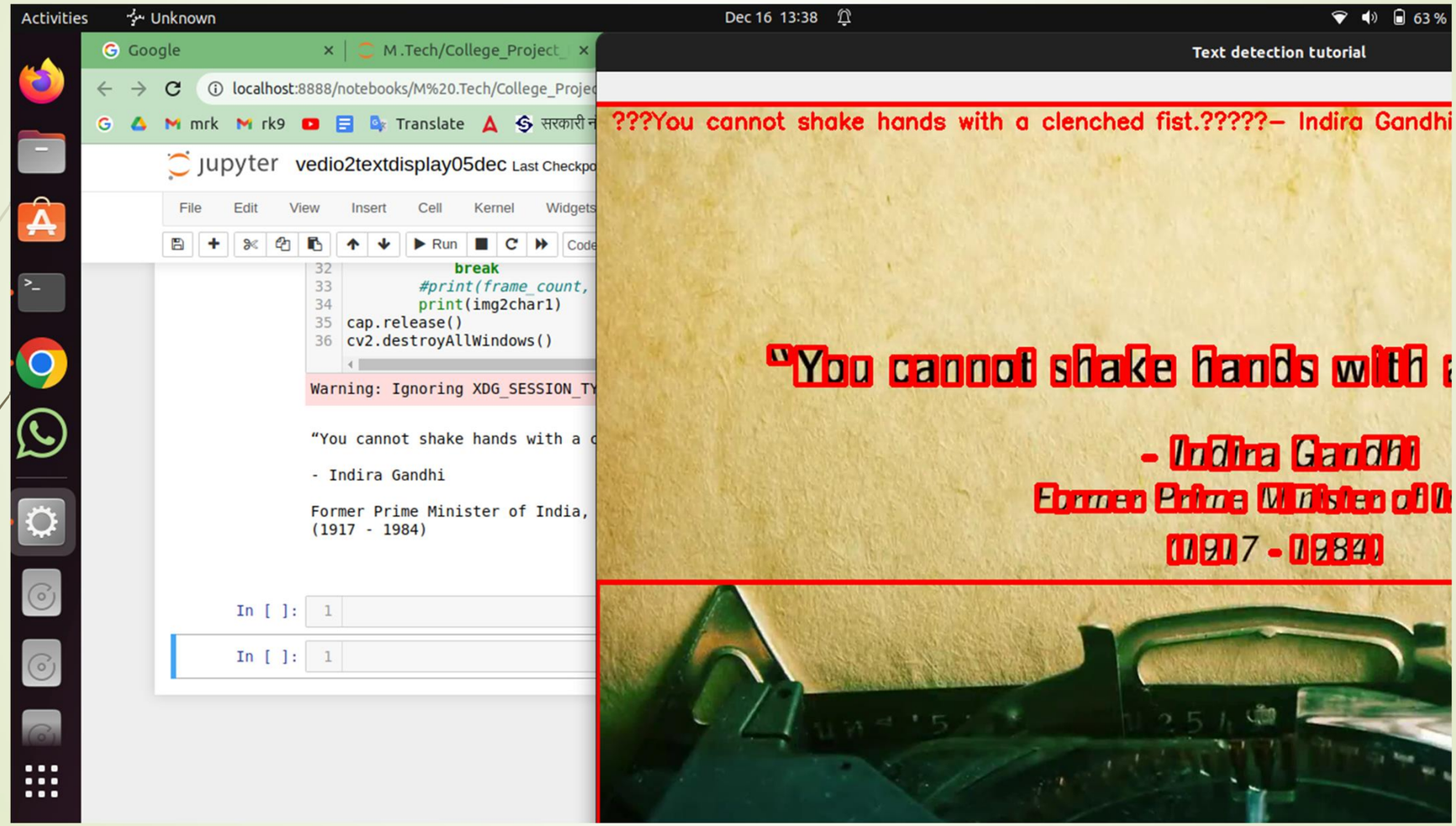


RESULTS



RESULTS

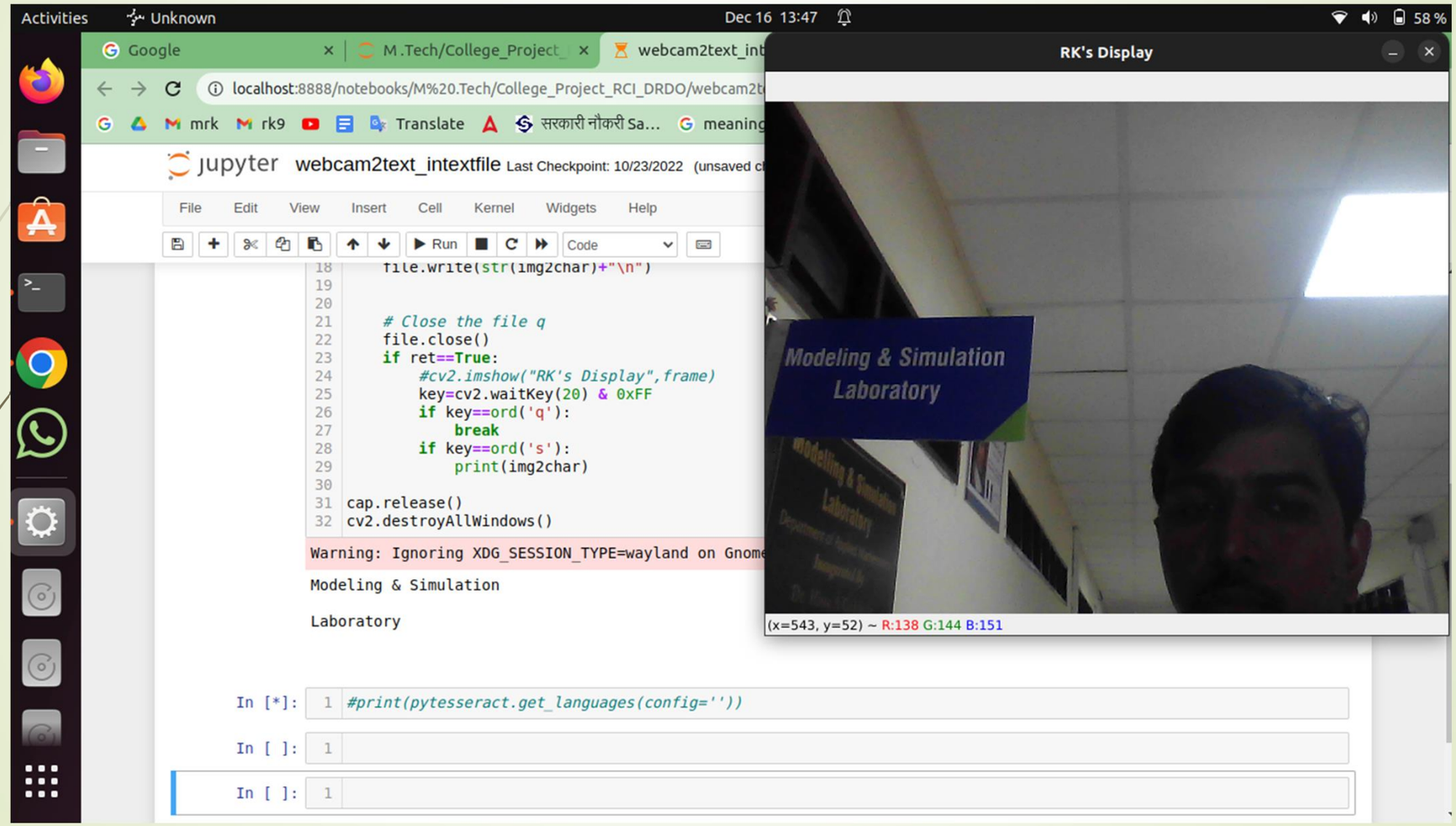
RESULTS


Activities
Unknown
Dec 16 13:38
63 %
Google
M .Tech/College_Project
localhost:8888/notebooks/M%20.Tech/College_Proje
mrk
rk9
Translate
jupyter
vedio2textdisplay05dec
File
Edit
View
Insert
Cell
Kernel
Widgets
Run
32 break
33 #print(frame_count,
34 print(img2char1)
35 cap.release()
36 cv2.destroyAllWindows()
Warning: Ignoring XDG_SESSION_TY
"You cannot shake hands with a c
- Indira Gandhi
Former Prime Minister of India,
(1917 - 1984)
In [ ]: 1
In [ ]: 1
Text detection tutorial
???You cannot shake hands with a clenched fist.?????– Indira Gandhi
"You cannot shake hands with a
- Indira Gandhi
Former Prime Minister of I
(1917 - 1984)

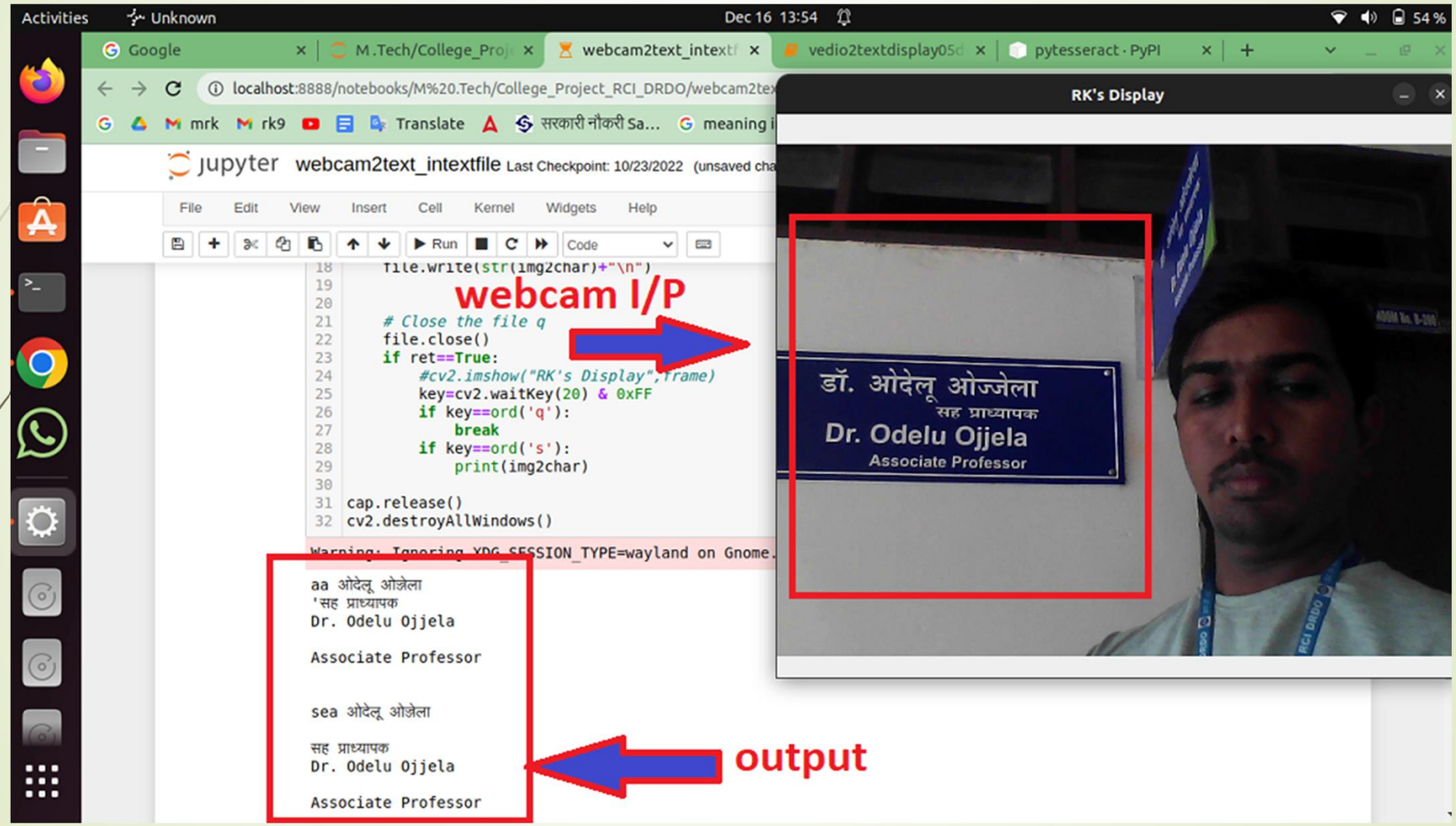
RESULTS


File
Edit
View
Insert
Cell
Run
QUICK |
BROWN
FOX
JUMPS
OVER
THE
LAZY
DOG
THE
THE

RESULTS

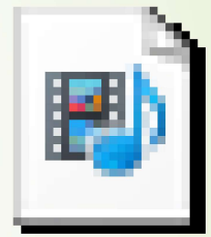

Activities
Unknown
Dec 16 13:47
58 %
Google
M.Tech/College_Project
webcam2text_int
localhost:8888/notebooks/M%20.Tech/College_Project_RCI_DRDO/webcam2t
mrk
rk9
Translate
jupyter
webcam2text_intextfile
Last Checkpoint: 10/23/2022
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Run
Code
file.write(str(img2char)+"\n")
# Close the file q
file.close()
if ret==True:
#cv2.imshow("RK's Display",frame)
key=cv2.waitKey(20) & 0xFF
if key==ord('q'):
break
if key==ord('s'):
print(img2char)
cap.release()
cv2.destroyAllWindows()
Warning: Ignoring XDG_SESSION_TYPE=wayland on Gnome
Modeling & Simulation
Laboratory
RK's Display
Modeling & Simulation Laboratory
(x=543, y=52) ~ R:138 G:144 B:151
In [*]:
#print(pytesseract.get_languages(config=''))
In [ ]:
In [ ]:

RESULTS


webcam I/P
output
RK's Display
डॉ. ओदेलू ओज्जेला
सह प्राध्यापक
Dr. Odelu Ojjela
Associate Professor
aa ओदेलू ओज़ेला
'सह प्राध्यापक
Dr. Odelu Ojjela
Associate Professor
sea ओदेलू ओज़ेला
सह प्राध्यापक
Dr. Odelu Ojjela
Associate Professor

RESULTS

Video1a.mp4



video2a.mp4

FUTURE WORK

- In our further work on this problem, we intend to develop the system will be optimized to work on live streams in addition to archive videos. The study can also be extended to include additional scripts by integrating their respective OCRs
- we intend to develop the system will be receive the text from video and convert into required Language
- we intend to develop the system will be receive the text from video and converted output in text format as well as voice (audio) format.

REFERENCES

- [1] Kiran Agre, Sairaj Gaonkar Ankur Chheda, and Prof. Mahendra Patil. Text recognition and extraction from video, international journal of engineering research technology. page (Volume 5 – Issue 01). IJERT, 2017.
- [2] Baseem Bouaziz, Tarek Zlitni, and Walid Mahdi. avitext: Automatic video text extraction; a new approach for video content indexing application. In 2008 3rd International Conference on Information and Communication Technologies: From Theory to Applications, pages 1–5. IEEE, 2008.
- [3] Datong Chen, Jean-Marc Odobez, and Herve Bourlard. Text detection and recognition in images and video frames. Pattern recognition, 37(3):595–608, 2004.

REFERENCES

- Hassani, H., Ershadi, M. J., & Mohebi, A. (2022). LVTIA: A new method for keyphrase extraction from scientific video lectures. *Information Processing & Management*, *59*(2), 102802.
- [4] R Deepa and Kiran N Lalwani. Image classification and text extraction using machine learning. In 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), pages 680–684. IEEE, 2019.
- Yu, H., Huang, Y., Pi, L., Zhang, C., Li, X., & Wang, L. (2021). End-to-end video text detection with online tracking. *Pattern Recognition*, *113*, 107791.

1. https://nanonets.com/blog/ocr-with-tesseract\

2. https://pypi.org/project/pytesseract/

THANK YOU!!!