# Content Based Document Video Retrieval

Abhimanyu Singh Gaur(B14CS001)
Shubham Jain(B14CS035)
Mentor: Dr. Chiranjoy Chattopadhyay

## Abstract

In this paper, we present a content based video retrieval module based on the document parts of the video. The aim of this module is to extract useful information from a video namely TV channel, sport, team and players. The module is based on a database designed specifically to yield the query result in the least possible time. We have built our module on previously published algorithms through which we extract text from individual frames. The OCR-ed text is then compared to a regular expression to classify the sports in the video. The OCR-ed text is also compared to known list of sport, team and athletes' names (or numbers), to provide a presence score for each player or team. The videos are maintained in a sorted order in the database to ensure fast retrieval of the videos. Extensive experiments show that out method is comparable to several other system doing similar work and yields faster and better results as compared to them.

## Objective

A. Given a query video extract SIMILAR videos from the database.
   Similarity Criteria :
   1. TV Channel
   2. Sport
   3. Team
   4. Player information
B. Given a text query, process the query and extract the videos yielding query results.
C. To create own data structures and database for video indexing and query processing in an optimized way to yield the search results in the least time.

## Motivation

Sports data analysis is becoming increasingly large-scale, diversified, and shared, but difficulty persists in rapidly accessing the most crucial information. Previous studies have focused on the methodologies of sports video analysis from the spatiotemporal viewpoint instead of a document-based viewpoint.This study develops a deeper interpretation of content-aware sports video indexing by examining the insight offered by document part of the videos.

## Research Issues and Challenges

The prime focus of our project is the text part in the videos. Design of an Optical Character Recognition(OCR) system directly related to the video documents is an important issue. Errors in OCR can yield faulty results. Our system is based on database which can't accomodate more than $2^{16}$-1 (=65535) entries. Since the position of text is not fixed and with the varying background in sports matches, text detection and recognition poses to be a challenging problem.

## Methodology

### A. Video Tagging

The aim of this part of the project is to assign appropriate tags to the video specifying TV channel, sport, player and team information. Below flowchart gives an overview of the tagging process:

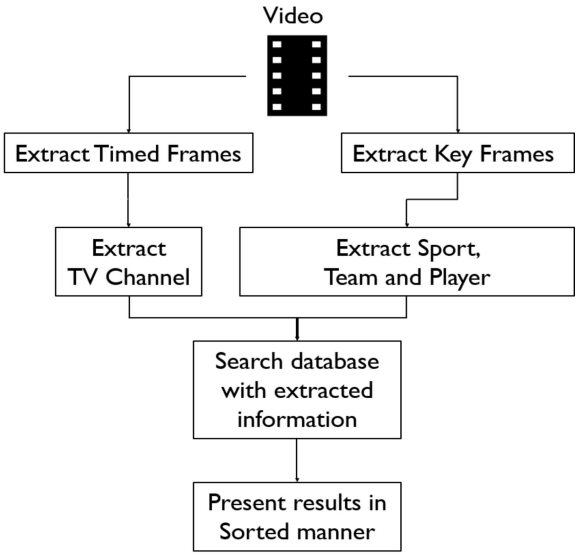



Fig.-1: Overall Video Tagging

Video tagging has 5 main parts:

1. Extracting TV Channel Logo: In order to find the logo of TV channel from the video, we first detect the logo region and then classify the detected region using SVM classifier. The following steps are taken in detection step:
    a. Extracting timed frames: Taking one frame per second from the video
    b. Using Canny edge detector to extract edges from the frames
    c. Calculating time averaged edges from the extracted edges
    d. Double thresholding, to extract prominent edges from the averaged edges
    e. Morphological Closing, followed by Morphological hole filling and Morphological opening, to get connected components.
    f. Applying logo shape constraints to extracted connected components, to decide whether the given component can be a logo or not. Shape constraints include:

i. Boundary distance check: Logo is never connected to the boundary of a frame. So we check it to be at least 5px away from the frame boundary.
ii. Area ratio check: We check that the candidate for logo occupies a certain percentage of corner area.
iii. Aspect ratio check: All the TV channel logos fall within a range of aspect ratios. We check the connected component to be within the expected range.

Once a connected component passes all the above steps, we pass its bounding box in the timed frames to SVM classifier to classify the potential logo region from each timed frame. We use Grid Descriptors[2] as features in SVM classification. Out of the predictions given by SVM, we choose the most occuring label as the logo class. Then we check whether the expected logo corner for the logo class is the same as the corner for current connected component or not. If the corner is same, we accept the logo class as the final TV channel logo for the video. Below figures explain the process for finding TV channel from the given video:

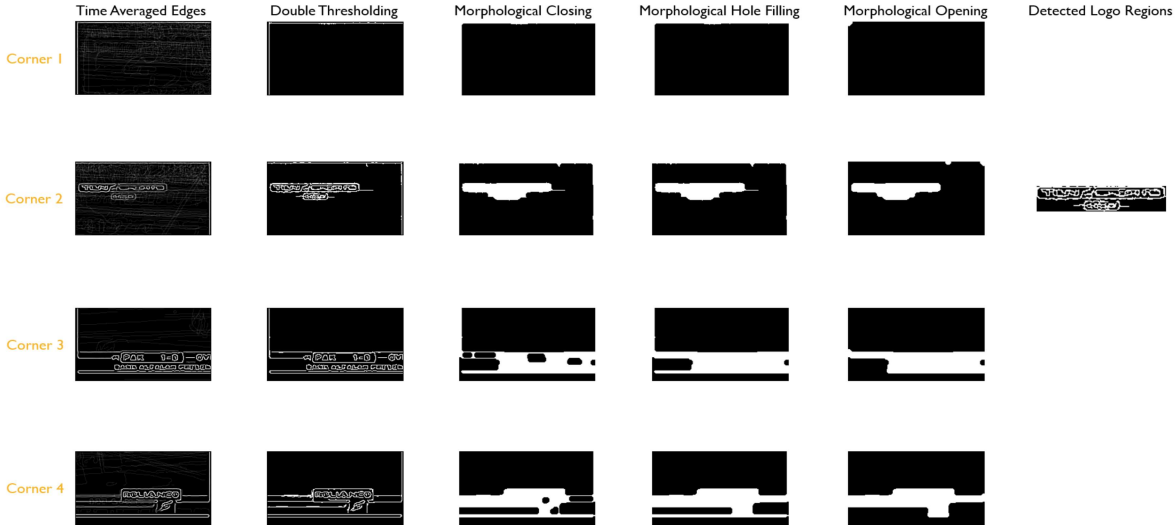
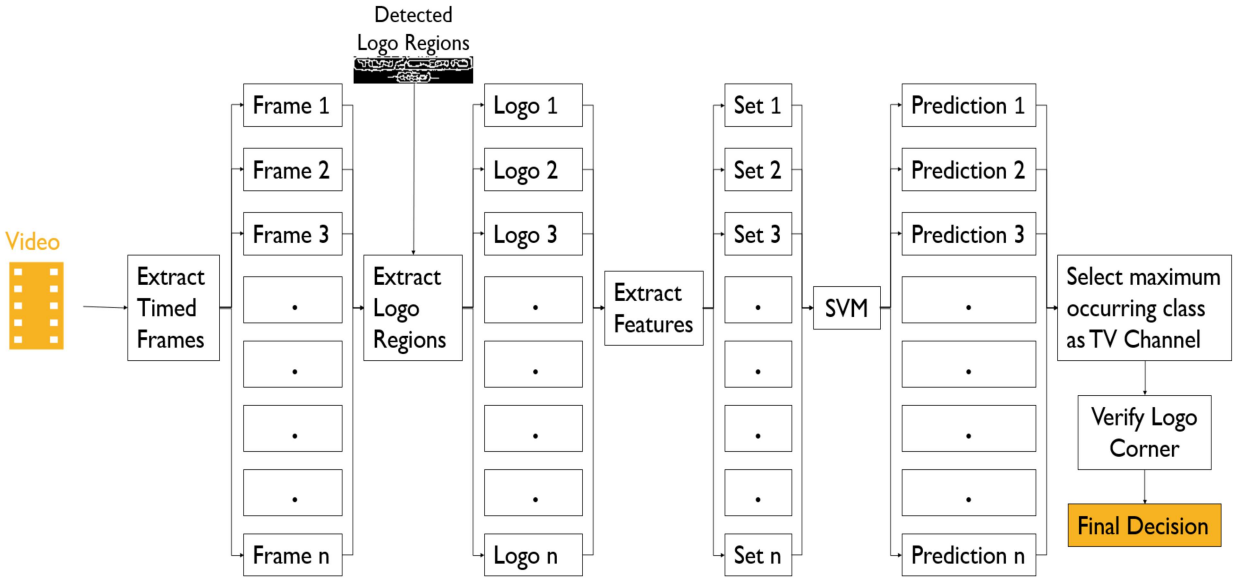



Fig-2: TV Channel Detection

Fig-3: TV Channel Classification



2. Estimating Scoreboard Region: Out of the non-logo corners, we check which corner has the maximum pixels in the double threshold binary image of that corner. Then we appropriately select the scoreboard region as shown in below flowchart:

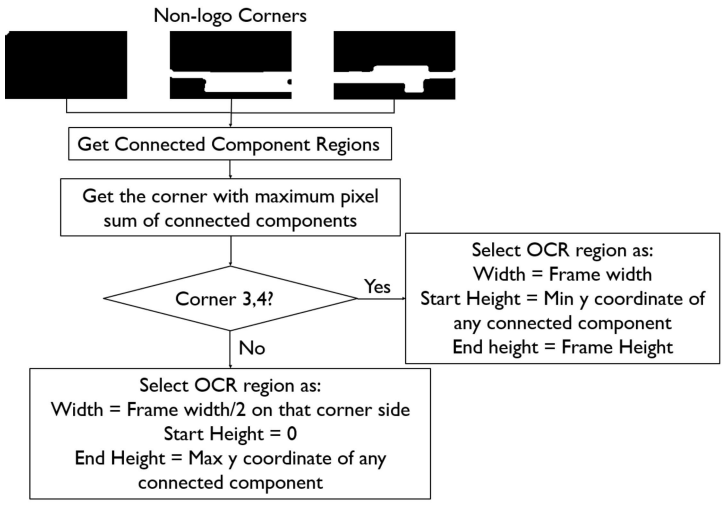



Fig.-4: Selecting scoreboard region to apply OCR

3. Extracting Key Frames (Video Summary): We use color histogram difference to find out candidate key frames from the given video. If the histogram difference between two consecutive frames is greater than the threshold, we save one of the frames as candidate key frame. Then we save the candidate key frames as final key frames if they pass a structural similarity criteria[1] as shown in below flowchart:

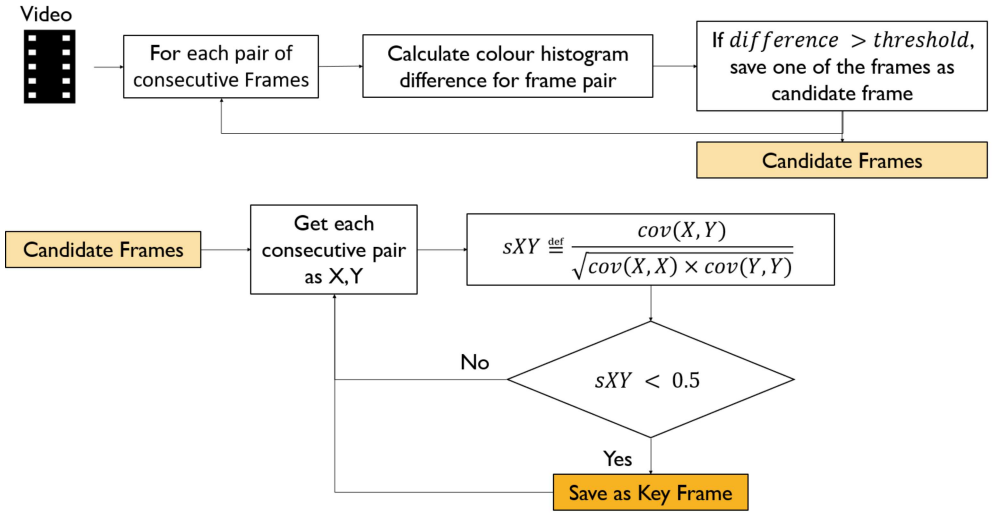



Fig.-5: Key Frame Extraction

4. Applying OCR on Key Frames in Scoreboard Region: Then we apply OCR[6] on the key frames on the detected scoreboard region to get the document text.
5. Extracting Sport, Team, and Player Information from OCRed text: We have prepared a dictionary of Sport, Team, and Player names. On the OCRed text obtained for each

keyframe, we perform a substring matching using the prepared dictionary words. If any substring in the OCRed text matches a word from our dictionaries, we accumulate the tag corresponding to that word in an array for that category. Finally, we select the maximum occuring tag for each category as the sport, team and player tag.

## B. Database Creation

We have implemented our own database, for optimized retrieval and constant time insertion/deletion of the videos. The database has the following directory schema:

- videoDB: Directory to store all the inserted videos.
- indexes: Directory to store all index categories
  - tvChannel: Sub-directory to store all the index files for TV Channels.
  - sport: Sub-directory to store all the index files for Sports.
  - team: Sub-directory to store all the index files for Teams.
  - player: Sub-directory to store all the index files for Players.
- dbRecords.dat: File to store records of inserted videos in the database.
- dbRecordsCount.dat: File to store the count of inserted videos in the database.

The structure of a record in the dbRecords.dat file is as follows:

```
Struct Record
{
        uint16 name;
        char[4] extension;
        IndexCategory[4] indexMap;
}
```

Where IndexCategory has the following structure:

```
Struct IndexCategory
{
        uint16 CategoryClass;
        uint32 positionInIndexFile;
}
```

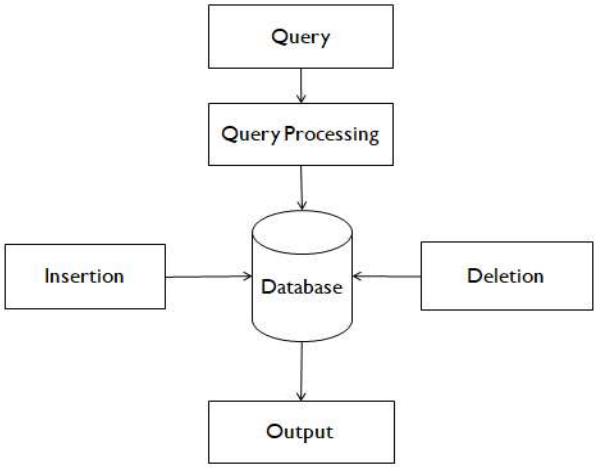



Fig.-6: Working of Database

We assign the name to new video being inserted as one more than the count of videos currently stored in the database. We create a index file for each class in each index category, to store the video names for that class in sorted order. There are three main operations supported by our database implementation:

1. Insertion
   a. Inserting videos in the database requires performing video tagging on videos to find out the index categories and hence update the index files in the database accordingly. This is a constant time operation in our implementation.
   b. We have ensured ACID properties in our implementation. So while inserting a video, the database is consistent.
      i. Atomicity : Changes in the database are reflected completely

ii. Consistency : No data is lost or created automatically
iii. Isolation : All changes are independent from each other
iv. Durability : The database remains even if the system fails.

2. Deletion
   a. Deleting videos in the database requires perform following operations:
      i. Deleting video from the videoDB directory.
      ii. Assigning a flag value of '0' to the name in the record of the selected video in dbRecords.dat file, to indicate that the video is deleted and incrementing the header in the dbRecords.dat file.
      iii. Updating the corresponding index files to mark the video deleted in them too, as well as incrementing the header of the index files.
   b. Deletion is a constant time operation, as we only perform a pseudo deletion from the database files. So, rewrite is not necessary after deletion.
3. Retrieval
   a. Searching videos in the database requires performing intersection of the appropriate index files according to the required search tags in the query for each category.
   b. Searching in worst case is a polynomial time algorithm (O($n$)), where $n$ is the maximum number of videos in any of the index files being intersected. We are able to perform intersection in O($n$), because we are storing the video names in index files in sorted order.
   c. We have also worked to make it output sensitive, i.e., if only 10 videos are required as the search output then the searching algorithm stops as soon as 10 videos are found matching the given search criteria.

## C. Query Processing and Searching

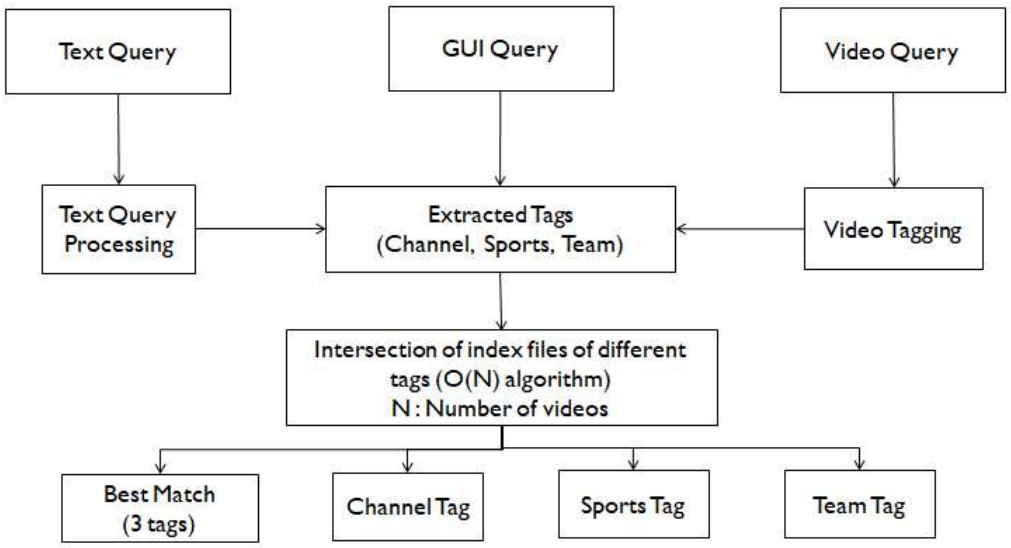



Fig.-7: Module for query processing and searching

We can process three types of queries:

1. Text Query
    a. A Dictionary of key:value is prepared where the key corresponds to any word and value corresponds to the different tags namely channel, sports, player and team.
    b. For the given text query, all the words are processed one by one and compared with the keys in the dictionary.
    c. If the word corresponds to a key in the dictionary then that word represents a tag and hence the value corresponding to the tag is assigned to the word.
    d. After processing the full query, we get the index category tags corresponding to the matched words. For example :

**Input Text**

Select videos from database having Tendulkar Cricket Pakistan SonyLiv

**Output Index Categories**

Team : Pakistan
Player : Tendulkar
Channel : SonyLiv
Sport : Cricket

    e. We have also prepared a dictionary of key:value pairs for each index category, where value represents the index class for that index category.
    f. Once we know index category, we also find the index class for the matched word in dictionary to get the resultant tags.
    g. A database search corresponding to the resultant tags yields the final result.

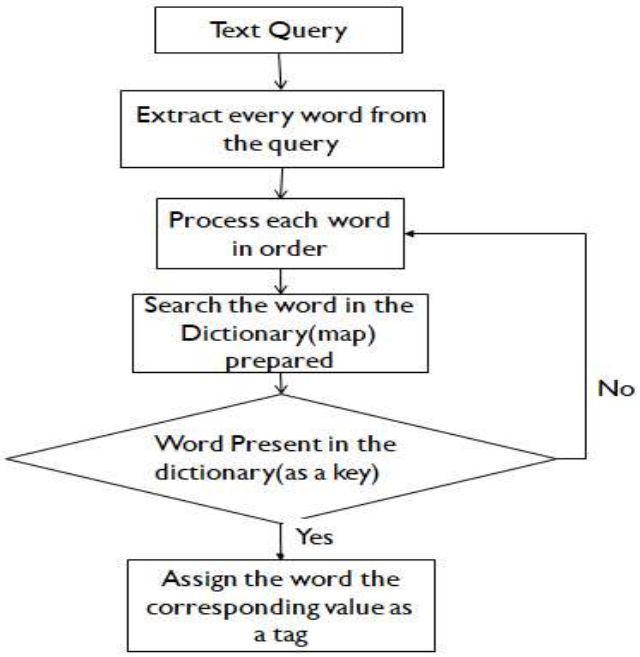



Fig.-8: Text Query Processing

2. Video Query
    a. In this we need to use the video tagging method as defined in section 1 of methodology to identify the tags in the query video.
    b. The intersection of the index files corresponding to the different tags yields the final result.
3. GUI Query
    a. In this we ask the users for values for the different tags from the drop down list.
    b. The user specifies the values for the type of video he wants to search.
    c. Channels supported : Sony Liv,Ten Sports, Sony Six, Star Sports, DD Sports
    d. Sports supported : Cricket, Football, Tennis and basketball.
    e. The intersection of the index files corresponding to the different tags yields the final result.

## Results and Findings

We tested our developed system on 500 videos of different TV channels and different sports with the following results :

- Channel : While most of the TV channels score almost 100%, the average accuracy remains at 96%.
- Sports : This seemed to work well in almost all the cases except in football because its regular expression doesn't have any distinguishing feature. Its average accuracy remains around 90%.
- Player : This had an accuracy of around 84% as well because in many cases the player information was not recognized by the OCR.
- Team : This had almost the same accuracy as player because player information also plays an important role in identifying the team. The accuracy we observed was 88%.

These are the main cases for poor performance instances :

- When there is no motion in the video and the background is very complex, confounding mask extraction.
- The second case is the presence of the alternate static contours (e.g. text lines, etc.) in the proximity of the actual logo. In this case, edges of those static contours become connected to the edges of TV logo and lead to deformations in the logo mask.
- Poor functioning of the OCR resulting in unrecognized results.

## Future Work

Till now our system gets the player information from the scoreboard itself which is itself not very reliable since having a player information in the scoreboard everytime is quite unlikely. So player information can be extracted from the the names on the back of the tshirts and then by checking its confidence score with a database of all the players. Our database is based on the principle that we are not altering its size when the database is more than half empty and hence reallocation techniques can be applied to save a lot of space when deleting videos from the database.This system can be further improved to incorporate more sports.

## Conclusion

In this work, we have developed a fully automatic system for content based video retrieval which consists of Video tagging, database creation and indexing, and query processing to achieve results in optimal time. Further to improve our approach, regular expressions can be improved to incorporate errors due to OCR. We found a number of difficult cases where extracting player information becomes near impossible.

## References


[1] Yunyu Shi et. al., A Fast and Robust Key Frame Extraction Method for Video Copyright Protection, Journal of Electrical and Computer Engineering, Volume 2017.

[2] Nedret OZAY, Bulent Sankur, “Automatic TV logo detection and classification in broadcast videos”, 17th European Signal Processing Conference (EUSIPCO 2009) Glasgow, Scotland.

[3] Jiri Matas, Keril Zimmermann, “Unconstrained Licence Plate and Text Localization and Recognition”, Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems Vienna, Austria, September 2005.

[4] Ngo, CW. & Chan, “Video text detection and segmentation for optical character recognition”, CK. Multimedia Systems (2005) 10: 261.

[5] Anirudh Vyas, Sangram Gaikwad, Chiranjoy Chattopadhyay ; "A Graphical Model for Football Story Snippet Synthesis from Large Scale Commentary", IAPR International Conference on Pattern Recognition and Machine Intelligence (PReMI'17), Kolkata, India (Dec 2017).

[6] Roth Mathias, “Free OCR API”. Retrieved from https://ocr.space/