

Concept Paper

Not peer-reviewed version

Efficient Indexing of Meta-Data (Extracted From Educational Videos)

[Shalika Kumbham](#)*, Abhijit Debnath, Sreenivasa Rao Krothapalli

Posted Date: 6 December 2023

doi: 10.20944/preprints202312.0116.v2

Keywords: image processing; keyframes; indexing; metadata



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

Efficient Indexing of Meta-Data (Extracted from Educational Videos)

Shalika Kumbham *, Abhijit Debnath and Krothapalli Sreenivasa Rao

Indian Institute of Technology, Kharagpur; abhijitdebnath@iitkgp.ac.in (A.D.); ksrao@cse.iitkgp.ac.in (K.S.R.)

* Correspondence: shalika@iitkgp.ac.in

Abstract: Video lectures are becoming more popular and in demand as online classroom teaching is becoming more prevalent. Massive Open Online Courses (MOOCs), such as NPTEL, have been creating high-quality educational content that is freely accessible to students online. A large number of colleges across the country are now using NPTEL videos in their classrooms. So more video lectures are being recorded, maintained, and uploaded. These videos generally contain information about that video before the lecture begins. We generally observe that these educational videos have metadata containing five to six attributes: Institute Name, Publisher Name, Department Name, Professor Name, Subject Name, and Topic Name. It would be easy to maintain these videos if we could organize them according to their categories. The indexing of these videos based on this information is beneficial for students all around the world to efficiently utilise these videos. In this project, we are trying to get the metadata information mentioned above from the video lectures.

Keywords: image processing; keyframes; indexing; metadata

1. Introduction

Students are going through numerous changes as they adjust to the COVID situation. The mode of classes is one of the most affected. A video is a digitally recorded and saved programme, movie, or other visual media product that contains moving visuals with or without audio. The frame is a single image in a sequence of pictures. In most cases, one second of video is made up of 24 or 30 frames per second (FPS). The frame is made up of the image and the time the image was exposed to the view when it was taken. The animation is created by extracting a series of frames in a row. Creating and managing web videos is a difficult task. It would be a lot easier if we could automatically categorise them without much human interaction. It saves a significant amount of time and work. As a result, we're attempting to create an interface for obtaining the necessary data and organising it into the appropriate categories. If you give it a video, it will provide you with information from those categories. For example, if there is an NPTEL video on Image Processing taught by Prof. K S Rao from IIT Kharagpur's Computer Science Department, we must map the information to the appropriate column, such as Publisher Name - NPTEL, Professor Name - K S Rao, Department Name - Computer Science, and so on.

2. Literature Review

Yan et al. [1] talked about key frame extraction from videos (human action videos) using a deep learning approach. An algorithm has been proposed that is mostly focused on the fusion of deep features and histograms to reduce the number of redundant frames.

Because of the combined elements of human position in movement and the background, detecting representative frames in movies based on human actions is fairly difficult. Gawande et al. [2] address the issue and define key frame detection as the process of identifying the video frames that contribute the most to distinguishing the underlying action category from all others. They tested their method on the UCF101 dataset, which is a difficult human action dataset, and found that it can detect critical frames with high accuracy.

Hamad and Kaya [3] discusses the need to index the images in a computer system for easy access. They mention the use of OCR to achieve the same. They went over the pre-processing, segmentation, normalisation, feature extraction, classification, and post-processing aspects of an OCR system and the challenges involved

Ding et al. [4] worked on improving the recognition ability of text in a video based on OCR. Machine Learning techniques were used to improve the accuracy of the recognition with suitable preprocessing.

Yang et al. [5] discussed the need for video indexing and offered an approach for automatic lecture video indexing based on video OCR technology. They developed a new technique for analysing and extracting slide structure based on the geometrical information of recognised text lines.

Yang et al. [6] implemented a new video segmenter for automatic slide video structure analysis and a word detector based on weighted DCT (discrete cosines transformation).

The TIB AV-Portal (AV-Portal [7]) is a portal for scientific videos with a focus on technology/engineering as well as architecture, chemistry, computer science, mathematics, and physics. Other scientific disciplines are also included in the AV-Portal. The videos include computer visualizations, learning material, simulations, experiments, interviews, video abstracts, lecture, and conference recordings, and (open) audiovisual learning and teaching materials. The portal uses various automatic video analyses. The added value: these video analyses enable pinpoint searches within the video content.

NDLI (NDLI [8]) is an integration platform that sources metadata from a massive number of sources to provide single window access to a wide variety of digital learning resources. Metadata summarizes basic information of a resource, which can make finding and working with particular resources easier. Sources have different standards for their metadata with a variety of resources. NDLI metadata standard has been developed to draw up this uniform schema. It is an envelope of several well-established global standards.

The Dublin Core Metadata Initiative (DCMI [9]), which formulates the Dublin Core, is a project of the Association for Information Science and Technology (ASIS&T). "Dublin Core" is also used as an adjective for Dublin Core metadata. It is a style of metadata that draws on multiple Resource Description Framework (RDF) vocabularies, packaged and constrained in Dublin Core application profiles. The Dublin Core Metadata Initiative (DCMI) provides an open forum for developing interoperable online metadata standards for a broad range of purposes and business models. DCMI's activities include

- consensus-driven working groups,
- global conferences and workshops,
- standards liaison, and
- educational efforts to promote widespread acceptance of metadata standards and practices.

2.1. Existing Methods

A similar type of problem is solved in a research article by Haojin Yang, Maria Siebert, Patrick Lühne, Harald Sack, and Christoph Meinel. It employs the keyframe detection technique, text localization, text reading, and text classification into title, subtitle/key point, normal content, and footnote based on the stroke width and height of the text identified. This article is the most closely related to the issue statement we're working on. We discovered few to no works that matched our problem description.

3. Our Methodology

3.1. Objective

To develop a dataset from NPTEL lecture videos and build a better model to categorize the videos based on variables such as Institute Name, Publisher Name, Professor Name, and Department Name. An example slide containing the four attributes is shown in Figure 1

3.2. Dataset Preparation

We presume that the features we require are present before the introductory section, i.e., before the professor begins teaching in the lecture video. As a result, we limited the length of the videos to the introduction part. For better outcomes, we meticulously collected the videos to ensure that they were not concurrent with one another. Furthermore, the occurrences to be categorized are entirely independent of one another.

The video lecture can be categorized into 2 types:

- All attributes in same slide (Figure 1)

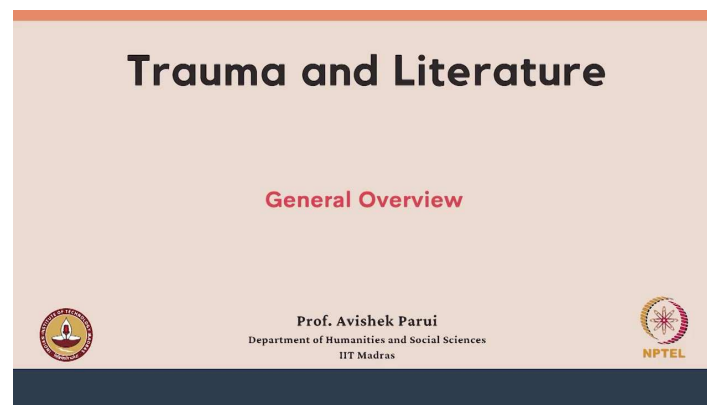


Figure 1. All the attributes in a frame.

- Attributes in different slides (Figure 2)

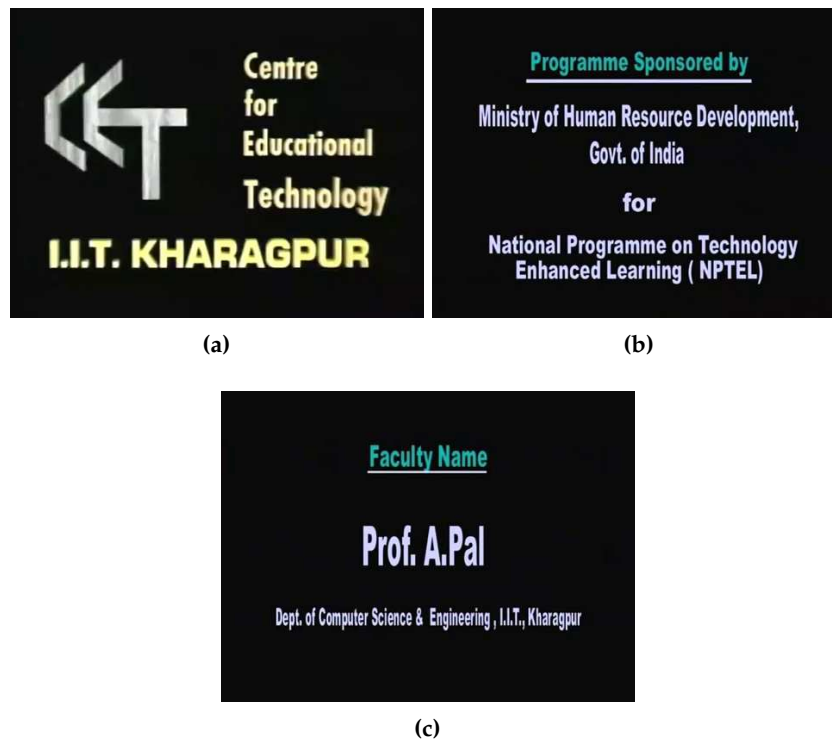


Figure 2. (a) Frame containing Institute Name (b) Frame containing Educational Video name (c) Frame containing Professor and Department Name.

All four attributes are not required to be present as seen in Figure 3 nevertheless, one or more attributes may be lacking in the videos. Even with the lacking metadata, we proceed. In the case of missing metadata, we strive to find as many attributes as feasible.

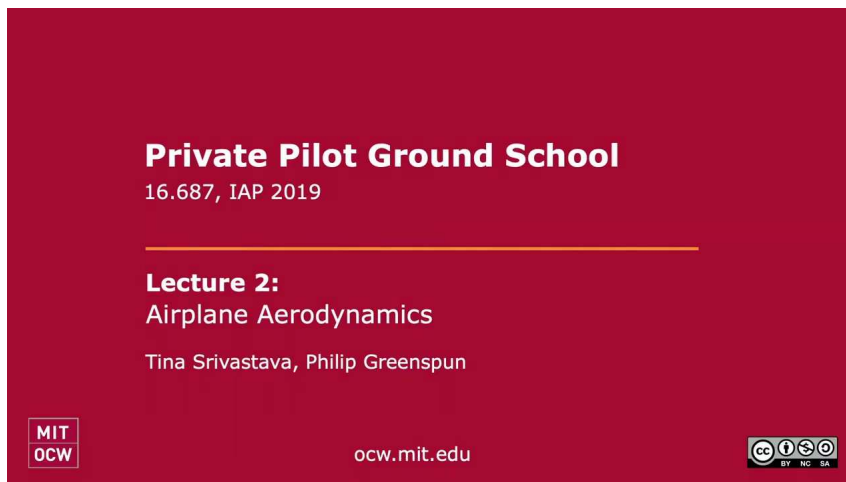


Figure 3. Image to depict missing attributes.

3.2.1. YouTube-dl

YouTube-dl is an open-source video and audio download manager for YouTube and over 1000 other video hosting websites. This library is used to obtain videos, which are subsequently cropped with FFmpeg. Steps followed:

- A CSV sheet containing the *URL*, *start-time* and *end-time* is prepared.
- `ydl.download(URL)` is used to download the entire YouTube video.

- Tags are then added to individual videos. Ex:- "Id of the video, title of the video"
- The videos are cropped using FFMPEG and stored locally.

```
ffmpeg inputFile -ss startTime -to endTime -c copy outputPath/fileName-TRIM.fileExtension
```

3.3. Key-Frame Detection

The process of extracting a frame or combination of frames that have a good representation of a shot is known as key frame extraction. It must retain the most important aspect of the shot while removing the majority of the repeated frames. Video is a vast data object, and its analysis is critical in many domains, including video summarization, video browsing, compression, marketing, etc. We extract crucial frames from videos to decrease the dataset while retaining all important information. These extracted frames provide the most accurate and concise description of the video's content. Various key-frame extraction methods are available. We describe some of the methods below which have been experimented with.

3.3.1. Pixel Subtraction

Image subtraction, also known as pixel subtraction, is the process of subtracting the digital numeric value of one pixel or entire image from another image. This is done largely for one of two reasons: leveling unequal areas of an image, such as half an image with a shadow on it, or identifying differences between two photos. This change detection can be used to determine whether or not something in the image has moved. The frames are considered unique if the mean value of all pixel deviations exceeds a specific threshold.

3.3.2. Frame count method

OpenCV's VideoCapture function is used to capture frames from videos. The frame count function uses the frames captured by OpenCV's VideoCapture method and saves a frame every x seconds. These frames are regarded as one-of-a-kind and are preserved for further processing. Here, x is a variable that can be modified to meet our standards and needs. However, this strategy is inefficient. Although we do not lose much information, we do receive a lot of repeated frames. As a result, it does not always decrease the data and defeats the goal of extracting keyframes.

3.3.3. ffprobe tool

ffprobe is a toolbox in the ffmpeg library. ffprobe collects information from multimedia streams and prints it in a human- and machine-readable format. When a video is fed into ffprobe for keyframe extraction, the frames are classified into three types: I, B, and P.

- Interconnected frames/ Key frames (I Frames)
- Predictive/Predicted frames (P Frames)
- Bi-Directional Predicted frames (B Frames)

So selecting I frames and saving them provides us with the unique frames we desire.

3.3.4. Mittal (2020)

Mittal [10] (The GitHub repository by Varun Mittal) has a pipeline for extracting keyframes. To extract the essential frames, the pipeline goes through three steps. Each phase focuses on a specific objective in finding keyframes and restricts the number of frames to be studied in order to save time. The Figure 4 below depicts these steps, which are followed by a brief description of each phase.

Steps followed:

- Step 1: Extract candidate frames This process entails finding potential candidate frames that could become keyframes. This is done to reduce the number of frames to be processed in the next phase while retaining all relevant frames. In this stage, consecutive frame differences are calculated, followed by the frame with the greatest difference in a window of frames.

- Step 2: Cluster similar candidate frames Clusters of similar frames are formed in this step. Similar frames that are relatively close to one other are clustered together to form a single cluster. Before clustering, each frame is treated to extract the relevant information from it - by scaling the frame, changing it to greyscale, and with some other preprocessing. All that is done to obtain the most useful or significant information from the frame.
- Step 3: Choose the best frames from each cluster. Finally, the best frame from each cluster, as well as all frames that could not be clustered, are identified as keyframes in this step. The best frame is chosen based on the brightness and image blur index. Because all frames in a cluster have similar content, all other frames except the best frame from each cluster are discarded.

For all the attributes - Institute Name, Publisher Name, and Department Name - the efficiency of the FFprobe tool is higher than the Mittal [10]

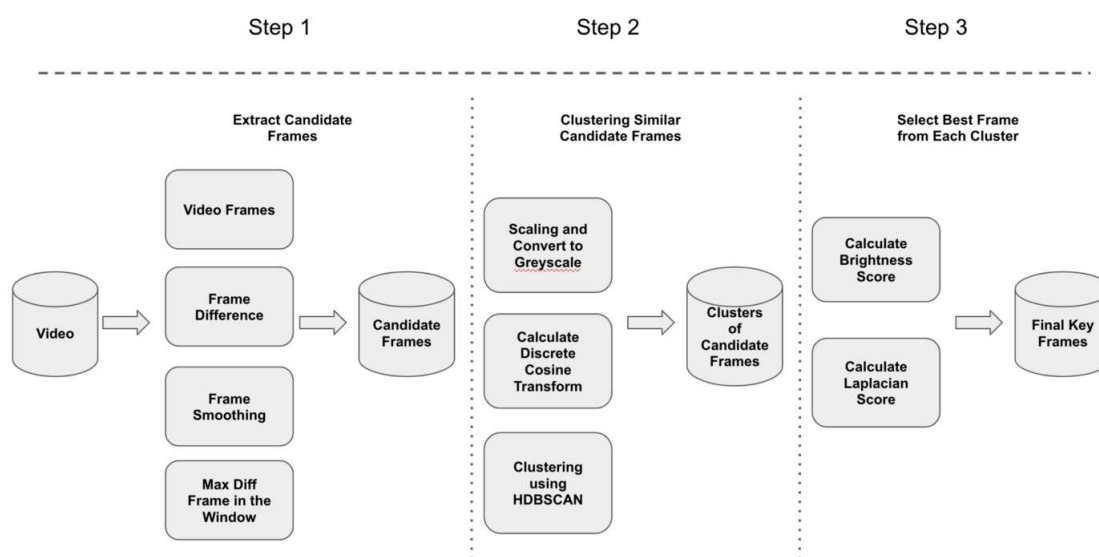


Figure 4. Steps for keyframe detection.

3.4. Text Detection

The process of translating text in an image to string representation is known as text detection. We employ Optical Character Recognition, often known as OCR, for this. We experimented with two types of OCRs.

- Easy OCR
- Tesseract OCR

3.4.1. Easy OCR

EasyOCR, as the name implies, is a Python library that enables computer vision engineers to perform Optical Character Recognition with ease. When it comes to OCR, EasyOCR is by far the easiest approach to implementing Optical Character Recognition. EasyOCR is a lightweight model that performs well on photos with a slightly lower resolution.

3.4.2. Tesseract OCR

Tesseract – the most popular and high-quality OCR library, is an optical character recognition engine with open-source code. Tesseract OCR employs artificial intelligence for text search and image identification. In order to recognize words, the Tesseract OCR engine employs language-specific training data. Like the human brain, the OCR algorithms favor words and sentences that frequently appear together in a given language, and they perform well on high-resolution photos.

We used various pre-processing procedures before sending them to both OCRs, such as

- Converting to binary image
- Blurring the images with Gaussian filter
- Detecting edges through different models

We discovered that preprocessing had little effect on enhancing accuracy because the image quality was poor. We also discovered that Easy OCR produces better results for low-quality images. We can deduce from this that Easy OCR performs some in-built pre-processing to improve the readability of the text.

3.5. Indexing System

A dictionary-based technique was utilized to categorize three qualities, Publisher Names, Institute Names, and Department Names. For these three qualities, a library called *fuzzy-wuzzy* is used.

3.5.1. *fuzzy-wuzzy*

Fuzzy String Matching, also known as Approximate String Matching, is the process of locating strings that match a pattern as closely as possible. The technique has a wide range of applications, including spell-checking, DNA analysis and detection, spam detection, plagiarism detection, etc. *Fuzzy-wuzzy* is a Python module that calculates the differences between sequences and patterns using Levenshtein Distance.

In this case, *fuzzy-wuzzy* is utilized to match and map the words in the categories mentioned above in the text produced by EasyOCR. We apply a predetermined set of rules and the Spacy NER Model to map the fourth attribute, Professor Name. Some general assumptions for developing the set of rules are as follows:

- Professor Name is almost always written in a single line
- Professor Name has either Prof. or Dr. as a prefix
- Professor Name always comes in the line succeeding the word "by".

3.5.2. Spacy NER model

Named Entity Recognition (NER) is an application of Natural Language Processing (NLP) that processes and understands enormous amounts of unstructured human language. Entity identification, entity chunking, and entity extraction are other terms for the same thing. Spacy is a highly efficient statistical method for NER in Python that can assign labels to consecutive sets of tokens. It has a default model that can recognise a wide range of named or numerical things such as person, organisation, language, event, and so on.

So we used the Spacy NER model to find professors' name where the set of rules do not return the professor's name. It works well with English-origin names so works well with MIT videos. It is also used in the videos where the set of rules/assumptions do not work, to atleast retrieve the professor's name up to some extent. We attempted to efficiently map the fourth feature, the Professor name, using the criteria and the NER model.

4. Results

The table given below depicts the accuracy in the percentage of each attribute experimented with ocr(EasyOCR/Tesseract OCR), keyframe detection tool(ffprobe,[10])

Category	EasyOCR , ffprobe	TesseractOCR, ffprobe	EasyOCR , Mittal [10]
Publisher Name	88.03	63.01	71.24
Institute Name	88.88	68.54	74.15
Department Name	82.47	65.87	75.07
Professor Name	85.89	64.44	75.60

From the table, we can observe that Easy OCR, ffprobe which is the baseline model is giving the best results when compared with the other two.

References

1. Yan, X., Gilani, S. Z., Qin, H., Feng, M., Zhang, L., and Mian, A. S. (2018). Deep keyframe detection in human action videos. *CoRR*, abs/1804.10021.
2. Gawande, U., Hajari, K., and Golhar, Y. (2020). Deep learning approach to key frame detection in human action videos. In Sadollah, A. and Sinha, T. S., editors, *Recent Trends in Computational Intelligence*, chapter 7. IntechOpen, Rijeka.
3. Hamad, K. and Kaya, M. (2016). A detailed analysis of optical character recognition technology. *International Journal of Applied Mathematics, Electronics and Computers*, 4:244–244.
4. Ding, J., Zhao, G., and Xu, F. (2018). Research on video text recognition technology based on ocr. In *2018 10th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, pages 457–462.
5. Yang, H., Siebert, M., Luhne, P., Sack, H., and Meinel, C. (2011b). Lecture video indexing and analysis using video ocr technology. In *2011 Seventh International Conference on Signal Image Technology & Internet-Based Systems*, pages 54–61.
6. Yang, H., Siebert, M., Luhne, P., Sack, H., and Meinel, C. (2011a). Automatic lecture video indexing using video ocr technology. In *2011 IEEE International Symposium on Multimedia*, pages 111–116.
7. AV-Portal, T. T. (2014). Tib av-portal. <https://av.tib.eu/>.
8. NDLI (2018). Ndli. <https://ndl.iitkgp.ac.in/>.
9. DCMI (2018). Dcmi. <https://www.dublincore.org/>.
10. Mittal, V. (2020). key frame extraction. https://github.com/varunmittal50/key_frame_extraction_public.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.