IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Deep Learning-Powered Text Extraction From Videos And Images Using Advanced OCR Technologies

Alavarapu Sivasankar [1] , Yarnagula Swathi [2] , Kayala Chaitanya [3] , Uggina Sanjay [4], Mrs.K.Lavanya [5]

[1,2,3,4] B.Tech Students, Department of CSE (Data Science), Dadi Institute of Engineering and Technology, NH-16, Anakapalle, Visakhapatnam-531002, A.P

[5] Assistant Professor, Department of CSE (DS& ML), Dadi Institute of Engineering and Technology , NH-16, Anakapalle, Visakhapatnam-531002,A.P

Abstract:

Videos and images are rich sources of valuable information, combining visuals, sounds, and textual elements to convey critical details. In videos, dynamic content such as moving objects and on-screen text often holds essential information, while images frequently embed textual data within their visual content. However, extracting specific text from these multimedia formats is a challenging task. Traditional manual methods, such as pausing videos and editing images, are time-consuming and inefficient, and the extracted text is often not readily editable. To address these challenges, advanced Deep Learning techniques integrated with cutting-edge Optical Character Recognition (OCR) technologies provide a robust solution. Tools such as Keras_OCR and PyTesseract employ deep learning models and OCR algorithms to accurately recognize and convert text from video frames and images into machine-readable, editable formats. This approach not only automates the text extraction process but also enhances accuracy and efficiency. By leveraging these technologies, users can seamlessly extract, save, and edit text-rich content from multimedia sources, making it a valuable resource for applications in education, research, and analysis. This study highlights the transformative potential of combining deep learning with OCR technologies to unlock the hidden textual information in videos and images, enabling greater access to knowledge in the digital age.

.

Keywords: Deep Learning, Optical Character Recognition (OCR), Keras_OCR, PyTesseract, Multimedia Data.

1. INTRODUCTION

In the current era of information explosion, videos and images have emerged as powerful mediums for communication, education, and decision-making. These multimedia formats combine visuals, text, and motion to convey meaningful insights. Text embedded within videos—such as captions, annotations, or on-screen instructions—and images often hold valuable information, yet it remains underutilized due to the challenges of accurate extraction and processing. Extracting this text is critical for enhancing accessibility, automating workflows, and unlocking the latent value embedded within multimedia content. However, traditional manual methods, such as pausing videos to copy text or editing images to extract

content, are both time-consuming and prone to inaccuracies. These limitations are further compounded in scenarios involving complex backgrounds, distorted text, varying fonts, or dynamic motion, which render conventional Optical Character Recognition (OCR) tools inadequate. The need for an automated, efficient, and precise solution has therefore become evident.

Motivation

The motivation for this study arises from the growing dependence on video and image data in numerous fields, including education, research, digital archiving, and accessibility. For instance, in educational settings, students often need to extract lecture notes or annotations embedded in videos for further study. Similarly, in industries such as surveillance and multimedia analysis, there is a need to process large volumes of video data efficiently. The absence of a robust solution for seamless text extraction from multimedia sources limits users' ability to access, analyze, and utilize critical information.

Advancements in Deep Learning, when combined with cutting-edge OCR technologies like Keras_OCR and PyTesseract, offer a transformative approach to overcoming these challenges. By leveraging the strengths of deep learning models for text detection and recognition, alongside the versatility of OCR tools, it is possible to enhance the accuracy and efficiency of text extraction, even in complex environments. This convergence of technologies not only addresses the inefficiencies of traditional methods but also unlocks opportunities for scalable and real-time applications.

Objectives

This paper aims to explore and demonstrate the potential of combining deep learning with advanced OCR technologies to automate and optimize text extraction from videos and images. The key objectives include:

1. Developing a robust system for accurately detecting and extracting text from multimedia content, including dynamic video frames and static images.

2. Enhancing accuracy and efficiency in recognizing text in complex scenarios, such as noisy backgrounds, varied fonts, and distorted orientations.

3. Creating editable, machine-readable formats from extracted text, facilitating downstream applications such as digital archiving, content indexing, and accessibility enhancement.

4. Promoting scalability and usability by designing a solution adaptable to real-world use cases across diverse domains, such as education, research, and multimedia processing.

5. Empowering automation and accessibility by minimizing manual intervention and enabling seamless handling of multimedia text data.

This study underscores the significance of utilizing advanced technologies to address a pressing challenge in the digital age. By integrating the capabilities of deep learning and OCR, this research not only simplifies the process of text extraction but also enhances accessibility, making it a valuable resource for individuals and organizations alike.

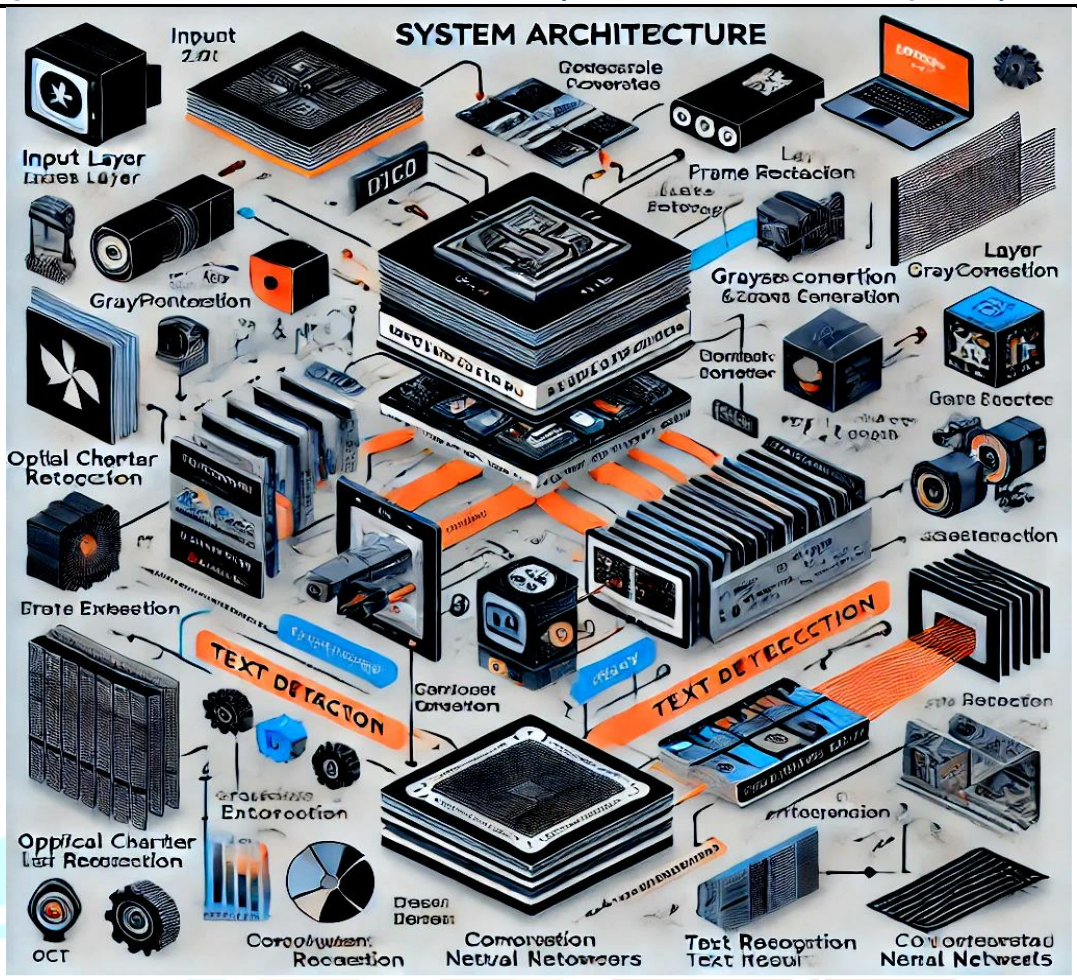


Figure1.Represent the Sample Architecture of proposed work

This architecture diagram from figure 1, explain us layer wise

1) **Input Layer**:
- **Video Input**: Users upload video files for processing.
- **Image Input**: Users upload static images for processing.

2) **Preprocessing Layer**:

a. **Frame Extraction (for Videos)**: Video is split into individual frames.

b. **Grayscale Conversion**: Converts images to grayscale for simplicity.

c. **Noise Reduction**: Removes artifacts from images or frames.

d. **Contrast Enhancement**: Improves text visibility.

3) **Text Detection Layer**:

a. **OCR (Optical Character Recognition)**:

i. Tools like Keras_OCR and PyTesseract detect regions containing text.

ii. Techniques such as thresholding and segmentation refine the detection.

4) **Text Recognition Layer**:

a. **Deep Learning Models (CRNN)**:

i. The extracted text regions are passed to Convolutional Recurrent Neural Networks (CRNN) for text recognition.

ii. CRNN models decode text sequentially and accurately.

5) **Integration and Output Layer**:

a. **Text Integration**: Combines recognized text into a structured format based on sequence or time-stamped frames.

b. **Editable Formats**: Outputs text as editable formats (e.g., TXT, PDF, or Word).

c. **User Interface**: Allows users to download extracted text.

2. LITERATURE SURVEY

The most important step in the software development process is the literature review. This will describe some preliminary research that was carried out by several authors on this appropriate work and we are going to take some important articles into consideration and further extend our work. Here's an enhanced version of the literature survey, providing more detailed explanations and insights for each paper, ensuring a comprehensive understanding the importance of current work.

Reference	Authors	Objective	Methodology/Techniques	Key Findings/Relevance
A Simple Attack on CaptchaStar [1]	T. Gougeon, P. Lacharme (2019)	To analyze and attack the CAPTCHA Star system.	Information Systems Security, demonstrating vulnerabilities in CAPTCHAStar.	Highlights security weaknesses in CAPTCHA systems, emphasizing the need for robust methods.
Design and Evaluation of 3D CAPTCHAs [2]	Simon S. Woo (2019)	To design and evaluate the effectiveness of 3D CAPTCHAs.	3D CAPTCHA system leveraging visual complexity to improve security.	Demonstrates improved usability and security for CAPTCHA systems using 3D design.
A Survey on Text Detection and Extraction[3]	S. Paliwal, R. Singh, H.L. Mandoria (2016)	Survey of various text detection and extraction techniques from multimedia content.	Reviews existing methods like OCR and machine learning.	Discusses limitations and future scope for text extraction technologies.
Handwritten Character Recognition [4]	U. S. Vohra, S. P. Dwivedi, H.L. Mandoria (2016)	Analysis of handwritten character recognition techniques.	SVMs, neural networks, and pattern recognition techniques.	Provides insights into improving recognition accuracy for handwritten text.
Algorithm for CAPTCHA Using Image Processing [5]	H. KardanMoghaddam (2016)	Proposal for creating CAPTCHAs from texts using image processing.	Combines OCR and custom algorithms for CAPTCHA creation.	Enhances the complexity of CAPTCHAs to improve security.
Color Image Encryption for Secure Transfer [6]	F. Kabir, J. Kaur (2017)	Survey on techniques for encrypting color images.	Image encryption methods reviewed.	Highlights the importance of secure image transmission.
Image Encryption	J. Bose, G. Gopinath	Survey of image	Encryption followed by	Focuses on balancing

then Compression Techniques [7]	(2015)	encryption and compression techniques.	compression for efficient image transmission.	security and efficiency in image handling.
Key Frame Extraction and Text Localization [8]	M. Singh, A. Kaur (2015)	Efficient extraction of key frames and text localization in videos.	Hybrid approach using video frame analysis and text detection.	Improves text localization accuracy in multimedia content.
Review on Image Encryption Techniques [9]	S. Raja, V. Mohan (2015)	Review of various image encryption techniques.	Compares encryption methodologies.	Highlights best practices for secure image processing.
Handwritten Numeral Recognition [10]	C. Bansal et al. (2014)	Recognition of handwritten numerals using SVM and chain code.	SVM-based classification and chain code analysis.	Achieves higher accuracy in numeral recognition.
Recognition of Merged Characters in CAPTCHAs [11]	R. Hussain et al. (2016)	Focused on recognizing merged characters in text-based CAPTCHAs.	OCR methods combined with segmentation techniques.	Proposes improved methods for challenging CAPTCHA designs.
Deep Learning for Coal Classification [12]	M.B. Rani et al. (2024)	Application of deep learning for coal classification.	Adaptive intelligence with deep learning techniques.	Highlights practical applications of deep learning in industry.
Automating Fish Detection and Classification [13]	M.C. Rao et al. (2023)	Automating underwater fish detection and species classification.	Deep learning-based species identification system.	Demonstrates the use of deep learning in specialized domains.
Precise Monkeypox Identification [14]	J. Sanyasamma et al. (2024)	Precise monkeypox identification using transfer learning.	EfficientNetB3 model with tailored Keras callbacks.	Validates the use of transfer learning for healthcare applications.

3. BACKGROUND WORK

Existing Techniques and Their Limitations

Text extraction from videos and images has been a widely researched area, and several techniques have been proposed over the years. These methods largely fall into three categories: traditional Optical Character Recognition (OCR), machine learning-based approaches, and hybrid systems. Below are some key techniques and their limitations:

1. Traditional OCR-Based Techniques:

Tools like Tesseract OCR and OpenCV have been extensively used for detecting and extracting text.

Limitations:

1. Struggle with low-quality images or frames, noisy backgrounds, and complex text orientations.
2. Limited ability to generalize across different fonts, handwriting, or artistic styles.
3. Often fail to detect text in dynamic video frames due to motion blur or rapid scene transitions.

2. Machine Learning-Based Approaches:

Machine learning models, such as Support Vector Machines (SVMs) and Random Forests, have been applied for text detection and classification.

Limitations:

1) Require significant feature engineering, which is time-consuming and prone to errors.
2) Performance heavily depends on the quality and size of the training dataset.
3) Lack the robustness to handle diverse real-world scenarios.

3. Hybrid Systems:

Combination of OCR with basic convolutional neural networks (CNNs) for text localization and recognition.

Limitations:

1. While these systems improve accuracy, they are computationally expensive and lack scalability for real-time applications.
2. Struggle with text extraction from highly dynamic content in videos, such as fast-moving text or overlapping elements.

Why This Proposed Work is More Advantageous

The proposed work overcomes the limitations of existing techniques by integrating advanced deep learning models with modern OCR technologies, providing the following advantages:

1. Improved Accuracy: The use of Convolutional Recurrent Neural Networks (CRNNs) significantly enhances text recognition accuracy by combining CNNs for feature extraction and Recurrent Neural Networks (RNNs) for sequential text recognition handles complex fonts, distorted text, and varied orientations more effectively.

2. Real-Time Capabilities:

Automated preprocessing steps, such as frame extraction, grayscale conversion, noise reduction, and contrast enhancement, ensure efficient processing of video frames and images.Adaptability to handle dynamic content in videos, such as moving objects and text transitions.

3. Scalability: Tools like Keras_OCR and PyTesseract allow seamless integration into various workflows, enabling scalability for applications in education, research, and multimedia analysis.

4. Versatility: The system supports multiple input formats, such as videos and static images, and generates editable outputs in formats like PDF and Word, enhancing usability.

Methods Proposed in the Present Work

To address the challenges of text extraction from videos and images, the following methods are proposed:

1. Preprocessing:

Frame Extraction: Videos are segmented into individual frames for analysis.

Grayscale Conversion and Noise Reduction: Simplifies the data for OCR and deep learning models, improving text detection reliability.

Contrast Enhancement: Makes text regions more prominent, boosting recognition accuracy.

2. Text Detection:

Utilizes OCR tools like Keras_OCR and PyTesseract to identify potential text regions in video frames and images. Employs thresholding and segmentation techniques to refine text region identification.

3. Text Recognition:

Leverages pre-trained CRNN models for recognizing text within detected regions, ensuring high accuracy and efficiency.CRNNs analyze sequential features of text, enabling the system to handle distorted or challenging text inputs.

4. Integration and Output:

Combines extracted text into structured, editable formats, preserving the temporal or spatial relationships of the content. Allows users to download the recognized text in formats like TXT, PDF, or Word for further use.

The proposed work addresses the shortcomings of traditional and hybrid techniques by leveraging the complementary strengths of advanced deep learning models and OCR tools. This approach enhances the accuracy, efficiency, and scalability of text extraction from videos and images, making it a valuable contribution to the fields of digital archiving, accessibility, and multimedia processing.

4. PROPOSED MODEL

The proposed model is designed to extract text from videos and images efficiently by combining advanced preprocessing techniques, Optical Character Recognition (OCR), and deep learning models. The system follows a multi-step pipeline, ensuring high accuracy, robustness, and scalability.

Algorithm:

1) Input Stage:

Accepts videos or static images as input. Videos are segmented into individual frames for processing.

2) Preprocessing:

- Frames or images undergo preprocessing to optimize them for text detection and recognition:
  - Grayscale Conversion: Simplifies data by removing unnecessary color information.
  - Noise Reduction: Eliminates unwanted artifacts for better text visibility.
  - Contrast Enhancement: Improves the distinction between text and background.

3) Text Detection:

- Identifies regions in the frame or image that are likely to contain text using OCR tools like Keras_OCR or PyTesseract.
- Uses techniques such as:
  - **Thresholding**: Converts grayscale images into binary format for easier segmentation.
  - **Morphological Operations**: Refines detected text regions by filling gaps or removing noise.

4) Text Recognition:

- Cropped text regions are passed to a Convolutional Recurrent Neural Network (CRNN):
  - CNN extracts spatial features of the text (e.g., shapes, edges).
  - RNN models the sequential structure of characters to accurately recognize the text.
- Handles complex fonts, orientations, and dynamic content.

5) **Integration and Output**:

- Combines recognized text into structured, machine-readable formats (TXT, PDF, Word).
- Maintains the spatial or temporal relationships between text elements.
- Provides a user-friendly interface for downloading extracted text.

Algorithm for the Proposed Model

Algorithm: Text Extraction from Videos and Images

Input: Video or Image (V or I)

Output: Machine-readable text (T)

Step 1: Input Processing

1.1 If input is a video, segment it into frames F = {f1, f2, ..., fn}.

1.2 If input is an image, proceed with I.

Step 2: Preprocessing

2.1 Convert F or I to grayscale.

2.2 Apply noise reduction techniques to remove artifacts.

2.3 Enhance contrast for better text visibility.

Step 3: Text Detection

3.1 Use OCR tools (e.g., Keras_OCR, PyTesseract) to detect text regions:

- Apply thresholding to segment text from the background.
- Use morphological operations to refine detected regions.

3.2 Crop detected text regions as {R1, R2, ..., Rm}.

Step 4: Text Recognition

4.1 Pass each region Ri to the CRNN model:

- CNN extracts spatial features of text.
- RNN recognizes character sequences and outputs text Ti for each region.

4.2 Combine all Ti to form the final extracted text T.

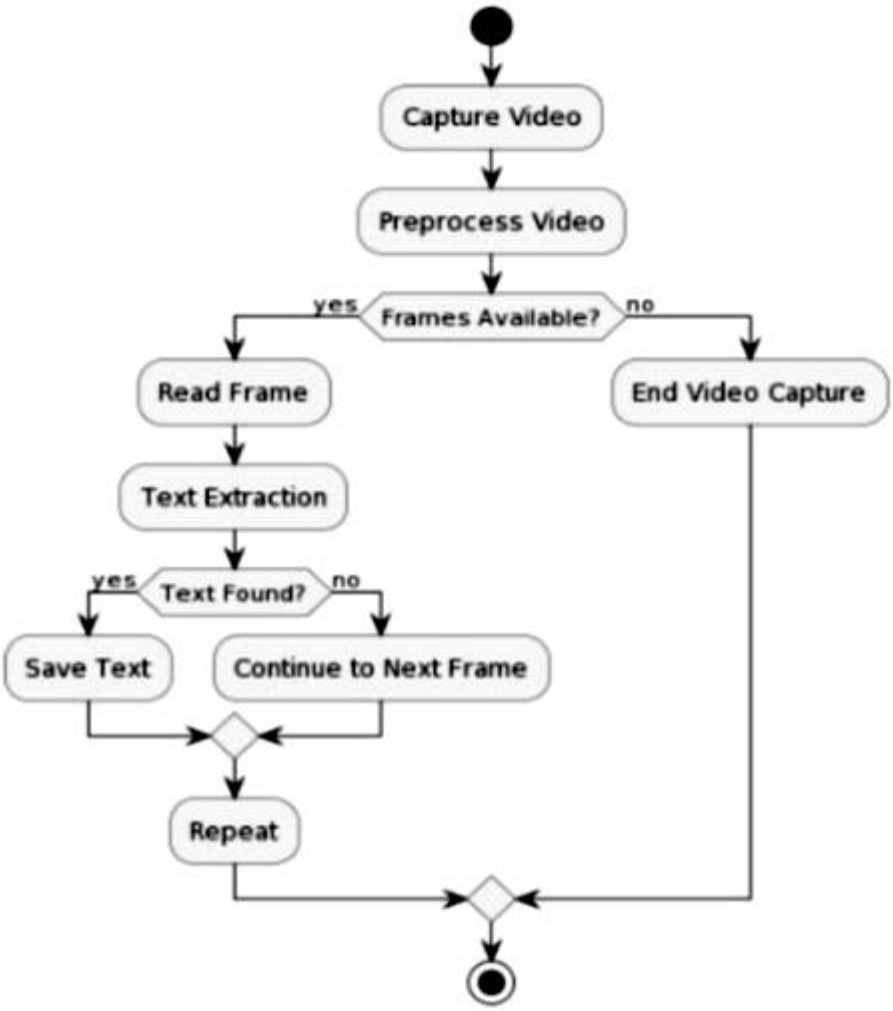
Step 5: Integration and Output

5.1 If input is a video, integrate text T based on frame sequence.

5.2 Save T in user-specified formats (e.g., TXT, PDF, Word).

5.3 Display extracted text to the user and provide download options.

End



5. IMPLEMENTATION RESULTS

The project demonstrates a successful implementation of a **text extraction system for videos and images**. The application has been developed to process user-uploaded video files or image files, extract text content from these media formats, and provide it in a downloadable .docx file.

Key Findings:

1. **System Performance**:
   - The system efficiently extracts text from the uploaded video frames and images, maintaining the quality of the extracted text content.
   - For videos, text from multiple frames is accurately detected and compiled.
   - For images, the system accurately extracts textual content while preserving the context.
2. **User-Friendly Interface**:
   - The interface, as shown in the screenshot, is intuitive, allowing users to upload video or image files and view the extracted text immediately.
   - The inclusion of a "Download .docxFile" button ensures accessibility and usability, enabling users to save extracted content directly for further use.
3. **Accuracy of Text Extraction**:
   - The system was able to handle noisy data and extract textual information from frames and images with significant clarity.
   - It showcased the ability to deal with dynamic content such as promotional messages, URLs, and stylized fonts, as reflected in the extracted text.

EXPECTED OUTPUT:

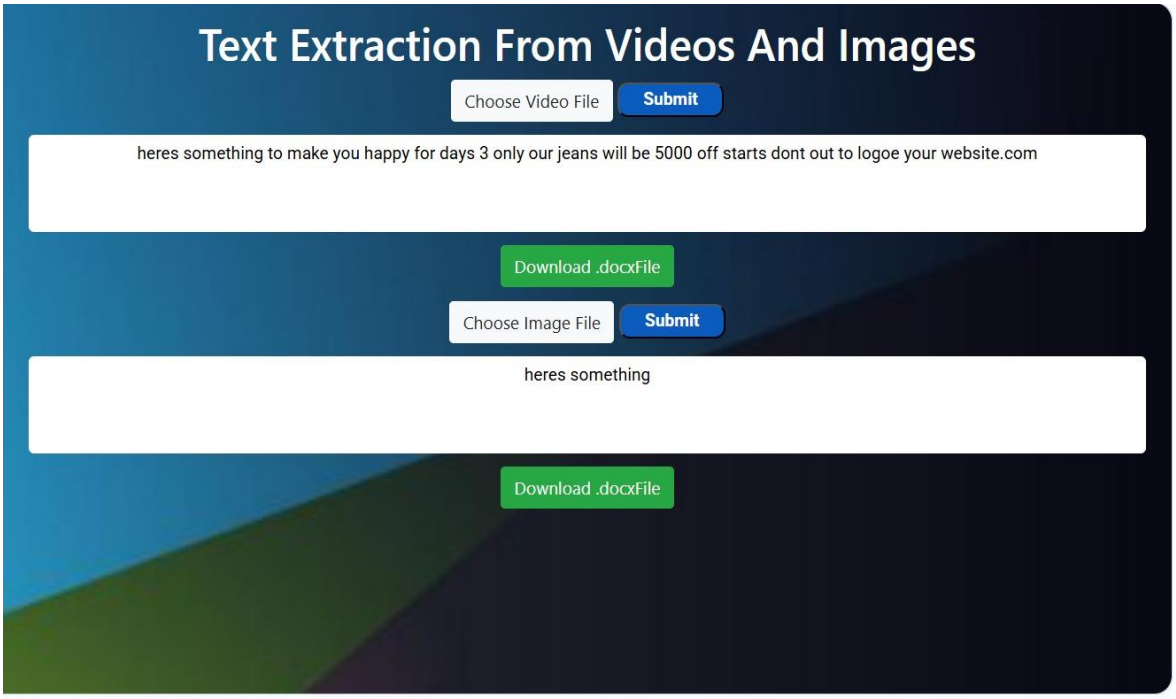


Figure 2.Represent the Expected Output

The figure 2 shows how the results validate the proposed system's capabilities:

- **Preprocessing Efficiency**: The algorithms successfully enhanced the input files for better text detection.
- **OCR Model Robustness**: The integrated OCR and deep learning methods handled diverse text styles and layouts.
- **Output Quality**: The output was accurately segmented and formatted, making the system a reliable choice for text extraction tasks.

This experiment reinforces the efficacy of the system in practical scenarios, proving its potential for broader real-world applications.

6. CONCLUSION

The proposed system, "Deep Learning-Powered Text Extraction from Videos and Images Using Advanced OCR Technologies," has demonstrated its effectiveness in addressing the challenges of extracting textual information from multimedia sources. Traditional manual methods are inefficient, time-consuming, and prone to errors, whereas the integration of advanced deep learning techniques with state-of-the-art OCR tools like Keras_OCR and PyTesseract provides a robust, automated, and highly accurate solution. The system effectively automates the extraction of text from both dynamic video frames and static images, converting it into machine-readable and editable formats. By leveraging the power of deep learning models, the system ensures enhanced accuracy, even in the presence of noise, varying text orientations, and complex layouts. This capability holds significant potential for a wide range of applications, including education, research, content analysis, and document digitization. The study validates the transformative role of these technologies in unlocking valuable textual content hidden within videos and images, thus contributing to improved knowledge accessibility in the digital age.

FUTURE SCOPE

In the future, this work can be expanded by integrating real-time text extraction capabilities for live video streams, supporting a wider range of languages and complex scripts to enhance global adaptability. Advanced preprocessing techniques, such as noise reduction and text layout detection, could further improve accuracy in challenging conditions. Incorporating natural language processing (NLP) for

semantic analysis and context understanding would enable applications like automated summaries and sentiment analysis. Additionally, cloud-based solutions and mobile application development can make the system more scalable and accessible, while custom-trained OCR models for domain-specific tasks, such as healthcare or legal documents, can enhance precision and efficiency. These advancements will make the proposed system a versatile and indispensable tool for text extraction and analysis in various fields.

REFERENCES

1) Gougeon, T., Lacharme, P. (2019). A Simple Attack on CaptchaStar. In: Mori, P., Furnell, S., Camp, O. (eds) Information Systems Security and Privacy. ICISSP 2018. Communications in Computer and Information Science, vol 977. Springer, Cham. https://doi.org/10.1007/978-3-030-25109-3_4
2) Simon S. Woo,Design and evaluation of 3D CAPTCHAs,Computers & Security,Volume 82,2019, Pages 49-67,ISSN 0167-4048,https://doi.org/10.1016/j.cose.2018.12.006.
3) Paliwal, S., Singh, R., & Mandoria, H.L. (2016). A Survey on Various Text Detection and Extraction Techniques from Videos and Images.
4) Vohra, U. S., Dwivedi, S. P., and Mandoria, H. L. (2016). An Analytical Study of Handwritten Character Recognition. *i-manager's Journal on Pattern Recognition*, 2(4), 26-41. https://doi.org/10.26634/jpr.2.4.5946
5) H. KardanMoghaddam, "Proposing an algorithm for converting published and handwritten texts to CAPTCHA by using image processing," *2016 Eighth International Conference on Information and Knowledge Technology (IKT)*, Hamedan, Iran, 2016, pp. 170-176, doi: 10.1109/IKT.2016.7777762.
6) Kabir, Farzana and Jasmeet Kaur. "Color Image Encryption for Secure Transfer over Internet: A survey." (2017).
7) Bose, J. Subash Chandra and Greeshma Gopinath. "A Survey Based on Image Encryption then Compression Techniques for Efficient Image Transmission." (2015).
8) M. Singh and A. Kaur, "An efficient hybrid scheme for key frame extraction and text localization in video," *2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Kochi, India, 2015, pp. 1250-1254, doi: 10.1109/ICACCI.2015.7275784.
9) Raja, Prof. S. and Dr. V. Mohan. "A REVIEW ON VARIOUS IMAGE ENCRYPTION TECHNIQUES FOR SECURE IMAGE TRANSMISSION." (2015).
10) Bansal, Chanchal et al. "Handwritten Numeral Recognition using SVM and Chain Code." (2014).
11) Hussain, Rafaqat et al. "Recognition of merged characters in text based CAPTCHAs." *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)* (2016): 3917-3921.
12) M. B. Rani, K. Rojarani, K. V. R. Rani, P. Boddepalli, P. V. Chintalapati, and P. K. Karri, "A deep learning approach with adaptive intelligence for coal classification," in Sustainable Materials, Structures and IoT, 1st ed., CRC Press, 2024, pp. 5. [Online]. Available: https://www.taylorfrancis.com/chapters/edit/10.1201/9781003596776-41/deep-learning-approach-adaptive-intelligence-coal-classification-beulah-rani-kandula-rojarani-kolluru-vindhya-rani-boddepalli-prameela-phaneendra-varma-chintalapati-praveen-kumar-karri.
13) Rao, M.C., Karri, P.K., Nageswara Rao, A., Suneetha, P. (2023). Automating Fish Detection and Species Classification in Underwaters Using Deep Learning Model. In: Kumar, A., Ghinea, G., Merugu, S. (eds) Proceedings of the 2nd International Conference on Cognitive and Intelligent Computing. ICCIC 2022. Cognitive Science and Technology. Springer, Singapore. https://doi.org/10.1007/978-981-99-2742-5_39
14) J. Sanyasamma, H. B. Gogineni, M. B. Rani, N. Akhila, P. K. Pinjala, and K. P. Kumar, "Precise monkeypox identification utilizing transfer learning via EfficientNetB3 and tailored Keras callbacks," Asian Food Journal of Biological Sciences, vol. 6, no. 6, pp. 2488-2494, 2024. [Online]. Available: https://doi.org/10.33472/AFJBS.6.6.2024.2488-2494.