Erratic Navigation in Lecture Videos using Hybrid Text based Index Point Generation

Geeta S Hukkeri[1], R. H. Goudar[2]
Research Scholar[1], Associate Professor[2]
Department of CSE, VTU
Belagavi, India

***Abstract*—The difficulty in lecture videos is an erratic navigation in lecture video for watching only the needed portion of video content. Machine learning technologies like Optical Character Recognition and Automatic Speech Recognition allows to easily fetch the information that is hybrid text from lecture slides and audio respectively. This paper presents three main analysis for hybrid text retrieval, which is further useful for indexing the video. The experimental results indicate that the key frame extraction accuracy is 94 percent. The accuracy of the Slide-To-Text conversion achieved by this study's evaluation of the text extraction capability of Tesseract, Abbyy Finereader, Transym, and the Google Cloud Vision Optical Character Recognition is 92.0%, 90.5%, 80.8%, and 96.7% respectively. Similarly the result of title identification is about 96 percent. To extract the speech text three different APIs are used namely, Microsoft, IBM, and Google Speech-to-Text API. The performance of the transcript generator is measured using Word Error Rate, Word Recognition Rate, and Sentence Error Rate metrics. This paper found that Google Cloud Vision Optical Character Recognition and Google Speech-to-Text API have achieved best results compared to other methods. The results obtained are very good and agreeable, therefore the proposed methods can be used for automating the lecture video indexing.**

Keywords*—*Automatic speech recognition; indexing; key-frames; lecture video; optical character recognition; title identification; text extraction

I. INTRODUCTION

The learning style of each individual learner has changed due to the lot of improvement in lecture videos as distance learning gives flexibility to access it independent of learner's time and place. Though the lecture recordings are convenient to learn from any place at any time there is a problem of watching only the needed topic from the long lecture recording. The focus of this paper is to generate the index points for non-linear navigation based on hybrid text. The processing of hybrid text extraction includes three analysis like "Visual screen analysis, Video OCR analysis, and Speech–to–text (STT) analysis."

Text in video pictures can be utilized as an indexing reason. Thus it is generally fair to initially identify elements from pictures. At the point when items have been effectively separated from their experiences, they likewise should be explicitly recognized. In this paper, a technique is presented that at the same time names contour and elements in binary images. There are numerous strategies that utilize certain contour highlights for ranking characters. The presented strategy marks every element utilizing a contour tracing method. A frame differencing method is used to obtain the key-frames [2]. Once the key-frames are retrieved, an OCR technique is used to extract the text from it. Current video OCR (Optical Character Recognition) methods depend on the mix of complex pre-handling methods for text extraction and conventional OCR engines. For video OCR, first video outlines must be recognized that acquire noticeable printed data; at that point, the content must be confined, the meddling background must be taken out, and mathematical changes must be applied before standard OCR engines can handle the content effectively and it is very powerful [3]. The general video OCR system comprises two fundamental advances: text detection and text recognition. Text detection measure decides the area of text inside the video picture. Microsoft Cognitive Services and Google Vision API [5] are some minimal expense answers that are presently available. The present status of technology says that recognition of object can be done using Convolutional Networks or Selective Search [7] Likewise, recognition of face is done using Fisherfaces or EigenFaces [8]. Google takes these methods and implement its AI cycles to further develop them. Google's Cloud Vision (GCV) is based on incredible PC vision models that power various Google administrations. Thus, a GCV OCR is applied to obtain sequence of strings from the key-frames.

Automatic Speech Recognition (ASR) [24] is being used in day-to-day applications. "The goal of speech recognition is to enable the humans and computers to have natural communication via speech". The accuracy of the model performance can be known with the results of transcription and segmentation obtained by the manual and automatic methods. The limitations of manual transcription such as costly, delayed performance, and error-prone when thousands of speech files are involved, lead to adaptation of automatic transcription; thus the study suggests to go for automated approach. The systems like automatic speech recognition (ASR) and text-to-speech (TTS) are performing in excess of 90% of accuracies. With AI ASR systems can result high-quality transcripts and with the usage of multi-modal data accuracy can be improved. Other than speech clarity there are many causes for the result of ASR system error rate [9] [10]. Now Google provides improved speech recognition with the usage of new technologies like "Voice Search on mobile, Voice Input, Goog411, Voice Actions, Voice Search on desktop, etc." The following are the objectives of this study:

- Retrieval of key-frames by analysing the visual screen.

- Video OCR analysis to identify title lines and extract text from lecture slides which is further helpful for creating index points.
- To extract the audio portion from the lecture videos to convert the instructor's spoken remarks into text for the purpose of creating index points.
- OCR and ASR performance is compared in order to determine which method is more effective.

II. Related Work

Effective searching and navigation of lecture video topics was especially intended for Slide Based Lecture Videos (SBLV) [11] that addresses a critical part of online talk recordings. A design for a successful Video Summarization [12] similarly as video motion rundown was proposed. A procedure for key edge removal was intended [13, 14] ward on the square based Histogram qualification and edge matching rate. Static video synopsis is perceived as a compelling style for users to rapidly peruse and understand enormous quantities of recordings [15]. Hence static video outline is considered as a clustering issue. Video skimming [16] normally viewed as a significant system for video summary. Generally, video text is installed in an exceptionally heterogeneous foundation with an incredible assortment of differences, which makes it hard to be perceived by standard OCR programming. GCV (Google cloud Vision) OCR is one of the popular API used in this case. GCV API enables the improvements of appliances that require AI help, notably for pictures comprehension [17, 18, 19]. A few examinations have successfully been made utilizing Cloud Vision API. Paper [20] efficiently executed GCV API for content-Based Image Retrieval. Mulfari carried out OCR capacity of GCV API to invent helpful innovation for humans with lack of ability, particularly for people who are evidently disabled or visually impaired. They removed text from pictures at that point express it through installed text-to-speech programming. From the study of previous works on GCV API we found that GCV API has been shown to offer one of trustworthy OCR appliance. Hence, its skill will also be examined in separating text from the talk video pictures.

For large vocabulary speech recognition, a DBN (Deep Boltzmann Machine) with a pre-trained ANN/HMM (Artificial Neural Network/Hidden Markov model) can be used. For recognizing disordered speech of the user a VIVOCA (Voice-Input Voice-Output Communication Aid) was evaluated [21], using this users can produce understandable speech from disordered speech. An Android-based application was developed for English learning using the Google Speech API, which has motivated authors to work on a speech recognition application in order to get text from the audio. The developers of the speech recognition system are expected to select Open API for the development of application speech recognition system [22]. As the study recommended to use automatic speech recognition, there is a demand for the efficiency and accuracy. Among most eminent Automatic Speech Recognition (ASR) systems, three are benchmarked on the bases of their performance namely the Google, wit, and IBM Watson [23], among these three the results of Google’s ASR is better [4] [23]. The comparative study between Google Speech with Pocketsphinx shows that the background noise filtering result of Google Speech is more impressive than Pocketsphinx. According to research [1],[6], we note that the Google’s Speech-to-Text outperformed other services and it has the less error rate in any case.

III. Hybrid Text Extraction from Lecture Video Images

The framework for hybrid text extraction has been shown in the Fig. 1, which includes “Visual screen analysis, Video OCR analysis, and Speech–to–text (STT) analysis.” The two main parts in Lecture Video (LV) are slides (visual content) and audio (explanation of slides given by the instructor).

With the visual screen analysis we have segmented input lecture video to extract the slides from video and applied frame-differencing method to obtained key-frames. With the Video OCR analysis we have performed text detection and recognition using OCR to extract the texts from the slides. Then identified the title line from the text bounded images using geometrical information. With Speech–to–text (STT) analysis, extracting text from the audio track of the LV using ASR technology. Text from speech is one of the principle wellsprings of data in a talk video. The teacher gives detailed data about the point in the video address. The speech text is generous and unconstrained. The speech is one of the significant variables in content-based recovery of a point in a long video address.

A. Visual Screen Analysis

The main aim of this analysis is to extract key-frames from the LV. Extraction of key-frames for different sorts of video should be possible by consolidating distinctive sort of video content. Keypoint-based framework was intended to address the keyframe assurance issue with the objective that close by features can be used in picking keyframes. Usually, picked keyframes ought to be both descriptive of video content and containing least abundance. At first the long lecture video is divided into number of segments. Normally, a video holds 24 frames in a second, among them, most of the frames are repetitive. Thus, it is necessary to extract only useful frames. Generally, a video of M minutes is divided into:

$$F = M*60*30\text{frames} \quad (1)$$

where F is variety of frames created from the video at the start.

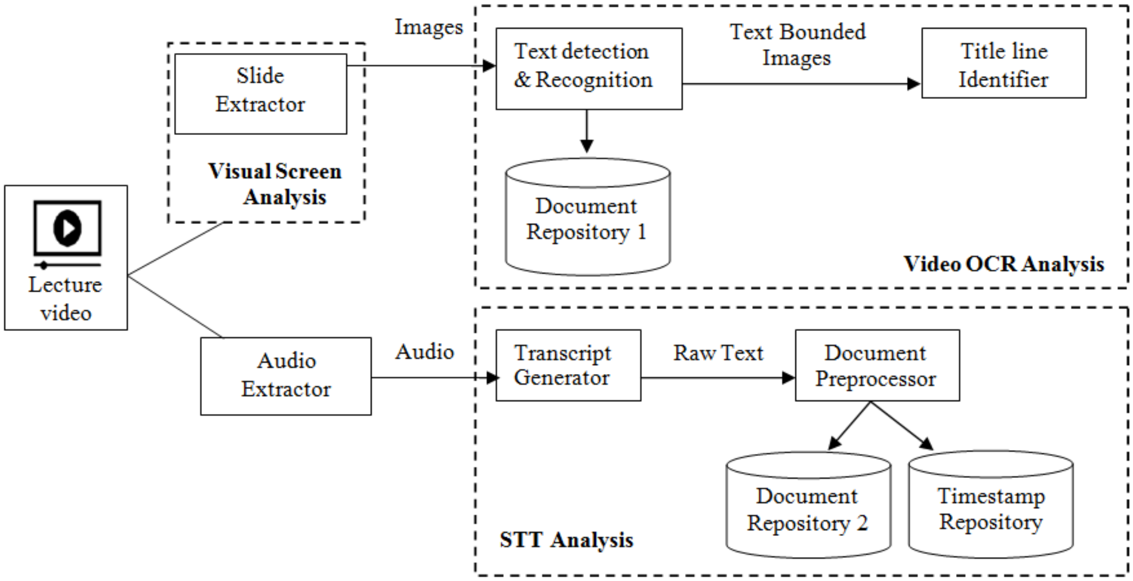


Fig. 1. Framework for Hybrid Text Extraction from Lecture Video.

In a lecture video, a topic will be deliberated for at least 10 seconds. Thus, to reduce the repetitive frames, 10 seconds delay is made in frame creation. The difference between adjacent frames is obtained to get the key-frames. The flow diagram of this procedure is shown in the Fig. 2.

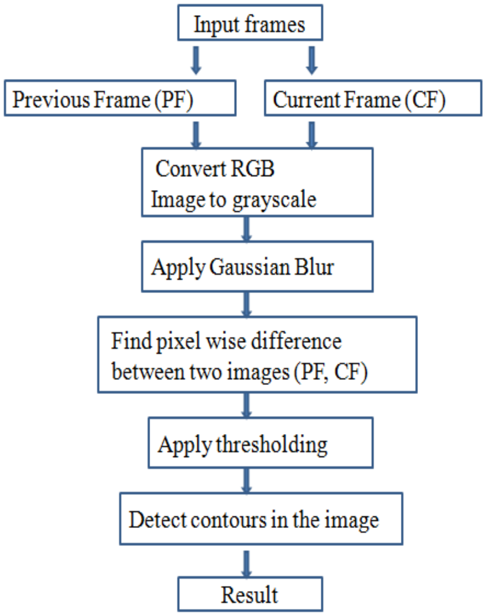

Input frames
Previous Frame (PF)
Current Frame (CF)
Convert RGB Image to grayscale
Apply Gaussian Blur
Find pixel wise difference between two images (PF, CF)
Apply thresholding
Detect contours in the image
Result

Fig. 2. Flow of Key-Frame Selection.

The method we developed for obtaining key-frames works in three stages. Initial stage is to get the difference between adjacent frames using frame-differencing method.

To compute this, frames are removed at 1 Hz from the informant video. Then compute the pixels whose difference beats the threshold value of 24. The bounding box is found for all such pixels whose change is higher than 1% of the total and analyse its size and overlay at the center of frame. A new segment is detected when the bounding box is moreover overlapping the center or is at least a third of the frame. After a time when the inter-frame difference gets steady for somewhat three seconds, a fresh key-frame is obtained as the final frame in the segment. Next, we test frames to build training sets for representing the talker/background, and slide images. Histograms are removed from the tested frames and train a SVM to discriminate slide and non-slide frames. Sample result of key-frames obtained from lecture video has been shown in the Fig. 3.

B. Video OCR Analysis

This section discusses two tasks.

1) Text detection and recognition: Text extraction process includes two subprocesses namely text detection and recognition which can be performed automatically by applying GCV OCR. GCV OCR is a part of GCV API. The GCV API permits developers to know the subject of a picture by enclosing wonderful AI designs in a user-friendly REST API. The Cloud Vision API rapidly groups pictures into many classes and peruses printed words contained inside pictures; it also recognizes singular items inside pictures. The Google permits the API to handle singular bits of a picture independently and return the outcome rapidly in brought together organizations.

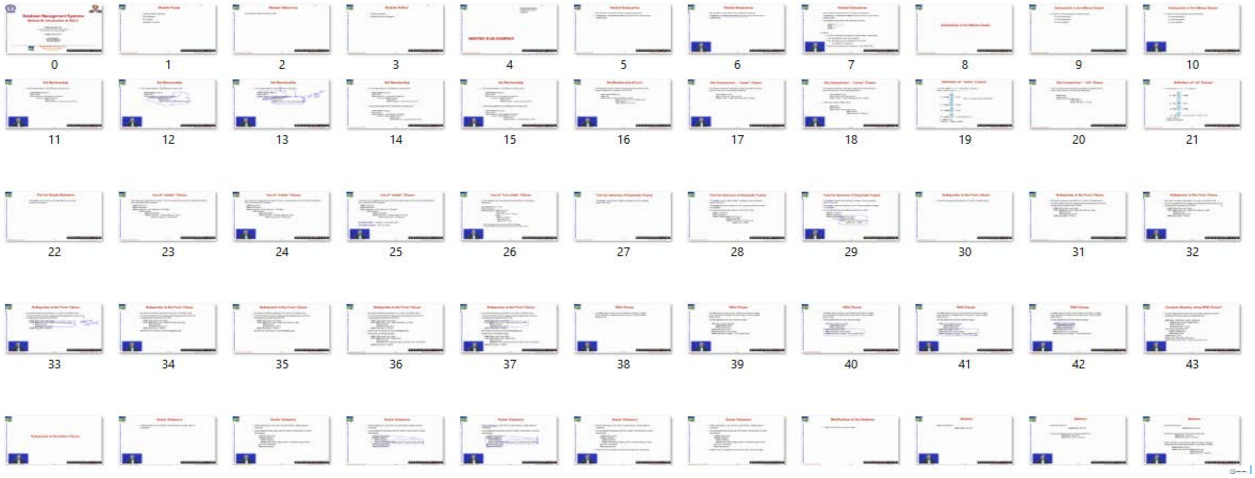


Fig. 3. Sample Result of Key-Frames.

One more asset of the GCV API is when doing a request for processing a picture; Google provides the power to imply the types of evaluation that must be on this picture. For instance, object detection, facial location, milestone recognition, and a lot more examination perform on the picture. The workflow of the Fig. 4 has been implemented using a python script.

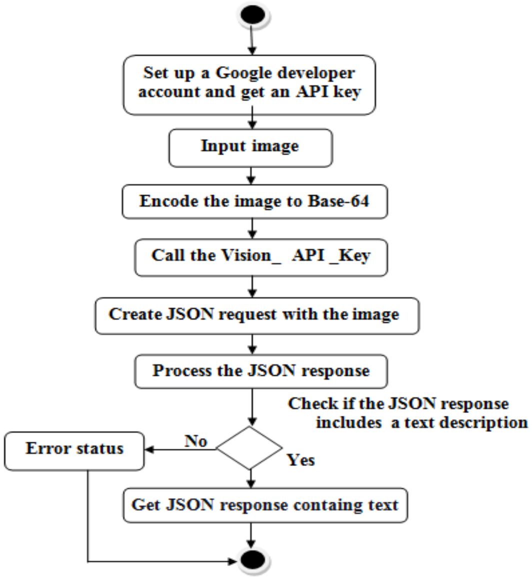

Set up a Google developer account and get an API key
Input image
Encode the image to Base-64
Call the Vision_ API _Key
Create JSON request with the image
Process the JSON response
Check if the JSON response includes a text description
No
Yes
Error status
Get JSON response containg text

Fig. 4. Workflow of Image Text Extraction Process.

Google OCR has different advantages, here we depict the hugest advantages:

- Robust - The two capacities, serving two kinds of text records subject to the clients' choice, make the Google Vision OCR similarly more robust than single-model OCR tools.
- Language support - Google has exhorted that its OCR is appropriate to in excess of 60 languages.
- Ease of utilization - The actual model is important for the in-constructed Google
- Vision library. The function-calling technique can be utilized in various languages in an extremely clear way.
- Scalability - Google's evaluating technique promotes clients to increase the use of the API, as more use prompts a less expensive normal cost.
- Speed - Google Cloud's warehouse stage superbly goes with the API utilization. By transferring the pictures into the drive, the response or reaction time of API can be extremely quick and versatile.

a) API call: Indicate the URL to the API and include the JSON data to POST to it. We first need to set up a Google developer account and obtain an API key [7] to perform OCR using Google Cloud Vision API.

b) Request: Send a JSON request containing a base64 encoded image file. The vision API performs feature detection on an image file.

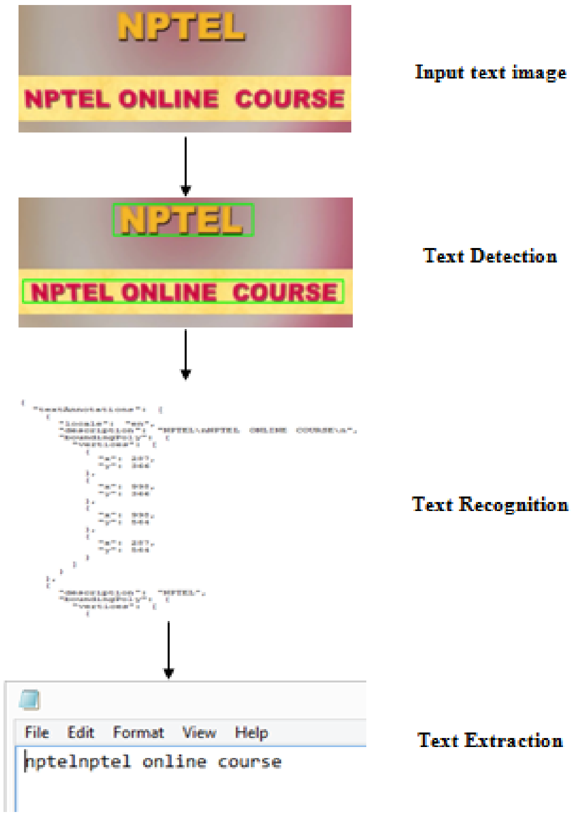
c) Response: We get a response in JSON format which includes text and bounding box containing location coordinates. Sample results of text extraction using Google Cloud Vision API is shown in Fig. 5.

2) *Title Identification:* For the most part, in the talk slides, the substance of title, caption, and main points has more importance than the typical slide text, as they sum up each slide. The design of text lines can mirror their diverse importance. This data is important for a talk video indexing. To recognize the potential title text lines, we apply the accompanying conditions.

a) The height of the title text line is more prominent than or equivalent to the normal height of text lines.

b) Title text line has, at any rate, three characters.

c) Horizontal start position of the text line ought to be, not exactly 50% of the frame width.



```
nptelnptel online course
```

Fig. 5. Sample Result of Text Extraction from an Image.

C. Speech-to-Text (STT) Analysis

Text from speech is one of the principle wellsprings of data in a talk video. The teacher gives detailed data about the point in the video address. The speech text is generous and unconstrained. The speech is one of the significant variables in content-based recovery of a point in a long video address. Utilizing Google Speech-to-Text Programming interface as a speech recognition instrument in our trial, the speech records of talk recordings are used for indexing purposes. The speech text may differ marginally; the educator might talk some irregular substance. In any case, accepted that speech includes significant theme data and can be utilized to accomplish topic division and index point creation. Speech-to-text APIs provide lot of pros like boosting productivity and efficiency, saves time, Reliability, helps physically disabled people, etc. This API is used in several applications like Chatbots, Automated dictation, Smart assistant, Voice commanding, Transcriptions for call centers, mixed language detection, etc. Among the popular APIs like Microsoft Google Speech-To-Text, Cognitive Services, Dialogflow, IBM Watson, etc., for speech-to-text processing the Google Speech-To-Text API gives more accurate results.

3) Google Cloud Speech-to-Text API: The Google Cloud Speech API is integrated with Google Translate API and Cloud Vision API. Machine Learning is essential for the Google Cloud Stage in the development of applications that can listen, view, and comprehend its general surroundings. With this complete Google Cloud Speech API developers can easily translate an audio into text by using neural network models. This API supports 110 plus languages and variations, to help a worldwide user base. The Google Speech Programming interface, otherwise called Speech-to-Text (STT), is a modern instrument that uses Google's AI innovation to change voice over to message. Google Speech Programming interface is one of the most amazing speech recognition service [1]. This API is an automated speech recognition (ASR) API adapted with deep neural networks. It can likewise deal with noisy sound in a variety of conditions. This API result include not only text but also the timestamp corresponding to each word. The flowchart of Talk-To-Text (TTT) processing is shown in the Fig. 6.

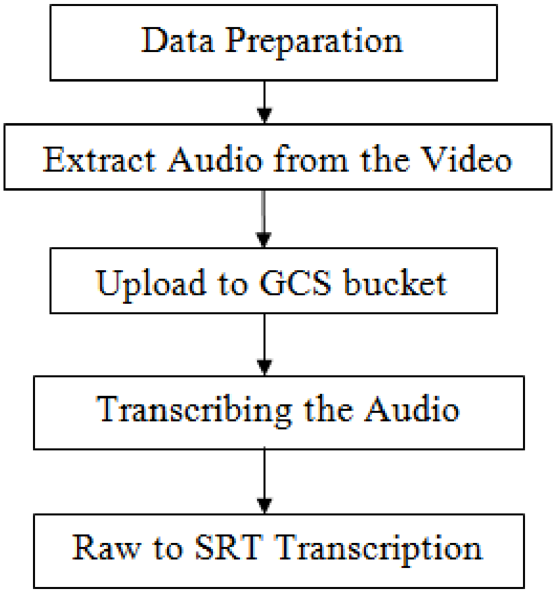


Fig. 6. Flowchart of TTT Processing.

a) Data preparation: Download lecture videos from different online courses like YouTube, NPTEL.

b) Extract audio from the video: Build the instance using Google speech-to-text API for Talk-To-Text (TTT) processing. Before doing anything, we have to install Ffmpeg to extract the audio from the lecture video. Here we are converting mp4 video file to ogg audio file. We have specified codec Opus in VoIP because of its audio compression with more quality and less delay rates. The sampling rate is set to 16000 hertz.

c) Upload to GCS bucket: We know that usually lecture video duration is longer than 60 seconds. Thus, we are requesting asynchronous speech recognition and we must store the audio file longer than 60 seconds in Google Cloud Storage bucket.

d) Transcribing the audio: The processing of audio file to obtain the transcription has been shown in the Fig. 7.

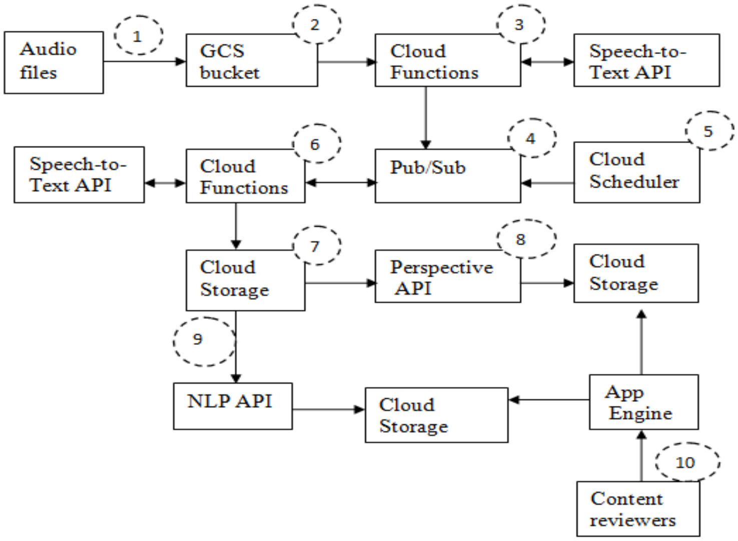
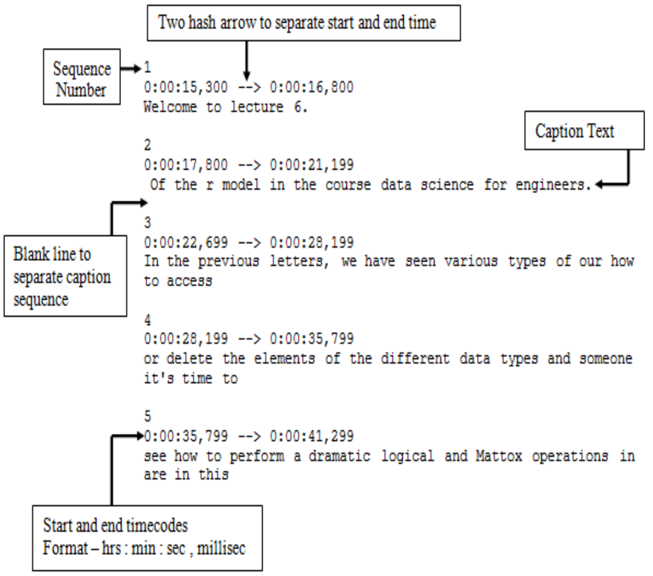


Fig. 7. Transcript Generation Process.

- Store audio file. The audio file is then stored in a Cloud Storage bucket. Before the audio file go through the remaining steps this step functions as a production bucket to maintain the files.
- Activate Cloud Function. When audio file meets the production bucket, a notification is generated. This notification triggers a Cloud Function to invoke a Speech-to-Text API.
- Invoke the Speech-to-Text API. Speech-to-Text API is invoked by Cloud Function to get a transcription of the audio file. This process is nonparallel, so a job ID is sent to the Cloud Function by Speech-to-Text API.
- Report Speech-to-Text job IDs. The audio filename and job IDs are then reported to the Pub/Sub point.
- Speech-to-Text voting. For every 10 minutes the Cloud programmer reports an announcement to a Pub/Sub point, which activates a next Cloud Function.
- Get Speech-to-Text API results. This Cloud Function extracts all announcements from the initial Pub/Sub point and pulls the filename and job IDs for every news. Each individual job status is checked by calling the Speech-to-Text API.
- In case when a job is over, the resulted transcription are registered to a next Cloud Storage bucket. Then Cloud Function moves the audio file to Cloud Storage bucket from the production Cloud Storage bucket. In case when a job is not over, a Pub/Sub announcement is added again to the Pub/Sub point. If there is no result from Speech-to-Text, the audio file is passed to a Cloud Storage error bucket. The obtained result (transcript of the audio file) is reported to a Cloud Storage bucket.
- Call Perspective API. The chance of "corruption" in the transcription is checked by calling Perspective API. Obtained results on this analysis is reported to another Cloud Storage bucket.
- Call the Cloud Natural Language API. The overall notion of the transcription is checked by calling the Natural Language API (called by fourth Cloud Function). The Cloud Function then reports the obtained results to another Cloud Storage bucket.
- Content reviewers. In the above diagram the App Engine enables user to check the outputs.


Two hash arrow to separate start and end time
Sequence Number
Caption Text
Blank line to separate caption sequence
Start and end timecodes Format – hrs : min : sec , millisec

```
1
0:00:15,300 --> 0:00:16,800
Welcome to lecture 6.

2
0:00:17,800 --> 0:00:21,199
 Of the r model in the course data science for engineers.

3
0:00:22,699 --> 0:00:28,199
In the previous letters, we have seen various types of our how
to access

4
0:00:28,199 --> 0:00:35,799
or delete the elements of the different data types and someone
it's time to

5
0:00:35,799 --> 0:00:41,299
see how to perform a dramatic logical and Mattox operations in
are in this
```

Fig. 8. SRT Formatted Text.

e) Raw to SRT Transcription: Raw text is the text obtained or collected from transcript generator before any manipulation. Thus, this text data is being send to the document pre-processor for analysing the raw text and produce resultant speech text with corresponding timestamps.

The raw text file is been formatted to SRT (“SubRip Subtitle”) formatted text file because each sequence of text in .srt file has five important parts shown in the Fig. 8.

IV. Result and Discussion

The implementation is done in an Intel Core 8 CPU @ 5.0 GHz, with ubuntu operating system.

A. *Keyframe Extraction*

To evaluate the performance of video keyframe extraction, we randomly chose seven lectures videos like data science (DS), cryptography(crypt), cloud computing (CC), computer networks (CN), DBMS, algorithms (Alg), and machine learning (ML) from different online courses with varying layouts, font size, and styles.

TABLE I. Key-Frames

Lecture Video	Duration in minutes	Keyframes	Total slides
DS	22	38	49
ML	60	39	43
CN	23	38	42
crypt	100	75	77
DBMS	33	72	75
CC	45	51	57
Alg	109	125	147

The number of desired slides in the lecture videos are manually annotated for ground truth. Then, we applied the slide extraction algorithm to these videos. We compare the results of extracted slides with ground truth using recall and precision. The precision and recall esteem recognize bogus alert rate and missed location outline rate individually. The estimation of precision diminishes if there is over-segmentation, i.e., superfluous frames are separated. The assessment of Recall lowers if there is under- segmentation, i.e., an ideal frame stays undetected. The Table I shows the result of obtained keyframes.

The F1 score is measured as:

$$\text{F1 score} = \frac{(2 * \text{Precision} * \text{Recall})}{(\text{precision}+\text{Recall})} \qquad (2)$$

Where,

$$\text{Precision} = \frac{\#\text{slides detected correctly}}{\#\text{slides detected}} \qquad (3)$$

$$\text{Recall} = \frac{\#\text{slides detected correctly}}{\#\text{ground truth slides}} \qquad (4)$$

Results of Precision, Recall, and F1 score obtained fromthe above formulas is 0.9, 0.98, and 0.94 respectively.

B. *Slide-to-Text Conversion*

With an average accuracy of 96.7 percent (Fig. 9), the text extraction results from each lecture video using the obtained key-frames showed that GCV outperformed other OCR APIs in this task. The findings are displayed in Table II. Transym's is 80.8 percent, Tesseract's is 92 percent, and Abbyy Finereader's is 90.5 percent. When the file size and resolution are taken into account, the accuracy of the GCV OCR is significantly higher than that of other methods. Additionally, low-resolution or small-size photos have the lowest accuracy. Performance is assessed using the three factors stated below.

$$Recall = \frac{Total\ Extracted\ text}{Total\ key-frames} \qquad (5)$$

$$Precision = \frac{Correctly\ extracted\ text}{Total\ Extracted\ text} \qquad (6)$$

$$F1 - score = equation\ (2)$$

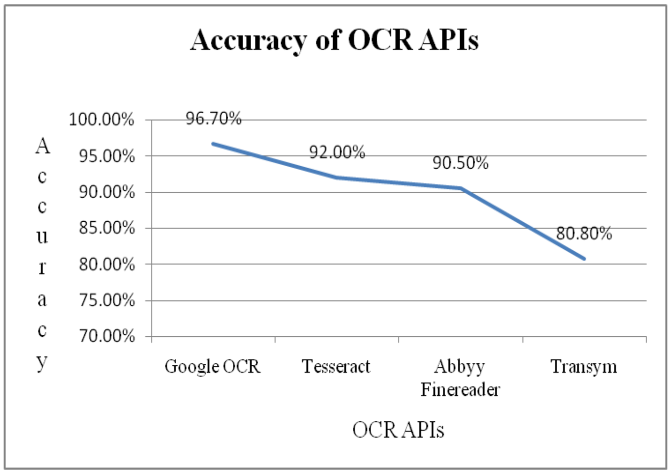


Fig. 9. Performance Comparison of Different OCR APIs.

C. *Title Identification*

Segmentation is done on lecture videos and taken 98 lecture slides to evaluate title identification. Then, the geometrical information of the text lines are used to identify the title line. The accuracy (Acc) of the title identification method is measured using the formula given below:

$$\text{Acc }\% = 1 - \left(\frac{\#\text{errors}}{\#\text{slides with title}}\right) * 100 \qquad (7)$$

As a result, we obtained that the title line in 94 slides was identified correctly among 98 slides. The accuracy gained is 96%.

D. *Transcript Generation*

Three different speech to text APIs are used to perform this. The performance of each method is evaluated using the below three metrics

1) Word Error Rate (WER): It is used to test the occurrence of word errors in the obtained transcript. Levenshtein Distance is applied to find the difference between two word placements. As the word placement can have varying length, there can be substitutions (S), deletions (D), and insertions (I) to alter one word into the other. The WER can be calculated using the below equation (8).

$$\text{WER} = ((\text{S}+\text{D}+\text{I})/\text{N})*100 \qquad (8)$$

TABLE II. RESULTS OF DIFFERENT OCR APIS

Method	**Pr (%)**	**Re (%)**	**F1-Score (%)**
Google OCR	97.2	94.7	**97.4**
Tesseract	88.2	89.4	88.7
Abbyy Finereader	87.8	86.8	87.2
Transym	65.6	84.2	73.7
Google OCR	94.7	97.4	**96.0**
Tesseract	86.1	92.3	89.0
Abbyy Finereader	91.4	89.7	90.5
Transym	72.7	84.6	78.1
Google OCR	97.2	97.3	**97.2**
Tesseract	88.8	94.7	91.6
Abbyy Finereader	88.2	89.4	88.7
Transym	62.5	84.2	71.7
Google OCR	97.2	97.3	**97.2**
Tesseract	91.5	94.6	93.0
Abbyy Finereader	91.3	92.0	91.6
Transym	83.3	88.0	85.5
Google OCR	98.5	98.6	**98.5**
Tesseract	92.6	94.4	93.4
Abbyy Finereader	95.5	93.0	94.2
Transym	86.1	90.2	88.1
Google OCR	98.3	96.0	**97.1**
Tesseract	93.4	90.1	91.7
Abbyy Finereader	88.6	86.2	87.3
Transym	75.6	80.3	77.8
Google OCR	89.4	98.4	**93.6**
Tesseract	96.6	96.8	96.6
Abbyy Finereader	94.1	95.2	94.6
Transym	88.8	93.6	91.1

2) *Word Recognition Rate (WRR):* Below equation (9) is used to calculate the WRR.

$$WRR = ((N-D-S) / N)*100 \quad (9)$$

3) *Sentence Error Rate (SER):* It is used to find the occurrence of errors in the sentences of the transcript. If there is a word by word match between manual transcription and recognition output, then it is taken into account as exact match. The SER accuracy is calculated using the below formula.

$$SER = \frac{\text{No.of inaccurate sentences}}{\text{Total No.of sentences}} * 100 \quad (10)$$

The results of WER, WRR, and SER obtained by three Speech-to-Text APIs is shown in Table III and it clearly shows that results of Google Speech-to-Text API is much more better than other two methods. IS stands for incorrect sentences. The comparison of three APIs (IBM, Microsoft, Google) in terms of WER, WRR, and SER is shown in Fig. 10.

TABLE III. RESULTS OF DIFFERENT SPEECH-TO-TEXT APIS

Google Cloud Vision OCR					
Words	***S***	***I***	***D***	***Sentences***	***IS***
2698	149	72	112	213	112
6904	349	123	294	548	288
3044	184	98	136	268	149
13922	773	388	528	1020	567
4620	238	118	178	358	192
7435	364	142	346	596	302
14222	788	391	593	1131	583
IBM Watson					
2530	189	122	142	205	133
6733	519	374	394	538	323
2874	272	128	186	259	186
13747	1298	949	1068	1009	589
4451	393	258	298	351	194
7267	614	392	436	587	349
14047	1328	992	1198	1119	595
Microsoft					
2488	197	184	188	198	132
6689	651	458	583	530	342
2826	255	203	218	248	181
13693	1543	1056	1284	1001	591
4404	468	282	346	341	198
7219	758	487	592	574	375
13997	1846	1083	1423	1113	602

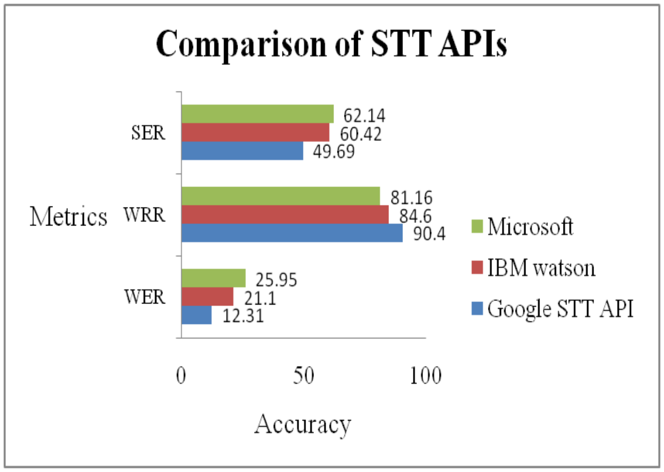


Fig. 10. Comparative Results of Three Different APIs.

V. CONCLUSION

This paper presents a whole work stream for keyframe extraction, hybrid text extraction and title identification proof. Frame differencing method is applied to accomplish superior key-frame extraction and achieved 94% of F1 score. To extract the text from key-frames four OCR methods have been proposed and found that Google cloud vision OCR is best achieving upto 97% accuracy. Google permits the API to handle singular bits of a picture independently and return the outcome rapidly in brought together configuration. The title lines are identified using the geometrical information of the text lines and gained 96% accuracy. To extract the speech text three different APIs are used namely, Microsoft, IBM, and Google. The WER, WRR and SER are computed to measure the accuracy of these model and the achieved result of these parameters is shown in this paper. This paper founds that Google speech to text API has achieved best result in terms of WER, SER, and WRR compare to other two APIS. The outcomes acquired are really exact and very helpful.

The future work includes the implementation of an indexing algorithm for lecture videos based on obtained hybrid text.

ACKNOWLEDGMENT

We are very thankful to our parents, family, and friends for supporting to complete this work.

REFERENCES

[1] Foteini Filippidou and Lefteris Moussiades, "A Benchmarking of IBM, Google and Wit Automatic Speech Recognition Systems," IFIP International Federation for Information Processing 2020 Published by Springer Nature Switzerland AG 2020. pp. 73–82, 2020. https://doi.org/10.1007/978-3-030-49161-1_7

[2] Yin J, Liu L, Liu Q. The infrared moving object detection and security detection related algorithms based on W4 and frame difference[J].Infrared Physics & Technology, 2016 (7), pp. 302 - 315.

[3] Akshay Parwar, Akansha Goverdhan, Apurva Gajbhiye, Prajkta Deshbhratar, Roshan Zamare, Prasanna Lohe, "Implementation to Extract Text from Different Images by Using Tesseract Algorithm," International Journal Of Engineering And Computer Science ISSN: 2319-7242 Volume 6 Issue 2, Feb. 2017, pp. 20298-20300.

[4] Joshua Y. Kim1, Chunfeng Liu, Rafael A. Calvo, Kathryn McCabe, Silas C. R. Taylor, Björn W. Schuller, Kaihang Wu,. "A Comparison of Online Automatic Speech Recognition Systems and the Nonverbal Responses to Unintelligible Speech", 2019. https://doi.org/10.48550/arXiv.1904.12403

[5] Nurzam, F.D., Luthfi, E.T. "Implementation of real-time scanner java language text with mobile vision android based." In: 2018 International Conference on Information and Communications Technology (ICOIACT). IEEE; 2018, pp. 724–729.

[6] Veton Këpuska, Gamal Bohouta, "Comparing Speech Recognition Systems (Microsoft API, Google API And CMU Sphinx)", Int. Journal of Engineering Research and Application, ISSN : 2248-9622, Vol. 7, Issue 3, (Part -2) March 2017, pp.20-24.

[7] Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Region-based convolutional networks for accurate object detection and segmentation," Pattern Analysis and Machine Intelligence, IEEE Transactions on, Vol. 38, no. 1, 2016, pp. 142–158.

[8] Shih-Hsin Chen, Yi-Hui Chen, "A New Content-Based Image Retrieval Method Based on the Google Cloud Vision API," ACIIDS 2017: Intelligent Information and Database Systems, 2017, pp 651-662.

[9] M. S. Barnish, D. Whibley, S. Horton, Z. R. Butterfint, and K. H. O. Deane, "Roles of cognitive status and intelligibility in everyday communication in people with Parkinsons disease: A systematic review," J. Parkinsons. Dis., vol. 6, no. 3, pp. 453–462, 2016.

[10] A. Behrman, "A clear speech approach to accent management," Am. J. speech-language Pathol., vol. 26, no. 4, pp. 1178–1192, 2017.

[11] Zhao, Baoquan, Songhua Xu, Shujin Lin, Ruomei Wang, and Xiaonan Luo. "A New Visual Interface for Searching and Navigating Slide-Based Lecture Videos." In 2019 IEEE International Conference on Multimedia and Expo, 2019, pp. 928-933,

[12] Huang, Cheng, and Hongmei Wang. "Novel key-frames selection framework for comprehensive video summarization," IEEE Transactions on Circuits and Systems for Video Technology (2019).

[13] A. M. Reddy, V. V. Krishna, L. Sumalatha and S. K. Niranjan, "Facial recognition based on straight angle fuzzy texture unit matrix," 2017 International Conference on Big Data Analytics and Computational Intelligence (ICBDAC), Chirala, 2017, pp. 366- 372.

[14] Purushotham Reddy, M., Srinivasa Reddy, K., Lakshmi, L., Mallikarjuna Reddy, A. "Effective technique based on intensity huge saturation and standard variation for image fusion of satellite images" International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249-8958, Volume-8 Issue-5, June 2019.

[15] Wu, Jiaxin, Sheng-hua Zhong, Jianmin Jiang, and Yunyun Yang. "A novel clustering method for static video summarization." Multimedia Tools and Applications 76, no. 7 (2017), pp. 9625-9641.

[16] Zhang, Lanshan, Linhui Sun, Wendong Wang, and Ye Tian. "KaaS: A standard framework proposal on video skimming." IEEE Internet Computing 20, no. 4 (2016), pp. 54-59.

[17] Irmanti, D.,Hidayat, M.R., Amalina, N.V., Suryani, D., et al. "Mobile smart travelling application for indonesia tourism." Procedia computer science 2017, pp. 556–563.

[18] Chen, S.H., Chen, Y.H. "A content-based image retrieval method based on the google cloud vision API and wordnet." In: Asian Conference on Intelligent Information and Database Systems. Springer; 2017, pp. 651–662.

[19] Mulfari, D., Celesti, A., Fazio, M., Villari, M., Puliafito, A.. "Using google cloud vision in assistive technology scenarios." In: 2016 IEEE Symposium on Computers and Communication (ISCC). IEEE; 2016, pp. 214–219.

[20] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, 2015, pp. 436–444.

[21] D Intan, S Saputra, SW Handani, GA Diniary. Utilization of Cloud Speech API for the Development of English Language Learning Media using Speech Recognition Technology (in Indonesia Pemanfaatan Cloud Speech API untuk Pengembangan Media Pembelajaran Bahasa Inggris Menggunakan Teknologi Speech Recognition). TELEMATIKA. 2017; 10(2): 92–105.

[22] Hyun Jae Yoo, Sungwoong Seo, Sun Woo Im, and Gwang Yong Gim, "The Performance Evaluation of Continuous Speech Recognition Based on Korean Phonological Rules of Cloud-Based Speech Recognition Open API", International Journal of Networked and Distributed Computing Vol. 9(1); January (2021), pp. 10–18

[23] Foteini Filippidou and Lefteris Moussiades, "A Benchmarking of IBM, Google and Wit Automatic Speech Recognition Systems", International Federation for Information Processing (IFIP), pp. 73–82, 2020.

[24] H. Roh and K. Lee, "A Basic Performance Evaluation of the Speech Recognition API ofStandard Language and Dialect using Google, Naver, and DaumKAKAO APIs", Asia-pacific Journal of Multimedia Services Convergent with Art, Humanities, and Sociology, Vol.7, No.12, pp. 819-829, December 2017.