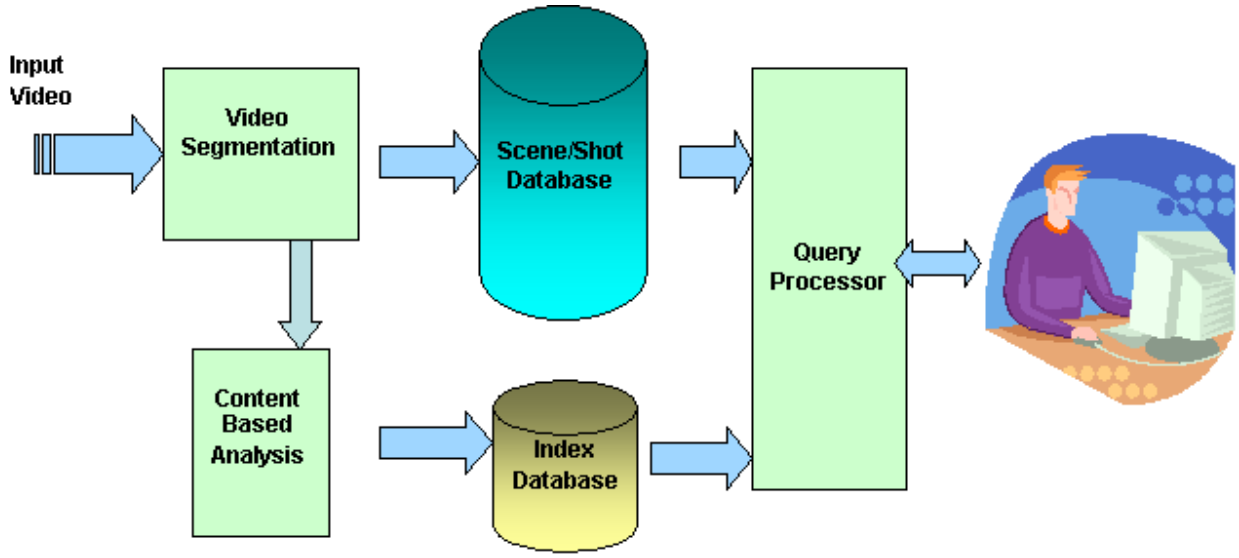
VIDEO INDEXING AND RETRIEVAL REPORT



Rupsi Kaushik

08.04.2019

INTRODUCTION

In today's digital world, the substantial growth in different forms of multimedia resources has highlighted the need for better approaches to characterize dynamic data. A major example of dynamic multimedia data that needs better structure for indexing and retrieval is video data. Over the recent years, the flood of complex and unstructured video content has caused video retrieval to be one of the fastest growing topics in multimedia research [1]. In this paper, a definition of Video Indexing and Retrieval (VIR) and its similarities to the traditional Information Retrieval (IR) concepts will be given. Next, the daily impact of VIR, its typical application in our society, and its industry use will be briefly presented. This paper will also aim to illustrate the main approaches, as well as some common techniques and algorithms for VIR.

Definition of Video Indexing and Retrieval

VIR, in simple terms, is the process of indexing and fetching video data from within large collections. Video data commonly consists of texts, sounds, and images, with a time dimension. Videos have a hierarchical organizational structure, meaning that videos can be further decomposed into scenes, shots, and frames. Videos are not only dynamic in nature but also contain unique features such as motion features, audio features, and object features, making effectively searching over video content very challenging [2].

Links to Traditional IR

The main goal of traditional IR is to index and retrieve relevant textual documents that satisfy an information need from within large collections [3]. VIR serves to satisfy the same objective but with the focus of indexing and retrieving video data. Due to the fact that VIR is simply an extension of traditional IR , there are many observable links between them, some of which will be underlined in the following paragraph.

In traditional IR and VIR, data gets preprocessed and indexed before retrieval takes place. The notion of clustering is applicable in both fields, which means that algorithms such as K-means, an iterative algorithm that keeps track of cluster means, can be applied to groups of shots, as well. Similarity and ranking scores are calculated among keyframes in VIR, as well as among documents in IR. Finally, the performance evaluation methods, namely precision and recall, can be applied to both methods. The only difference is that in VIR, precision refers to the ratio of number of videos retrieved that are relevant to the query over total number of retrieved videos. Recall refers to the ratio of number of relevant videos retrieved over the total relevant videos in the database [14].

The Impact, Application, and Industry Use of VIR

VIR is an important field of research because it seeks to make the retrieval of meaningful content more conveniently accessible. As mentioned before, in this age of high volumes of multimedia data, proper VIR tools can make a world of difference.

There are many industry uses of video retrieval and many more being proposed in important fields like health care. As users search for a scene of their favourite movie or search for a highlight of a soccer game, video retrieval is being implemented. The Web itself is a multimedia system where video data can be found in digital libraries, publications, broadcasting, and entertainment [4]. Large amounts of video can be found online, such as on YouTube and Google Video, as well as on channels like CNN, where videos are indexed and archived frequently. On YouTube, meta-data information, such as title and description, is used in order to index videos. After that, the viewer interaction with the video is used to rank these videos. 500 hours of video is uploaded every minute on YouTube, which shows how much of an integral role VIR plays for this search engine [5].

MAIN APPROACHES

From a text-based approach to sound-based approach, there exists various options for indexing and retrieving video information [6]. Currently, the majority of web based video retrieval systems index and retrieve videos based on text data. In this approach, sections containing textual information in a frame are identified and the video is annotated accordingly. It is clear to see that this approach cannot possibly capture enough video information to be able to provide optimal results. For this reason, Content Based Video Retrieval (CBVR) has been an interest of research and is said to be “the best practical solution for better retrieval quality.” [7]

An Overview of Content Based Video Retrieval

CBVR is the process of retrieving relevant video data from large collections, based on the content information of the video. Deriving content information includes but is not limited to extracting shape, colour, and object features that are present in video frames. These various extracted features are indexed before being stored in the database and are the basis for classifying and retrieving relevant videos. When a query is prompted, the features are extracted from the query and matched with the shots in the database, retrieving the relevant results [7]. In the following sections, the main approaches to CBVR will be presented.

Segmentation

During segmentation, a video sequence gets decomposed into elementary shots, which are composed of a sequence of frames. In this process, the primary concern is being able to find a quantitative measure that is able to capture the frame-to-frame differences [6], [7], [9]. While there are numerous approaches to segmentation, this paper will focus on illustrating only some of the common techniques.

As video data can be both temporal and static, video segmentation techniques can, therefore, be temporal, spatial, or spatio-temporal. In spatial segmentation, image segmentation methods can be applied to the static frames. On the other hand, temporal segmentation, also known as shot detection, is able to locate temporal differences in video information, such as color and motion [8]. One of the simpler approaches to detect temporal information during shot detection is through the use of pixel differences. In this approach, if certain pixels have differences exceeding a threshold value, then a boundary can be detected. Another popular approach is to use local and global histogram comparisons, which will cover frame differences rather than just pixel-wise discrepancies. A local histogram comparison encompasses a combination of a block-based approach and histogram-based approach, taking into account spatial information. In global histogram comparisons, a boundary is declared if the absolute sum of histogram differences between adjacent frames surpasses a threshold. Discrete Cosine Transform coefficients can also be used for segmentation for MPEG compressed videos [9].

Classification and Indexing

During this process, essential information such as motion, color, and object features are obtained in order to classify segmented shots, which determines the classification of videos. A video can contain many shots which suggests that using every frame in order to retrieve a video is computationally costly. For this reason, representative frames, or keyframes, are often used in order to represent a shot and create a shot index. In this type of retrieval, image IR techniques can be applied in order to extract features from keyframes. As each shot and its corresponding keyframes are linked to one another, a shot can be searched by simply identifying its key-frame [7]. However, with this technique, the questions of how many representative frames to select per shot and how exactly to select them arises. Firstly, it is possible to keep it simple and choose one frame per shot, at the expense of losing content length and change information. Moreover, it is possible to choose the number of representative frames based on the length of the video, which solves the problem of capturing content length information. Lastly, it is possible to take length and content information by dividing shots into sub-shots and selecting keyframes from those subshots. For the question of how to select these representative frames, there are also different techniques, with

their own advantages and disadvantages. The first method assumes that a video segment can be described by the first few frames and, hence, selects the first frame of the shot as the representative frame. The second method aims to match a frame within the shot that is most similar to a defined average frame, choosing the most similar one as the representative frame. In the third method, the histogram of the frames are averaged and the frame that is the closest to this averaged frame is chosen. In the last method, there is a notion of background and foreground division. A background is created from the background of all the frames and the main foreground of all the frames is superimposed onto this background [9].

Feature Extraction

As mentioned in [7], the primary features that need to be extracted from frames are color, object shape, and motion. The color and object features can be obtained using histograms. The motion features, such as motion content, panning, uniformity, and tilting can be obtained using two dimensional motion and color histograms.

Similarity Measures

As previously mentioned, video retrieval is based on the similarity between the query features and the feature vectors in the database. The most common method of determining similarity is using Euclidean distance, where minimum Euclidean distance is the best similarity. Kullback-Leiber distance can also be used. In some cases, Neural Network cluster can be used to cluster similar shots and classify videos to the best matching cluster [7].

A Working Example and Algorithms

This section of the report aims to help visualize approaches to VIR by bringing together and applying the aforementioned concepts in a working scenario. It will introduce a proposed approach to CBVR and elaborate on some of the common algorithms.

Object Based Video Retrieval Using Multiple Features

As found in the research article [10], in this type of retrieval, the focus is taken away from frame based retrieval and put on object based retrieval. It proposes to firstly segment the video into objects before continuing with feature extraction. 15 frames are chosen as representative frames for each shot such that they represent visual information in the rest of the shot. In order to reduce computational complexity, the Graph cut Segmentation method is used to segment the objects from the representative frames. The proposed algorithm extracts color, shape, and edge features from the frames in order to store in the feature database. Based on this, similar objects are grouped together. The

features of the query object are compared with the groups through a distance measure and if a matching object is found, then all the corresponding frames are retrieved.

Graph cut Segmentation Algorithm Illustration

The goal of the Graph cut Segmentation algorithm in VIR is to segment the main objects out of a frame. Here, we represent frames, which are still images, as fully connected graphs. There are pixels that are represented by nodes, there is an edge between every pair of pixels (p,q) , and there's a weight w_{pq} that measures similarity for each edge. This algorithm aims to cut the graph in a way that the least similar links between segments are deleted. This means that the similar pixels should be in the same segments, while distinct pixels should be in different segments. There are many methods to measure similarity, including similarity of distance, color, texture, and much more [11]. There is a source node (S), which represents the foreground, and a sink node (T), which represents the background. The segmentation itself can be done by finding the minimum cut in a graph, a cut "whose cutset has the smallest number of elements (unweighted case) or smallest sum of weights possible." The minimum cut algorithm, thus, cuts through edges with least weight (least capacity), essentially separating foreground from background [12].

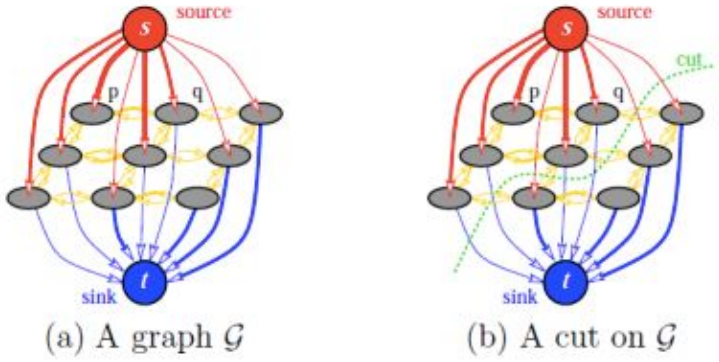


Figure 1: An illustration of a Graph cut Segmentation on a graph

Ford-Fulkerson's Algorithm for Maximum Flow

This algorithm can be used in order to determine the minimum cut for graph segmentation as the algorithm states that maximum flow is equal to the capacity of the minimum cut.

Summarized Steps from [13]:

1. Find an augmenting path - a path with non-full forward edges or non-empty backward edges

2. Compute the bottleneck weight - edge in the selected path with the smallest weight
3. Augment each edge and the total flow

Example of Minimum Cut

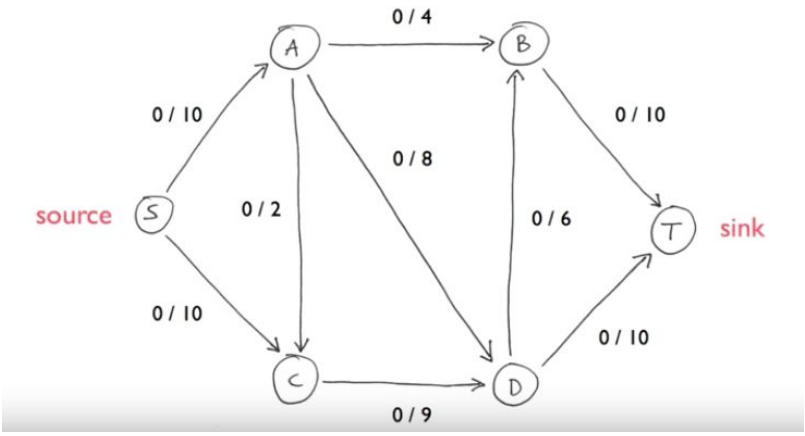


Figure 2: A graph with source node S and sink node T

Step 1 and 2: Let's assume a path from S, T that goes through nodes S, A, and D. The bottleneck capacity is A, D with weight value of 8 and flow value of 0.

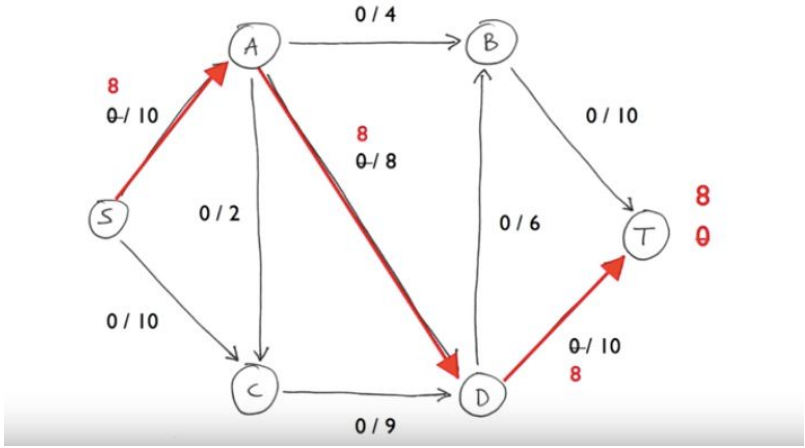


Figure 3: An augmented graph

Step 3: Augment the edges and total flow.

Continue on with the next path with the same steps until there are no more paths from S to T. In this case, the maximum flow value is 19, which is our capacity of minimum cut.

Discrete Cosine Transformation

In [2], Discrete Cosine Transformation (DCT) coefficient based block matching method is used in order to segment the video into shots. It is illustrated that, with this DCT method, a frame is divided into 8X8 blocks, for which a 2D Discrete Cosine transform is calculated. Blockwise comparison is then performed for adjacent frames. If the comparison exceeds a threshold then a shot is detected and similar shots are grouped together. [12]

$$\mathbf{D(u,v)} = \frac{2}{M*N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \mathbf{P(x,y) * \cos\left(\frac{(2x+1)un}{2M}\right) * \cos\left(\frac{(2y+1)vn}{2N}\right)}$$

Figure 3: The equation for two-dimensional DCT from [12]

Example of DCT Calculation

1	3
2	0

Let's say we have this 2X2 image, where M = 2 and N = 2. We want the DCT for points (0,0) and (0,1).

$$D(0,0) = \frac{1}{M*N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} P(x,y), \text{ since } \cos(0) = 1.$$

$$= \frac{1}{4} [1+3+2+0]$$

$$= 1.5$$

$$D(0,1) = \frac{2}{M*N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} P(x,y) * \cos\left(\frac{(2x+1)un}{2M}\right) * \cos\left(\frac{(2y+1)vn}{2N}\right)$$

$$= \frac{1}{2} [\cos(\Pi/4) + 3\cos(3\Pi/4) + 2\cos(\Pi/4)]$$

$$= 0$$

Colour Feature Extraction

In this report, segmentation algorithm and techniques have been touched on. This section will touch a little on the steps for colour quantization for feature extraction as proposed by the research paper [10]. In this paper, a modified K-means Clustering algorithm is proposed for color quantization. In this modified algorithm, the local maximum of the histogram is selected as the initial number

of centroids. Secondly, some number of vectors are chosen at random and assigned to the nearest cluster. Thirdly, the new centroids are computed for the clusters. The cluster that is used the most is divided into new clusters. This process is repeated from step two until the desired number of clusters is found. The classic K-Means algorithm is then used after the initial centroids have been selected. After color quantization, color histogram is calculated, creating a feature vector.

CONCLUSION

In this report, we discussed the growing need of VIR in today's multimedia age and its application to society. The report defined VIR and its connection to traditional IR. The main approaches to VIR, including an overview of CBVR, was also presented. The process of segmentation, feature extraction, and similarity measures in CBVR were thoroughly described. Finally, a practical example of a CBVR was illustrated, taking a look at a proposed approach using Graph cut Segmentation algorithms, DCT coefficient based block matching, and colour feature extraction. With these illustrations, it is clear to see that VIR is a complex yet captivating field of research with plenty of room for learning and novel discoveries.

REFERENCES

[1] Patel, B., & Meshram, B. (2012, October). CONTENT BASED VIDEO RETRIEVAL. Retrieved April 07, 2019, from https://arxiv.org/pdf/1211.4683.pdf

[2] Kanagavalli, R., & Duraiswamy, K. (2012). Shot Detection Using Genetic Edge Histogram and Object Based Video Retrieval Using Multiple Features. Retrieved April 07, 2019, from https://thescipub.com/pdf/10.3844/jcssp.2012.1364.1371

[3] Barrière, C. (2019, January 11). Introduction to Information Retrieval [PowerPoint Slides]. Retrieved April 07, 2019.

[4] Chen, J., Dr. (2008). Video Structure, Representation, and Image Retrieval system. Retrieved April 07, 2019, from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.472.5559&rep=rep1&type=pdf

[5] What Makes Your Video Rank in Search Results on YouTube? (2019, March 25). Retrieved April 07, 2019, from https://www.3playmedia.com/2019/03/25/rank-search-results-youtube/

[6] Insani, R. W. (2014, December 17). Video Indexing and Retrieval. Retrieved April 07, 2019, from https://www.slideshare.net/rachmatwahid/slide-video-indexing-and-retrieval

[7] Ansari, A., & Mohammed, M. H. (2015, February 7). Content based Video Retrieval Systems - Methods, Techniques, Trends and Challenges. Retrieved April 07, 2019, from https://pdfs.semanticscholar.org/0446/e221787d9551b90e28dba8090d9f38601717.pdf

[8] Dhiman, P., & Dhanda, M. (2016, April). A Review on Various Techniques of Video Segmentation. Retrieved April 07, 2019, from http://www.ijirst.org/articles/IJIRSTV2I11077.pdf

[9] DElab. (n.d.). Content-based Video Indexing and Retrieval [PowerPoint Slides]. Retrieved 2019, from

delab.csd.auth.gr/courses/c_mmdb/VideoRetrievalByContent.ppt

[10]Kanagavalli, R. (n.d.). OBJECT BASED VIDEO RETRIEVAL USING MULTIPLE FEATURES. Retrieved April 07, 2019, from http://shodhganga.inflibnet.ac.in/bitstream/10603/22919/11/11_chapter 6.pdf

[11]Li, F. (2011, October 6). Clustering and Segmentation. Retrieved April 07, 2019, from http://vision.stanford.edu/teaching/cs231a_autumn1112/lecture/lecture6_clustering_and_seg_p2_cs231a.pdf

[12] Saberi, A. (2013, March 16). Digital image processing: P043 Graph Cuts. Retrieved 2019, from https://www.youtube.com/watch?v=HMGX8HXskKk

[13] Operations Research Methods. (2009, October 28). Max-Flow Problem and Augmenting Path Algorithm. Retrieved 2019, from http://www.ifp.illinois.edu/~angelia/ge330fall09_maxflowl20.pdf

[14] ESAT Publishing House. (2014, August 20). Content based video retrieval system. Retrieved 2019, from https://www.slideshare.net/ijreteditor/content-based-video-retrieval-system