Video Content Extraction: Scene Segmentation, Linking and Attention Detection

by

Yun Zhai
B.S. Bethune-Cookman College, 2001

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the School of Electrical Engineering and Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Summer Term
2006

Major Professor: Mubarak Shah

Abstract

In this fast paced digital age, a vast amount of videos are produced every day, such as movies, TV programs, personal home videos, surveillance video, etc. This places a high demand for effective video data analysis and management techniques. In this dissertation, we have developed new techniques for segmentation, linking and understanding of video scenes. Firstly, we have developed a video scene segmentation framework that segments the video content into story units. Then, a linking method is designed to find the semantic correlation between video scenes/stories. Finally, to better understand the video content, we have developed a spatiotemporal attention detection model for videos.

Our general framework for temporal scene segmentation, which is applicable to several video domains, is formulated in a statistical fashion and uses the Markov chain Monte Carlo (MCMC) technique to determine the boundaries between video scenes. In this approach, a set of arbitrary scene boundaries are initialized at random locations and are further automatically updated using two types of updates: diffusion and jumps. The posterior probability of the target distribution of the number of scenes and their corresponding boundary locations are computed based on the model priors and the data likelihood. Model parameter updates are controlled by the MCMC hypothesis ratio test, and samples are collected to generate the final scene boundaries. The major contribution of the proposed framework is two-fold:

(1) it is able to find weak boundaries as well as strong boundaries, i.e., it does not rely on the fixed threshold; (2) it can be applied to different video domains. We have tested the proposed method on two video domains: home videos and feature films. On both of these domains we have obtained very accurate results, achieving on the average of 86% precision and 92% recall for home video segmentation, and 83% precision and 83% recall for feature films.

The video scene segmentation process divides videos into meaningful units. These segments (or stories) can be further organized into clusters based on their content similarities. In the second part of this dissertation, we have developed a novel concept tracking method, which links news stories that focus on the same topic across multiple sources. The semantic linkage between the news stories is reflected in the combination of both their visual content and speech content. Visually, each news story is represented by a set of key frames, which may or may not contain human faces. The facial key frames are linked based on the analysis of the extended facial regions, and the non-facial key frames are correlated using the global matching. The textual similarity of the stories is expressed in terms of the normalized textual similarity between the keywords in the speech content of the stories. The developed framework has also been applied to the task of story ranking, which computes the interestingness of the stories. The proposed semantic linking framework and the story ranking method have both been tested on a set of 60 hours of open-benchmark video data (CNN and ABC news) from the TRECVID 2003 evaluation forum organized by NIST. Above 90% system precision has been achieved for the story linking task. The combination of both visual and speech

cues has boosted the un-normalized recall by 15%. We have developed PEGASUS, a content based video retrieval system with fast speech and visual feature indexing and search. The system is available on the web: http://pegasus.cs.ucf.edu:8080/index.jsp.

Given a video sequence, one important task is to understand what is present or what is happening in its content. To achieve this goal, target objects or activities need to be detected, localized and recognized in either the spatial and/or temporal domain. In the last portion of this dissertation, we present a visual attention detection method, which automatically generates the spatiotemporal saliency maps of input video sequences. The saliency map is later used in the detections of interesting objects and activities in videos by significantly narrowing the search range. Our spatiotemporal visual attention model generates the saliency maps based on both the spatial and temporal signals in the video sequences. In the temporal attention model, motion contrast is computed based on the planar motions (homography) between images, which are estimated by applying RANSAC on point correspondences in the scene. To compensate for the non-uniformity of the spatial distribution of interest-points, spanning areas of motion segments are incorporated in the motion contrast computation. In the spatial attention model, we have developed a fast method for computing pixel-level saliency maps using color histograms of images. Finally, a dynamic fusion technique is applied to combine both the temporal and spatial saliency maps, where temporal attention is dominant over the spatial model when large motion contrast exists, and vice versa. The proposed spatiotemporal attention framework has been extensively applied on multiple video sequences to highlight interesting objects and motions present in the sequences. We have

achieved 82% user satisfactory rate on the point-level attention detection and over 92% user satisfactory rate on the object-level attention detection.

This work is dedicated to my parents for their passionate and self-giving supports throughout my past twenty years of studies. They have sacrificed many things in their lives to help me get to this point! This work is also dedicated to my dear love, Tian, who has always stood beside me and believed in me when I was in my low times!

Acknowledgments

I would like to express my deepest gratitude to my advisor, Dr. Mubarak Shah, for his insightful guidance, encouragement and generous support. He has provided me with incessant support in each of my steps during my graduate studies. He has always been the source of inspiration and motivation to me. His thoughts not only directed me on how to get research ideas, but also influenced my way of effectively organizing my time and energy, which is critical in one's life. His guidance and support, both academically and personally, has been pivotal to my career. I am very fortunate and honored to have the opportunity to work with him during the past five years!

I would like to thank my committee members, Dr. Charles Hughes, Dr. Niels Lobo and Dr. David Nickerson, for their precious services in my committee and valuable comments on my research work.

Lastly, I would like to thank the entire UCF Computer Vision Group. Many ideas emerged from the frequent discussions between me and my colleagues. I am very happy to work with all these bright researchers!

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Due to the rapid development of video production technology and the decreasing cost of video acquisition tools and storage, a vast amount of video data is generated around the world everyday, including feature films, television programs, personal/home/family videos, surveillance videos, game videos, etc. There necessitates techniques for automatically managing this vast amount of information, such that users can structure them quickly, understand their content and organize them in an efficient manner.

In this dissertation, we present three multimedia processing and content understanding techniques. Firstly, we have developed a stochastic framework for temporal video scene segmentation, which divides video sequences into semantic units. Then, we present a novel semantic linking technique to correlate semantically similar video stories. Finally, we present a spatiotemporal video attention method, which automatically generates spatiotemporal saliency values for interesting objects or activities in video sequences.

1.1 Motivations

Videos are often constructed in the hierarchical fashion: [Frame]→[Shot]→[Scene] →[Video]. The lowest level contains the individual frames. A series of continuous frames with consistent background settings constitute a shot. The video shots are caused by different camera operations, e.g., turning camera on/off, the switching between cameras, and other video editing techniques. Consider the situation where a tourist is recording a video around a monument. He would like to have different views of the monument. First, he takes one sequence for the frontal view of the monument and shuts the camera off. Then he walks to the other side of the monument and records another sequence of the side view. In this case, the entire scene is composed of two shots, which are generated by the on/off operations of a single camera. On the other hand, in movies or TV programs, shots are generated from different cameras and are later appended one after another to constitute the story lines. A scene or a story is a group of semantically related shots, which are a coherent subject or theme. A scene sometimes can be composed of a single shot. For instance, in the example described above, the tourist could have the camera on all the time and record the video continuously. However, more often, scenes are composed of multiple shots, such as movies or TV programs. At the highest level, the entire video is composed of multiple scenes, which results in the complete storyline.

Scenes/stories are the meaningful units of the video. A single shot is insufficient to reveal the story line of the video content. For instance, in feature films, how could one answer a

query related to a suspense scene based only on the content of a single shot? These types of scenes can only be identified with multiple shots showing the increasing tension in the video. In other domains, more often the semantic concepts are difficult to determine by using only a single shot, since they are introduced to viewers over time. Thus, a meaningful result can only be achieved by exploiting the video scenes, which are the interconnections of the shot contents. To achieve this, temporal video scene segmentation is needed. Temporal scene segmentation is defined as a process of clustering video shots into temporal groups, such that shots within each group are related to each other with respect to certain aspects. This is an important and fundamental problem in video processing and understanding. This process provides more meaningful and complete information for understanding the video content compared to the shot-level analysis. Scene segmentation has many applications in various domains. For example, in feature films, scene segmentation provides the chapters that correspond to the different sub-themes of the movies. In television videos, segmentation can be used to separate the commercials from the regular programs. In news broadcast programs, segmentation can be used to identify different news stories. In home videos, scene segmentation may help the consumers logically to organize the videos related to the different events (e.g., birthdays, graduations, weddings, or vacations like city tours, sightseeing).

With the availability of video scenes/stories generated from the temporal scene segmentation, as described above, one can better understand the semantic content of the video. To archive the videos efficiently and retrieve them in future tasks, the inter-video relationship must be discovered. The discovery of such a relationship is usually referred to as the index-

ing process of the videos. To achieve this goal, the videos need to be linked based on their semantic similarities. In this dissertation, we present a semantic linking method for the new video domain. There are many news agencies nowadays that broadcast what is happening around us and around the world. Their reporting is real-time and comprehensive, covering politics, economics, health, sports, etc. Large-scale news networks provide more national and global news, while local stations concentrate more on the regional issues. Due to the variety of audiences, one may only be interested in a few areas or topics, e.g., sports or politics. Thus, finding a particular story that fits to the user's preference is important. Furthermore, even though every news network in the industry claims that their reporting is objective, the actual opinion presented or the attitude of the reporter may be biased and differs from network to network due to the differences in their culture backgrounds. Therefore, watching the same news from multiple sources provides the audience with a more comprehensive and balanced view of a particular story. To accomplish this goal, the semantic linkage between stories must be established. As suggested by the name, semantic linkage between two stories represents their similarity in terms of their semantic contents. For example, two stories that focus on the same news event have strong semantic linkage. On the other hand, stories that have little overlap in their themes have weaker semantic linkage. Other motivations for the semantic linking of stories include finding the most recent stories, tracking the development of the same stories over time, and finding the most interesting stories on a particular date.

Taking a video segment, often we want to better understand what is happening in the scene, such as who is doing what. In this situation, automatic detection of interesting ob-

jects and activities is necessary. Let us consider how humans achieve this goal. Human perception firstly picks the points or regions in an image that stimulate the vision nerves the most before continuing to interpret the rest of the scene. Visual attention simulates the human visual system to automatically produce a saliency map of the image. These attended regions could correspond to either prominent objects in the image or interesting actions in video sequences. Visual attention analysis simulates this human vision system behavior by automatically producing saliency maps of the target image or video sequence. It has a wide range of applications in tasks of image/video representation, object detection and classification, activity analysis, small-display device control and robotics controls. Visual attention deals with detecting the regions of interest (ROI) in images and interesting activities in video sequences that are the most attractive to viewers. For example, in the task of object/activity detection, visual attention detection significantly narrows the search range by giving a hierarchical priority structure of the target image or sequence. Consider the following scenario, a video sequence is captured by a camera that is looking at a classroom entrance. At the time the class is dismissed, the majority of the students will be going out of the classroom. In this situation, if two people are trying to walk back into the room, their actions would be considered "irregular" compared to the rest of the students. Attention analysis is able to quickly highlight the abnormal regions and perform further activity analysis on these regions.

1.2 Proposed Work

We have developed several techniques to solve the problems described in the previous section. First, we present a general framework for the temporal video segmentation by using the Markov chain Monte Carlo (MCMC) technique. We have developed an iterative method to evaluate the segmentation parameters, including the number of scene segments and their corresponding locations. These two parameters are estimated in a statistical fashion using the MCMC technique, which has been used in several applications in the fields of image processing, video content analysis and computer vision in the past few years. Geman *et al.* [28] were the first to apply the MCMC technique in image analysis using the Gibbs sampler. The MCMC technique involving the jump and diffusion method was introduced by Grenander *et al.* [30], and Green [29] further proposed the reversible jumps. It has been applied in sampling and learning by Zhu *et al.* [125]. For 1D signal segmentation problems, Phillips *et al.* has discussed the change-point problem in [82]. Dellaert *et al.* [19] proposed an EM-based technique for solving the structure-from-motion (SFM) problem without known correspondences. The MCMC algorithm [36] with symmetric transition probabilities was used to generate samples of the assignment vectors for the feature points in each frame. Senegas [88] proposed a method for solving the disparity problem in stereo vision. The MCMC sampling process was applied to estimate the posterior distribution of the disparity. Tu *et al.* [98] and Han *et al.* [33] have applied the data-driven Markov chain Monte Carlo (DDMCMC) technique to optical and range image segmentations.

Our developed Markov chain contains three types of updates: shifting of boundaries, merging of two adjacent scenes and the splitting of one scene into two scenes. Due to these updates, the solution can jump between different parameters spaces, i.e., the parameter vector dimension can change, as well as diffuse inside the same space, i.e., the elements in the parameter vector are changed without changing the vector dimension. We assume that each shot in the video has a likelihood of being declared as the scene boundary. Shots with higher likelihoods coincide more with the true boundaries. Initially, two segments are assumed, and they are separated by a randomly selected shot. Then, in each iteration of the updates in the MCMC process, several shots are declared as the scene boundaries. Their likelihoods are accumulated, while the likelihoods of other shots are kept the same. Several Markov chains are executed independently to avoid the possible mis-detections caused by a single chain, and the samples from all the chains are collected for the computation of the shot likelihoods. Finally, the shots with the highest likelihoods in their neighborhoods are declared as the scene boundary locations. One advantage of using the sampling technique is that both the weak and strong boundaries can be detected without defining any specific threshold. We have tested the presented framework on two video domains, home videos and feature films, and we have obtained very accurate and competitive results.

Once the videos are segmented into scenes or stories that possess meaningful semantic content, these logical units can be further linked by their similarities in the context of semantics. We present a framework for the semantic linking of news stories. Unlike the conventional video content linking methods, which are based only on the video shots, the

developed framework links the news video across different sources at the story level. Another advantage is that the developed method uses more semantic features compared to other methods, such as face-related features and textual information. The semantic linkage between the news stories is computed based on their visual and textual similarities. The visual similarity is carried on both of the story key frames, which may or may not contain human faces. To overcome the limitations of the conventional face correlation approach, we analyze the information from the person's body that appears in the video. The detected face region is extended to cover the upper body of the person, and the facial similarity is computed based on the resulting "body" patches. For non-facial key frames, point correspondences between matching images are used to estimate homography, whose goodness is considered as the non-facial visual similarity between key frames. The textual similarity is computed using the automatic speech recognition (ASR) output of the video sequences. The normalized textual similarity is defined for comparison of speech information from different new stories. The proposed method is tested on a large open benchmark dataset. Furthermore, the output of the story linking method is applied in a news ranking task. The matched stories are modelled in a bipartite graph. The graph is segmented into sub-graphs using the connected-components algorithm, and story ranking is performed by analyzing the corresponding component's size. The proposed semantic linking framework and the story ranking method have both been tested on a set of 60 hours of open-benchmark video data from the TRECVID 2003 evaluation forum, and very satisfactory results have been obtained.

In the last portion of this dissertation, we propose a bottom-up approach for modelling the spatiotemporal attention in video sequences. The proposed technique is able to detect the attended regions as well as attended activities in video sequences. Unlike previous methods, most of which are based on the dense optical flow fields, our proposed temporal attention model utilizes the interest point correspondences and the geometric transformations between images. In our model, feature points are firstly detected in consecutive video images, and correspondences are established between the interest-points using the Scale Invariant Feature Transformation (SIFT [59]). RANSAC algorithm is then applied on the point correspondences to find the moving planes in the sequence by estimating their homographies and corresponding inliers. Projection errors of the interest points based on the estimated homographies are incorporated in the motion contrast computation. In the spatial attention model, we have constructed a hierarchical saliency representation. A linear time algorithm is developed to compute pixel-level saliency maps. In this algorithm, color statistics of the images are used to reveal the color contrast information in the scene. Given the pixel-level saliency map, attended points are detected by finding the pixels with the local maxima saliency values. The region-level attention is constructed based upon the attended points. Given an attended point, a unit region is created with its center to be the point. This region is then iteratively expanded by computing the expansion potentials on the sides of the region. Rectangular attended regions are finally achieved. The temporal and spatial attention models are combined in a dynamic fashion. Higher weights are assigned to the temporal model if large motion contrast is present in the sequence. Otherwise, higher

weights are assigned to the spatial model if less motion exists. To demonstrate the effectiveness of the proposed spatiotemporal attention framework, we have extensively applied it to many video sequences, which contain both sequences with moving objects and sequences with uniform global motions. Very satisfactory results have been obtained and presented in this dissertation.

1.3 Dissertation Overview

The structure of this dissertation is as follows: First, we summarize previous works on the target topics in Chapter 2. The stochastic scene/story segmentation method is presented in Chapter 3. Then, the method for the story semantic linking is presented in Chapter 4. Finally, we present the spatiotemporal video attention detection in Chapter 5.

CHAPTER 2

RELATED WORK

In this chapter, we review the current approaches and solutions in the fields of the three proposed problems: temporal video scene segmentation, video semantic linking and spatiotemporal video attention detection.

2.1 Temporal Video Segmentation

Several temporal segmentation methods have been developed for different types of videos. Hanjalic *et al.* [35] proposed a method for detecting boundaries of logical story units in movies. In their work, inter-shot similarity is computed based on block matching of the key frames. Similar shots are linked, and the segmentation process is performed by connecting the overlapping links. Rasheed *et al.* [84] proposed a two-pass algorithm for scene segmentation in feature films and TV shows. In the first pass, potential scene boundaries of the video are initially detected based on the color similarity constraint, *Backward Shot Coherence (BSC)*. Over-segmented scenes from the first pass are then merged in the second pass, based on the

analysis of the motion content in the scenes. Sundaram *et al.* [92] used the audio-visual features of the video in movie scene segmentation. First, two types of scenes, audio scenes and video scenes, are detected separately. Then, the correspondences between these two sets of scenes are determined using a time-constrained nearest-neighbor algorithm. Adams *et al.* [1] proposed the "tempo" for the segmentation of the movies. The "tempo" of a shot is a combination of the shot length and the motion content of shot. The dramatic story sections or events in the movie are detected by finding the zero-crossings of the "tempo" plot. Yeung *et al.* [101] proposed a graph-based representation of the video data by constructing a Shot Connectivity Graph. The graph is split into several sub-portions using the complete-link method of hierarchical clustering such that each sub-graph satisfies a color similarity constraint. These methods are based on the "film grammar", which is a set of production rules of how the movies or TV shows should be composed. For instance, in action scenes, the shots are generally short, and their motion content is high. On the other hand, the shots are long and the visual appearance is smooth in drama scenes. However, these heuristics are not applicable to the other types of videos. For instance, home videos are recorded in a completely "free" style. Shooters are not trained with recording techniques, and often no obvious format or pattern exists in the video. Furthermore, since the rules in the production of films and TV shows are different, the methods for these two domains of videos cannot be used interchangeably.

There is a particular interest in the story segmentation of the news broadcast videos. Hoashi *et al.* [38] has proposed an SVM-based news segmentation method. The segmen-

tation process involves the detection of the general story boundaries, in addition to the special type of stories, e.g., finance report and sports news. Finally, anchor shots are further analyzed based on audio silence. Hsu *et al.* [39] proposed a statistical approach based on discriminative models. The authors have developed *BoostME*, which uses the Maximum Entropy classifiers and the associated confidence scores in each boosting iteration. Chaisorn *et al.* [11] used Hidden Markov Models (HMM) to find the story boundaries. The video shots are first classified into different categories. The HMM contains four states and is trained on three features: type of the shot, whether the location changes (true or false) and whether the speaker changes (true or fase). These methods were developed based on the unique characteristics of news video. The video shots are commonly classified into news program related categories, e.g., anchor person, weather, commercials and lead-in/out shots. These categories are not available in other domains of videos, such as home videos or feature films. Furthermore, the news segmentation methods usually involve the special treatment on the anchor person shots, which exist only in news videos.

2.2 Semantic Linking of Videos

Semantic video linking is related to the problem of video matching, which is a long studied problem. Hampapur and Bolle [32] proposed a video copy detection method by exploiting multiple video features. These features are image-based and computed from video keyframes. Hoad and Zobel [37] have proposed a fast video matching technique using the

signature alignment. The videos are represented by a sequence of number, each of which is computed based on the individual frames. Video matching is achieved by comparing the representation sequences. Authors in [2] and [124] have proposed similar approaches based on the string matching techniques, where small video elements (frames or shots) are represented by numerical features, which are used in the distance/similarity measures. Various frameworks have been proposed for shot-level video matching. Tavanapong and Zhou [95] has proposed shot clustering method for the purpose of video scene segmentation. The shot image is constructed from the corresponding key-frames. The links for grouping the shots are established by comparing the sub-blocks in the shot images. Odobez *et al.* [73] used the spectral technique to cluster the video shots. Multiple key-frames were employed for representing a single shot. The color histograms were used for the visual similarity measure. The correlation was further scaled by the temporal distance. Sivic *et al.* [91] extended their object grouping framework for clustering the video shots in the movie. First, an object is extracted by a series of actions, including feature extraction, feature tracking, homography estimation and object grouping. The 3D structure of the object is computed and used for searching the same object in other shots. Ngo *et al.* [70] has proposed a two-level hierarchical clustering method for grouping the shots. Both color and motion information are used as features. A color histogram in the YUV space is computed for each shot from its discrete cosine (DC) images and used in the first level clustering. Temporal slice analysis is used to compute the tensor histogram, which is a motion feature, for the second level clustering. Cheng and Xu [16] proposed a structure called *Shot Cluster Tree*. First, the shots that are

visually similar and are adjacent in time are grouped into shot groups. The shots groups are later merged into shot clusters based on their content similarity. The color histogram of the key-frame of each shot is used as the similarity feature.

Several video matching techniques have been designed for the story-based linking of news videos. Ide *et al.* [41] proposed a database management system for TV news programs. The news programs are first segmented into topics. The topics are further threaded into the video database in a chronological order, based on the semantic linkage between each other. Kender and Naphade [47] proposed a story tracking method utilizing the mid-frequency high-level semantic features. The similarity between stories is defined in terms of the high-level feature correlation, and normalized cut method is used to cluster the stories based on their similarities. Zhang *et al.* [120] proposed a simpler version of the spectral clustering technique. The stories from two sources are modelled as the vertices in a bipartite graph, and the computation of the eigenvalues for the similarity matrix is dramatically reduced. The clustering for the stories is based on the analysis of text information, e.g., term frequency and inverse document frequency (TF-IDF), and the clustering of video shots is based on the mid-level or high-level visual concepts.

2.3 Spatiotemporal Video Attention

Visual attention detection in still images has been long studied, while there is not much work on the spatiotemporal attention analysis. Psychology studies suggest that human vision

system perceives external features separately (Treisman and Gelade [97]) and is sensitive to the difference between the target region and its neighborhood (Duncan and Humphreys [22]). Following this suggestion, many works have focused on the detection of feature contrasts to trigger human vision nerves. This is usually referred as the "stimuli-driven" mechanism. Itti *et al.* [42] proposed one of the earliest works in visual attention detection by utilizing the contrasts in color, intensity and orientation of images. Han *et al.* [34] formulated the attended object detection using the Markov random field with the use of visual attention and object growing. Ma and Zhang [62] incorporated a fuzzy growing technique in the saliency model for detecting different levels of attention. Lu *et al.* [60] used the low-level features, including color, texture and motion, as well as cognitive features, such as skin color and faces, in their attention model. Different types of images have also been exploited. Ouerhani and Hugli [75] has proposed an attention model for range images using the depth information.

Besides the heavy investigation using the stimuli-driven approach, some methods utilize the prior knowledge on what the user is looking for. Milanese *et al.* [65] constructed the saliency map based on both low-level feature maps and object detection outputs. Oliva *et al.* [74] analyzed the global distributions of low-level features to detect the potential locations of target objects. A few researchers have extended the spatial attention to video sequences where motion plays an important role. Cheng *et al.* [15] has incorporated the motion information in the attention model. The motion attention model analyzes the magnitudes of image pixel motion in horizontal and vertical directions. Bioman and Irani [10] have proposed a spatiotemporal irregularity detection in videos. In this work, instead of using read motion

information, textures of 2D and 3D video patches are compared with the training database to detect the abnormal actions present in the video. Meur *et al.* [64] proposed a spatiotemporal model for visual attention detection. Affine parameters were analyzed to produce the motion saliency map.

Visual attention modelling has been applied in many fields. Baccon *et al.* [8] has proposed an attention detection technique to select spatially relevant visual information to control the orientation of a mobile robot. Driscoll *et al.* [21] has built a pyramidal artificial neural network to control the fixation point of a camera head by computing the 2D saliency map of the environment. Chen *et al.* [13] has applied the visual attention detection technique in devices with small displays. Interesting regions with high saliency values have higher priority to be displayed comparing to the rest of the image. Attention models were used in image compression tasks by Ouerhani *et al.* [76] and Stentiford [93], where regions with higher attention values were compressed with higher reconstruction quality. Peters and Sullivan [79] has applied visual attention in computer graphics to generate the gaze direction of virtual humans.

CHAPTER 3

TEMPORAL VIDEO SCENE SEGMENTATION

In this chapter, we present a general framework for the temporal video segmentation by using the Markov chain Monte Carlo (MCMC) technique. Many of the previously developed methods are based on fixed global thresholds, which are not desirable in many cases. Moreover, due to the fixed thresholds, these methods are likely to generate either over-segmentation or under-segmentation. Further, these methods may use some special knowledge about a particular domain, which may not be appropriate for other domains. For example, there is no obvious video structure in home videos. Hence, it is not easy to generalize these methods to other domains. In contrast, we do not use any fixed threshold or utilize any structure information of the video. Instead, we have developed an iterative method to evaluate the segmentation parameters, including the number of the scene segments and their corresponding locations. In our formulation, if the number of the segments changes, the dimension of the vector containing the boundary locations also changes. The solution space for these two parameters is too complex for direct analytical computation. Therefore, these two parameters are estimated in a statistical fashion using the MCMC technique.

The rest of this chapter is organized as follows: Section 3.1 describes the MCMC algorithm and presents the computations of the transition probabilities and the posterior probability. Sections 3.2.1 and 3.2.2 deal with the applications of the general framework on the segmentations of the home videos and the feature films, respectively. Section 3.3 presents the discussions of the proposed work on other video domains. Finally, Section 3.4 provides the conclusion and discussions of the proposed framework.

3.1 Proposed Framework

By the problem definition, given shots in the video, scene segmentation of the video is a process of grouping the related shots into clusters. In each scene, the shots are related to each other in terms of the corresponding *central concept*. The *central concepts* are different in various contexts. For instance, in home videos, the *central concept* sometimes refers to the same physical environmental setting, e.g., shots related to the same historical monument, or sometimes it refers to the same event, e.g., shots related to a birthday party or a wedding ceremony. In news programs, the *central concept* refers to a specific story topic, e.g., shots related to a political reporting, a weather forecast or a sports reporting. In the feature films, *central concept* refers to the same sub-themes of the story line, e.g., shots related to an action scene or a suspense scene. Different scenes are distinguished by their differences with respect to the *central concept*, and the scene boundaries are the locations where the intrinsic properties of the *central concept* change.

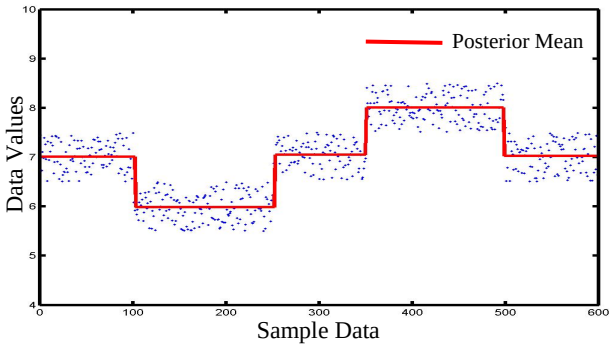


Figure 3.1: An example of the change-point problem. There are five segments containing over 600 observations that are generated by the uniform distributions with different parameters. The red plot is the posterior mean of the segments, and the locations of the steps are the change-points in the data, i.e., the places where the mean changes.

Based on this, we have developed a statistical solution for the two model parameters, the number of scenes and their corresponding boundary locations. The boundary locations are considered as the change-points of the *central concept*, and the problem is formulated as a change-point problem. In a typical change-point problem, the random process has different controlling parameters over time. The goal is to find the points where these parameters change. A simple example of a change-point problem is shown in Figure 3.1. In this example, 600 observations are generated from five different uniform distributions. The change-points are the locations where the distribution mean changes (the steps in the plot). In our application of the temporal scene segmentation, the controlling parameters become the *central concept*, and the steps in the posterior mean plot become the scene boundaries in the video. To estimate the boundary locations, the MCMC technique is used. In the iterative process of MCMC, the posterior probability of the model parameters is computed based on the model priors and the data likelihood of the video. The samples are collected

based on the ratio tests involving the posterior probabilities and the transition probabilities. In the rest of this section, we first introduce the overall MCMC algorithm. Then we present a detailed description of the different types of update proposals. Finally, we describe the computation of the posterior probability.

3.1.1 General MCMC Algorithm

We use a hierarchical Bayesian model in the Markov chain Monte Carlo process. We assume that the model set $\{M_k, k \in \Phi\}$ is a countable set, where k is the number of detected scenes, and $\Phi = \{1, 2, \cdots\}$ is a set of all the possible scene numbers. Model M_k has a parameter vector θ_k, which contains the $k-1$ scene boundary locations (Note: since the first scene always takes the first shot as its starting boundary, it is ignored in our estimation process). Let y denote the video features selected for the data likelihood computation. Based on the Bayes rule, the posterior probability of the parameter k and θ_k given y is:

$$p(k, \theta_k|y) \propto p(y|k, \theta_k)p(\theta_k|k)p(k), \tag{3.1}$$

where $p(k)$ is the prior probability for the number of scenes, $p(\theta_k|k)$ is the conditional prior for the boundary locations θ_k given k, and $p(y|k, \theta_k)$ is the likelihood of the data given the parameters k and θ_k. Since the boundary vector, θ_k, implicitly determines k, the above equation can be further simplified as,

$$p(k, \theta_k|y) \propto p(y|\theta_k)p(\theta_k|k)p(k). \tag{3.2}$$

In the rest of this paper, we use the shorter term $\pi(x) = p(k, \theta_k|y)$ to denote this target posterior, with $x = \{k, \theta_k\}$ considered as a combined parameter vector of k and θ_k.

The general Metropolis-Hasting-Green algorithm [29] is well suited for our task, where the dimension of the parameter vector, x, may change during the updates. It is described as follows:

- Initialize the model parameter x_0.
- At each iteration i, perform the following actions:
  1. Generate Th_α from $Uni[0, 1]$.
  2. Create a new parameter x'_{i-1} from some trial distribution based only on x_{i-1} with a proposal transition (diffusion or jump).
  3. Calculate the ratio $\alpha(x_{i-1}, x'_{i-1})$ as,
$$\alpha(x_{i-1}, x'_{i-1}) = min\left\{1, \frac{\pi(x'_{i-1})q(x'_{i-1}, x_{i-1})}{\pi(x_{i-1})q(x_{i-1}, x'_{i-1})}\right\}. \tag{3.3}$$
  4. Update $x_i = x'_{i-1}$, if $\alpha > Th_\alpha$. Otherwise, set $x_i = x_{i-1}$.

In this algorithm, $q(x, x')$ is the transition probability from x to x'. The transition probability from one state to another depends on the type of the updates. It should satisfy the

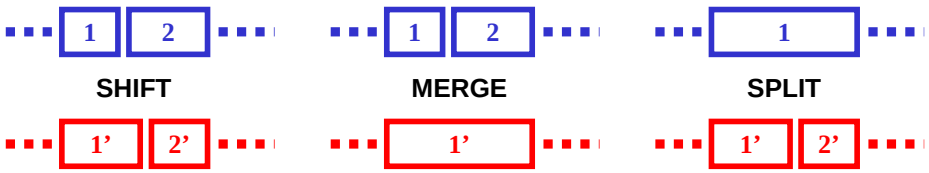


Figure 3.2: Graphical representation of three types of updates. The top row shows the scenes before updates, and the bottom row shows the update results.

reversibility property. Therefore, the proposed updates should also be reversible to ensure this property.

Before going into a detailed description of the updating process, we first present the notations for the variables. Let k be the current number of detected scenes, T be the total number of shots in the video, S_m be the m-th scene with shots $\{s_m^1, s_m^2, \cdots, s_m^{n_m}\}$, where n_m is the number of shots in scene S_m, S'_m be the m-th scene after update, $\mathbb{L}(y|\theta_k)$ be the data likelihood of the entire video, $\mathbb{L}(y_m|f_m)$ be the likelihood of scene S_m given the corresponding features f_m. Finally, k_{max} is the maximum number of the scenes allowed.

The proposed updates contain two parts, diffusion and jumps. Diffusion is defined as the update without changing the structure of the parameter vector x. It traverses within the same sub-space. On the other hand, jumps change the structure and traverse across different sub-spaces. In our case, the diffusion is the shifting of the boundaries between the adjacent scenes. There are two types of jumps: the merging of two adjacent scenes and the splitting of an existing scene. Figure 3.2 shows the graphical representations of the updates. In many applications ([33, 29, 98]), two more updates were proposed: diffusion on the segment model parameter(s) and the change of the segment models. The segment model parameters are the ones that control the generation of the sample data, e.g., posterior means

in Figure 3.1. In our application of the video scene segmentation, based on the underlying assumption that each segment is coherent to its *central concept*, there is often only one scene model for a single video domain. Thus, changing between models is not needed in this case. Furthermore, in some cases like home videos, the data size (number of shots in our case) is small. The maximum likelihood estimator is adequately effective for computing the parameter(s). Therefore, the model parameter diffusion steps can also be dropped.

Let η_k, b_k and d_k denote the probabilities of choosing shifting, merging and splitting, respectively. They satisfy $\eta_k + b_k + d_k = 1$. Naturally, η_1=b_1=0 and $d_{k_{max}}$=0. We use a computation similar to the one proposed in [29], where $b_{k+1} = c \cdot min\{1, p(k)/p(k+1)\}$ and $d_k = c \cdot min\{1, p(k+1)/p(k)\}$, with constant c such that $b_k + d_k \leq C$, $\forall k = 1, \cdots, k_{max}$. This results in $b_{k+1}p(k+1) = d_k p(k)$.

3.1.2 Stochastic Diffusions

The diffusions involve the shifts of the scene boundaries between adjacent video scenes. The update is carried out as follows:

- A number m is randomly drawn from the discrete uniform distribution $[1, k-1]$, such that the boundary between S_m and S_{m+1} is updated.

- The new boundary s^t is drawn from a 1D normal distribution with the mean at the original boundary s^1_{m+1} in the range of $[s^1_m,\ s^{n_{m+1}}_{m+1}]$. The updated scene S'_m contains shots of $\{s^1_m, \cdots, s^{t-1}\}$, and the updated scene S'_{m+1} contains $\{s^t, \cdots, s^{n_{m+1}}_{m+1}\}$.

Assume the number of the current scenes is k, and the current parameter vector is $x = \{k, \theta_k\}$. Then the probability for selecting scene S_m is $1/(k-1)$. Since the potential shift is drawn from a normal distribution around the original scene boundary $\hat{t}$, this drawing probability for the new boundary t is computed as,

$$p(t) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{\Delta t^2}{2\sigma^2}) \Big(I_{[s^1_m, s^{n_{m+1}}_{m+1}]}(t) \Big), \tag{3.4}$$

where $\Delta t = t - \hat{t}$, and σ is the standard deviation of the movement (in our experiment, $\sigma = 2$). The indicator function $I(t)$ controls the shift, such that the new boundary is within the correct range. The normal distribution is assumed since the new boundary is not expected to deviate from the old boundary too far. In summary, the forward transition probability for the shift update is $q(x, x') = (\frac{1}{k-1})p(t)$.

During this entire update, the total number of scenes, k, is not changed, and the new boundary remains in the original range $[s^1_m,\ s^{n_{m+1}}_{m+1}]$. The reverse transition is the process of shifting from the new boundary t back to the original boundary $\hat{t}$. Thus, the relationship between $q(x, x')$ and its reverse version $q(x', x)$ is equal due to the symmetrical property of the normal distribution.

3.1.3 Reversible Jumps: Merge and Split

For the jump updates, the transition during a merge is related to the transition of a split, since merge and split are a pair of reversed updates. Let us consider the splits first. The number of scenes is increased by 1 by splitting a scene $S_m = \{s_m^1, \cdots, s_m^{n_m}\}$ into two new scenes $S'_m = \{s_m^1, \cdots, t-1\}$ and $S'_{m+1} = \{t, \cdots, s_m^{n_m}\}$, where t is the new boundary. The process contains two portions: selecting a scene S_m and selecting a new boundary between its old boundaries. The selection of the new boundary in the split process can be performed assuming the uniform distributions [29]. However, to achieve better performance, the data-driven technique is often used ([33] and [98]) to propose the jump transitions. We assume the uniform probability for selecting scene S_m. The new boundary t is chosen, such that it provides the maximum likelihoods for the two new scenes,

$$t = arg\ max\Big(\mathbb{L}(S'_m|f'_m) + \mathbb{L}(S'_{m+1}|f'_{m+1})\Big), \tag{3.5}$$

where $\mathbb{L}(S'_m|f'_m)$ and $\mathbb{L}(S'_{m+1}|f'_{m+1})$ are the likelihoods of the new scenes S'_m and S'_{m+1}, given their corresponding features. If we consider that video scenes are independent events in the time series, the proposal probability for a split can be expressed in the following form,

$$q(x, x') = \frac{1}{k}\mathbb{L}(S'_m|f'_m)\mathbb{L}(S'_{m+1}|f'_{m+1}). \tag{3.6}$$

The reversed update of the split is the merging of two scenes into one. The construction of the proposal probability for the merge can be carried out similarly to the one for the split.

Again, we assume the uniform distribution for selecting scene S_m, such that scenes S_m and S_{m+1} are merged into S'_m. The proposal probability for the merge transition is constructed as follows,

$$q(x, x') = \frac{1}{k-1}\mathbb{L}(S'_m|f'_m). \tag{3.7}$$

3.1.4 Posterior Probability

Since Poisson distribution models the number of incidents happening in a unit time interval, we assume the number of scenes, k, is drawn from such a distribution with mean λ. The model prior on k is computed as

$$p(k) = e^{-\lambda}\frac{\lambda^k}{k!} \cdot I_{[1,k_{max}]}(k), \tag{3.8}$$

where $I_{[1,k_{max}]}(k)$ is an indicator function. $I_k = 1$, if $1 \leq k \leq k_{max}$; $I_k = 0$ otherwise. A plot of the prior distribution is shown in Figure 3.3.

If there are k segments (scenes) in the video, then there are $k-1$ scene boundaries, since the boundary for the first scene is always the beginning of the video. The probability of $p(\theta_k|k)$ is the same as the probability of selecting a subset with size $k-1$ from the remaining $T-1$ shots. Therefore, the conditional prior can be defined in terms of the combinations,

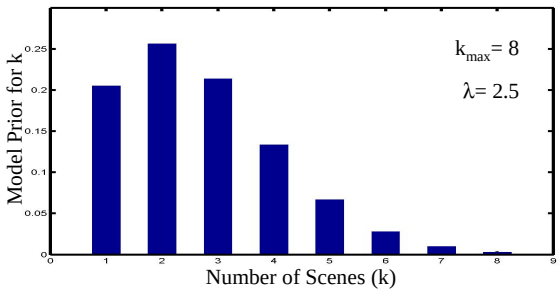


Figure 3.3: Prior distribution (Poisson) of the model parameter k, the number of scenes in the video. The mean of the distribution, λ, is pre-assigned as 2.5, and k_{max} is 8.

$$p(\theta_k|k) = \frac{1}{C_{k-1}^{T-1}} = \frac{(k-1)!(T-k)!}{(T-1)!}. \tag{3.9}$$

The last term to be computed is the likelihood. Let $\mathbb{L}(y|\theta_k) = p(y|\theta_k)$ denote the global likelihood of the video data y given the parameter vector θ_k. As discussed in Section 3.1, each scene possesses a different *central concept*. It is meaningful to make an assumption that each scene is independently recorded from others. Therefore, the overall likelihood can be expressed as,

$$\mathbb{L}(y|\theta_k) = \left(\prod_{m=1}^{L} \mathbb{L}(y_m|f_m) \right)^{\frac{1}{L}}, \tag{3.10}$$

where $\mathbb{L}(y_m|f_m)$ is the individual likelihood of data y_m in scene S_m, based on the feature values f_m. The geometric mean of the individual likelihoods is considered for the normalization purpose. In order to make the ratio test meaningful, the likelihood should be scaled to the same level during each iteration. The definition of the *central concept* is different across domains. Therefore, the features selected to compute the likelihoods are different for the

different types of videos. Here, $\mathbb{L}(y|\theta_k)$ is a general representation of the likelihood rather than a specific computation.

The target posterior probability is proportional to the product of the model prior $p(k)$, the conditional prior $p(\theta_k|k)$, and the data likelihood $\mathbb{L}(y|\theta_k)$,

$$\pi(x) \propto \mathbb{L}(y|\theta_k)p(\theta_k|k)p(k). \tag{3.11}$$

To determine whether the proposed update in the parameter space is accepted or rejected, we compute the ratio of the two terms: $\pi(x')q(x', x)$ and $\pi(x)q(x, x')$. If the ratio, $\alpha(x, x')$, satisfies the stochastically generated threshold, the proposed update is accepted; otherwise, the model parameters are kept the same as in the previous iteration.

3.2 Applications and Discussions

In this section, we demonstrate the proposed scene segmentation method on two video domains. If we examine the generation process of the videos, we can classify them into two categories:

- *Produced Videos*: This group contains feature films, television news programs and other TV talk or game shows. They are initially recorded in raw format and are later modified to produce the carefully organized video programs with accordance to the certain video production rules.

- *Raw Videos*: Compared to the previous group, this category involves little post-modifications and contains videos that are mostly in the form in which they were originally recorded. Common domains in this category are home, surveillance and meeting videos.

Due to the large variety of video domains, we have selected two representative domains to demonstrate the effectiveness and the generality of the proposed method, with one domain from each of the categories described above. The home video domain is chosen as the representative domain of the *Raw Video* category, and the feature film domain is selected for the *Produced Videos* category. In this paper, we assume the video shots are available. In the experiment, we used a multi-resolution method provided in [118] to detect and classify the video shot boundaries in both home videos and feature films.

3.2.1 Home Videos

Home video is a broad term that refers to videos composed with a "free style", e.g., family videos, tour videos, wedding tapes or ground reconnaissance videos (GRV). They are recorded from handhold cameras, spy cameras, cameras mounted on ground vehicles, etc., and come in different forms. Some are in high resolution, while others have been shot at lower quality. Some have a full field of view, and some may be recorded by cameras hidden in bags (GRV), so part of their field of view is blocked by the carrier. Some example key frames are shown

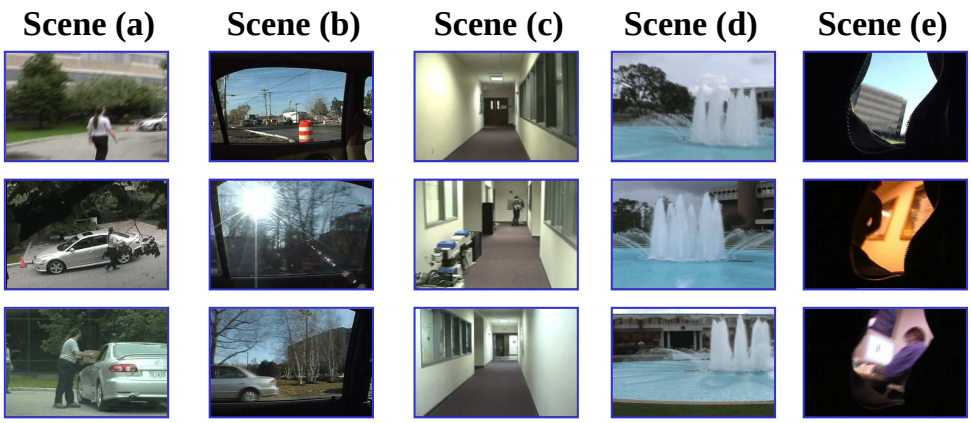


Figure 3.4: Five example home video scenes with their key frames. Some of them are indoors (c); some are outdoors (a,b,d,e). Scenes (a,b) were taken by cameras mounted on ground vehicles, (e) was taken by a spy camera in a bag, and (c,d) were taken by handhold cameras.

in Figure 3.4. Temporal scene segmentation of home videos provides the logical units related to interesting locations or events, and the output segments can be used for the further analysis and processing of the videos, e.g., indexing, storage, retrieval of the video and action recognition. Since there is no grammar involved in the production process of the home videos, temporal segmentation emphasizes the analysis of the features derived from the video than on the video structure. As mentioned at the beginning of this chapter, this type of analysis could be threshold based, zero-crossing based, etc., with or without the training of the features. Home videos are not as well controlled as other domains like TV programs. The scene boundaries sometimes are clearly identifiable (strong boundaries), but many times they are difficult to determine using the same criteria as strong boundary detection. Due to this uncertainty in the home videos, it is likely to result in either under-segmentation or over-segmentation using any fixed threshold, and it is not practical to train the system for the threshold selection. On the other hand, the proposed approach finds the

boundary locations by detecting the local peaks in the likelihood plot of the video shots, and therefore, avoids the previously mentioned problems.

3.2.1.1 Feature Selection

In the context of temporal scene segmentation, a variety of features have been exploited. The commonly used features include color, motion content, shot length, etc. Since home videos are taken in a "free style", the patterns for motion content and shot length are not distinctive across different scenes. Usually the shots in the same temporal scene are coherent with respect to the same environment; there are visual similarities that exist among these shots. On the other hand, the shots from different scenes should be visually distinctive. Therefore, we have focused our efforts on the analysis of the color information in the shots. We use the histograms to represent the color information in the video frames. The color histogram for each frame is the 3-dimensional histogram in the RGB space with 8 bins in each dimension. Let h_i be the histogram for frame f_i. Furthermore, we define the histogram intersection between frames f_i and f_j as,

$$HistInter(f_i, f_j) = \sum_{b \in Allbins} min(h_i^b,\ h_j^b), \tag{3.12}$$

where b is the individual bin in the histogram.

Instead of using all the frames in the shot, we extract the key frames as the representation of the shot, and further analysis is performed based on the key frames only. It is common to select a single key frame for each shot. However, for shots with long durations and with high activity content, multiple key frames provide a better representation. Several key frame selection approaches have been proposed in the past few years ([31, 35, 84, 122]). In this paper, we use the method proposed in [84]. Assume there are a total of n frames in shot s, the procedure for selecting the key frames is described as follows:

- Include the middle frame into the key frame set $\mathbb{K}_s$ as the first key frame κ_s^1;
- For $i = 1 : n$, do

  If $max(HistInter(f_i, \kappa_s^j)) < Th$, $\forall \kappa_s^j \in \mathbb{K}_s$

  Include f_i into $\mathbb{K}_s$ as a new key frame.

In this algorithm, Th is the threshold for selecting a new key frame, and we use the histograms of the key frames as their representation.

3.2.1.2 Likelihood Computation

We define the visual similarity between two shots in terms of the Bhattacharya distance, which is the distance between two histograms h_1 and h_2, defined as $d_B(h_1, h_2) = -ln\Big(\sum_{b \in allbins} \sqrt{h_1^b h_2^b}\Big)$. The visual similarity between shots s_i and s_j is as follows:

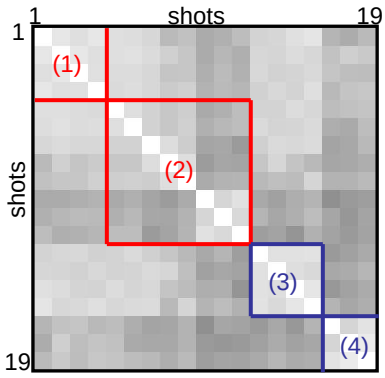


Figure 3.5: Visual similarity map of the shots in a testing video. Brighter cells represent higher similarity. The shots in the same scene possess higher similarity compared across scenes. The bright blocks on the diagonal gives the idea of temporal scenes. The figure shows the intermediate results for one iteration, where the red scenes (1 and 2) are not matched with the correct boundaries, and the blue scenes (3 and 4) show the correct detections. A short sequence of updates demonstrated on the similarity map is shown in Figure 3.8.

$$Sim(s_i, s_j) = max(\mathbb{C} - d_B(\kappa^m_{s_i}, \kappa^n_{s_j})), \tag{3.13}$$

where $\kappa^m_{s_i} \in \mathbb{K}_{s_i}$, $\kappa^n_{s_j} \in \mathbb{K}_{s_j}$, and $\mathbb{C}$ is a constant. After computing the visual similarity between all pairs of shots in the video, a similarity map is generated. One such map is shown in Figure 3.5. In this map, the brighter cell represents higher similarity value. The shots that are in the same temporal scene form a bright block along the diagonal in the similarity map. If the shots $[s_a, \cdots, s_b]$ are clustered into scene S_m, the likelihood for this scene is computed as:

$$\mathbb{L}(y_m|f_m) = avg\Big(\mathbb{M}(a:b, a:b)\Big), \tag{3.14}$$

which is the average similarity value of the sub-block in the similarity map $\mathbb{M}$ starting from row a to row b. It is intuitive that the correct segmentation of the video gives the diagonal blocks to reach the maximum likelihood. To compute the overall likelihood, substitute Eqn. 3.14 into Eqn. 3.10. Up to this point, the overall likelihood $\mathbb{L}(y|\theta_k)$, the conditional prior $p(\theta_k|k)$ and the model prior $p(k)$ are determined. Therefore, acceptance for proposal updates is decided by the ratio test described in the MCMC algorithm.

3.2.1.3 System Performance

The proposed method has been tested on four home videos with 23 total scenes. These scenes were recorded with various environmental settings. Each scene is composed of multiple video shots. Some of them are indoor scenes (Scenes (c,e) in Figure 3.4), while others are out-door scenes (Scenes (a,b,d) in Figure 3.4). Furthermore, the videos were taken in different styles. Some scenes were recorded from handhold cameras (Scenes (a,c,d) in Figure 3.4), some were recorded by a spy camera hidden in bag (Scene (e) in Figure 3.4), and others were recorded by a camera mounted on a ground vehicle (Scene (b) in Figure 3.4).

It is well known that samples generated from a single Markov chain may not give an accurate solution. Rather, the solution generated from a single chain may be in the neighborhood of the true solution. To overcome this problem, we independently execute multiple Markov chains. The results from each individual chain provide the votes for the shots that have been declared as scene boundaries. After certain runs, the shots with the locally highest

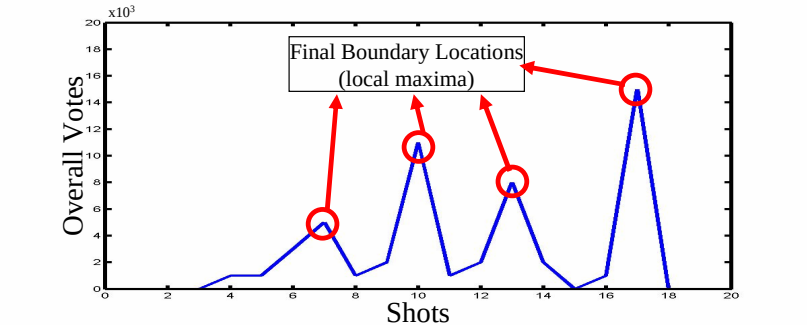


Figure 3.6: The overall votes of the shots declared as scene boundaries from multiple independent Markov chains. The red circles represent the shots that are declared as the final scene boundary locations, which correspond to the local maxima in the overall vote plot.

votes represent the final scene boundaries. Figure 3.6 shows the overall votes of the scene shots being declared as scene boundaries from all runs, and the red circles represent the local maxima, which correspond to the true boundaries. Even though one single chain may not provide the correct result, there is an issue of the posterior probability reaching the "confidence" level. This is referred as the "burn-in" period. As shown in Figure 3.7, after certain iterations, the posterior probability reaches a level and stays there with only minor fluctuations. For this particular testing video, the "burn-in" time is short, due to the small size of the data (number of shots). A simplified version of the iteration process is shown in Figure 3.8.

The matches between the ground truth data and the segmented scenes are based on the matching of their starting boundaries. For a given home video with n scenes, let $\{t_1, t_2, ..., t_n\}$ denote the starting shots of the reference scenes and $\{s_1, s_2, ..., s_k\}$ denote the starting shots of the detected scenes. Scene t_i is declared as matched if one of the detected scenes s_j has the same starting shot. Figure 3.9 shows a graphical representation of the video matching.

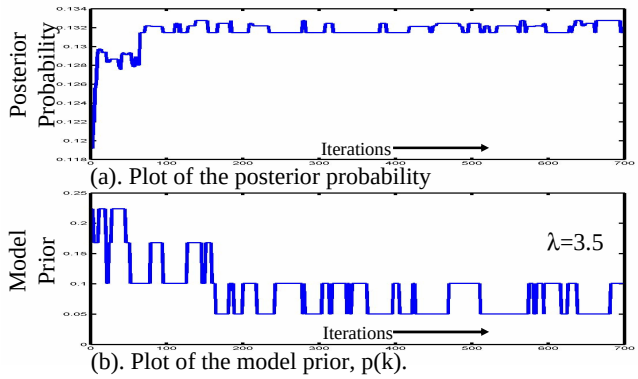


Figure 3.7: (a). The plot of the posterior probability of the parameter estimation during a single Markov chain (run). As demonstrated in the figure, after certain iterations, the posterior reaches a "confidence" level and stays there with minor fluctuations. It should be noted that if the data size (number of shots in our application) is small, the process reaches this level quickly. (b). The plot of the model prior for the number of scenes, k, where the model mean, λ, is set at 3.5. The horizontal axis in both plots represents the number of iterations. At the end of the process, plot (a) gives the posterior probability of the parameters given the video data, and plot (b) gives the information on the number of scenes, k.

In these videos, shots in each scene are coherent with respect to the same environmental settings. For instance, there are five scenes in video 2. The first scene is an indoor scene, which shows the interior of a building. The next scene shows the exterior of the same building. The third scene is a sequence around a fountain. Finally, the last two scenes shows the exterior and the interior of the same building again. It is evident that the shots within the same scene are visually similar, while shots in different scenes are visually distinctive.

Two accuracy measures are used to measure the system performance: precision and recall,

$$Precision = \frac{X}{A}, \qquad Recall = \frac{X}{B}, \tag{3.15}$$

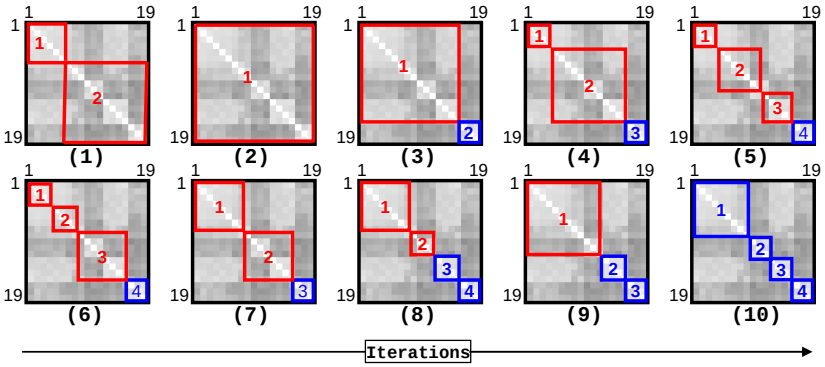


Figure 3.8: Demonstration of a simplified MCMC iteration process. We show ten updates during a single run. The red boxes represent the detected scenes that do not match the true boundaries, while the blue boxes show the detected scenes that do match the ground truth. The sample video contains 19 shots, which are initially split into two arbitrary scenes (1). After a series of updates, including shift (6), merge (2,7,9) and split (3,4,5,8,10), the final detected scenes (10) match the true boundary locations. As illustrated in the figure, the scenes are eventually "locked" with the bright diagonal blocks in the similarity map.

where X is the number of correct matches between the system detections and the ground truth scenes; A is the total number of system detections; B is the total number of ground truth references. The detailed precision/recall measures are shown in Table 3.1. If the matches in all the videos are treated as equally important, the overall precision and recall are 0.840 and 0.913, respectively.

To further demonstrate the effectiveness of the proposed method, we also compare our system output with the results generated by one of the previously developed methods. As the most relevant technique to our scenario, we choose the *Backward Shot Coherence (BSC)* approach proposed in [84]. The BSC approach is a two-pass algorithm, which first segments the video into initial scenes using the color consistency and then merges them based on the similarity between their motion contents. In home videos, the same recorder often

Table 3.1: Accuracy measures of four home videos. Insertion is the number of over-segmentation (false positives), and deletion is the number of the mis-detections (false negatives).

Measures	***clip1***	***clip2***	***clip3***	***clip4***
Length	12:42	06:53	07:31	17:53
Num. of Shot	47	16	19	25
Num. of Scenes	8	5	5	5
Detected Scenes	8	5	5	7
Match	7	5	5	4
Insertion	1	0	0	3
Deletion	1	0	0	1
Precision	0.875	1.000	1.000	0.571
Recall	0.875	1.000	1.000	0.800

Table 3.2: Comparison between the proposed Markov chain Monte Carlo (MCMC) method and the *Backward Shot Coherence (BSC)* [84]. The overall precision and recall are computed as if every scene in all videos were equally important. The last column shows the number of the reference scenes in each clip.

Measures	***MCMC***	***BSC***	***Reference***
Clip 1 Detection	8	7	8
Clip 1 Match	7	4	-
Clip 2 Detection	5	4	5
Clip 2 Match	5	4	-
Clip 3 Detection	5	6	5
Clip 3 Match	5	4	-
Clip 4 Detection	7	7	5
Clip 4 Match	4	4	-
Total Detection	25	24	-
Total Match	21	16	-
Total Insertion	4	8	-
Total Deletion	2	7	-
Overall Precision	0.840	0.667	-
Overall Recall	0.913	0.696	-

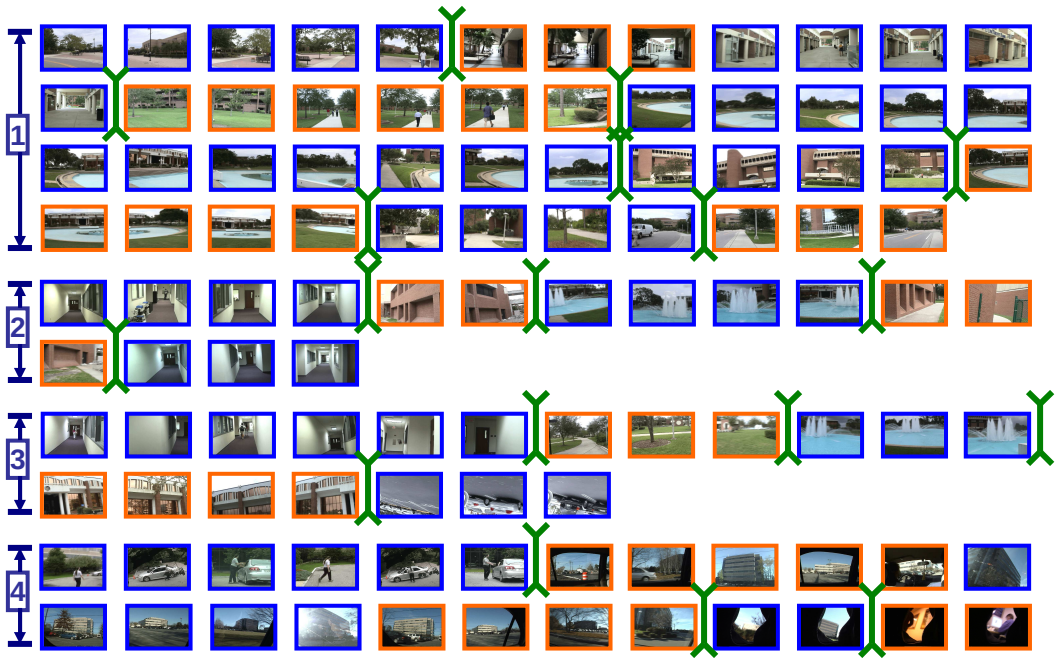


Figure 3.9: Matches in the testing home video clips. The figure shows the key frames of the videos. In each video, the detected scenes are labelled by alternating blue and orange groups of shots, and the true boundary locations are shown by the deep green separators.

exhibits similar motion of the camera. Furthermore, unlike other domains, motion content in home videos is less meaningful and not distinctive across scenes. Based on the experimental observations, results obtained using both passes in the BSC algorithm are the same as the results obtained using only its first pass, which generates the scene segments using color information. Since only the visual information is useful in our application, we compare the system performance between the results generated by the proposed MCMC method and the BSC method for the sake of fairness. The comparison results are shown in Table 3.2.

3.2.2 Feature Films

To demonstrate the generality of the proposed framework, we have also tested the proposed system on three feature films: *Gone in 60 Seconds*, *Dr. No* (James Bond) and *The Mummy Returns*.

3.2.2.1 Feature Selection

Based on the definition provided by the Webster dictionary [100], a movie scene is one of the subdivisions of a play, or it presents continuous actions in one place. Movie scenes are composed according to the *film grammar*, which is a set of rules about how the movies are produced. In a scene, the shots often exhibit similar patterns, which can be reflected by low-level features. For example, in action scenes, shots are generally short in length, and the visual content, which indicates the activity level of the scene, changes rapidly. On the other hand, in drama scenes, the shots are much longer, and the visual content is relatively consistent. For feature films, we use these two features computed from the movies, shot length and visual content, to group the semantically coherent shots into scenes. Let l_s denote the length of shot s and v_s be the visual content in that shot. The shot length represents the pace of the movie, and the visual content shows how much is going on in the shot. The visual content is defined as,

$$v_s = \frac{1}{N_s}\sum_{i=1}^{N_s}(1 - HistInter(f_i, f_{i+1})), \tag{3.16}$$

where $HistInter(f_i, f_{i+1})$ is the color histogram intersection between the i-th and $(i+1)$-th frames, and N_s is the number of frames in shot s. The plots of the shot length and the visual content are shown in Figure 3.10. These two features are used in the construction of the data likelihood.

3.2.2.2 Likelihood Computation

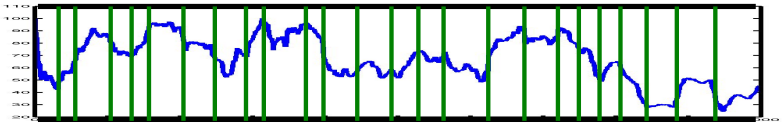
In film production, the patterns for different features are related to each other. For instance, in action scenes, the short shots are accompanied by a high degree of visual content. Therefore, the features l_s and v_s should not be considered independently of each other. We use a two-dimensional normal distribution to model the features in a scene S_m,

$$N(g_s, m) = \frac{1}{\sqrt{2\pi S}} exp\Big(- \frac{(g_s - \hat{g_m})^T G^{-1} (g_s - \hat{g_m})}{2}\Big), \tag{3.17}$$

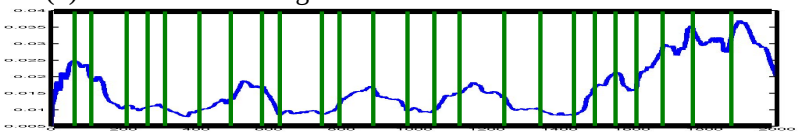
where g_s is the feature vector $[l_s\ v_s]^T$. The vector $\hat{g_m}$ is computed as the sample means for the entire scene S_m, and G is the covariance matrix with determinant S. Again, by considering shots as recorded independently, the likelihood in each scene S_m is,

$$\mathbb{L}(y_m|f_m) = \Big(\prod_{s=1}^{n_m} N(g_s, m)\Big)^{\frac{1}{n_m}}. \tag{3.18}$$

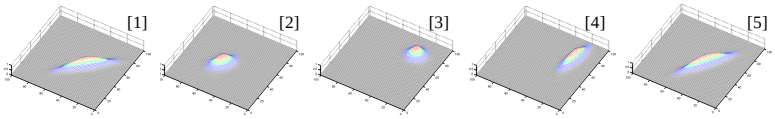
(a) Representative frames of example scenes in the movie



(b) Plot of the shot length in the movie.



(c) Plot of the visual content in the movie.



(d) PDFs of the 2D normal distributions of first five scenes.

Figure 3.10: (a). Representative frames of some example scenes in the movie *Gone In 60 Seconds*; (b). Plot of the shot length variable; (c). Plot of the visual disturbance feature. Usually, shots with shorter length are accompanied by a high level of visual disturbance. The green bars represent the scene boundaries in the movie, which were detected by the proposed method; (d). PDF plots on the 2D normal distribution of the first five scenes in the movie. The distribution parameters, mean and covariance, are different across the scenes.

We substitute Eqn. 3.18 in Eqn. 3.10, and perform the ratio test for the acceptance decisions. A similar argument is applied here for taking the geometric mean as in Eqn. 3.10.

3.2.2.3 System Performance

We have experimented our approach on three feature-length films: *Gone in 60 Seconds*, *Dr. No* and *The Mummy Returns*. Each movie contains thousands of shots. The matching follows similar procedure as used in Section 3.2.1.3. However, the matching technique is

Table 3.3: Accuracy measures for three movies: *Gone in 60 Seconds*, *Dr. No*, and *The Mummy Returns*

Measures	***Gone in 60 Seconds***	***Dr. No***	***The Mummy Returns***
Length	01:46:09	01:30:55	01:45:33
Num. of Frames	152665	130811	151802
Num. of Shot	2237	677	1600
Num. of Scenes	29	17	18
Detected Scenes	25	20	18
Match	24	14	15
Insertion	1	3	3
Deletion	5	6	3
Precision	0.960	0.700	0.833
Recall	0.828	0.824	0.833

slightly different. In movies, usually there is not a concrete or clear boundary between two adjacent scenes due to editing effects. Movie chapters are sometime segued with a smooth transition. Therefore, matching based on boundaries is not meaningful and often returns incorrect results. Instead, we use a "recovery" method. Suppose there are a set of the reference scenes $\{T_1, T_2, ..., T_n\}$ and a set of the detected scenes $\{S_1, S_2, ..., S_k\}$. A reference scene T_m is said to be "recovered", if a majority of this scene ($> 50\%$) overlaps one of the detected scenes. The "recovery" is a one-to-one correspondence, i.e., one reference scene can only be matched with at most one detected scene, and one detected scene can cover at most one reference scene. The scene matching for the movie *The Mummy Returns* is shown in Figure 3.11. In this example, we consider the chapters provided by the DVD as the ground truth scenes. The key frames of both the ground truth scenes and the detected scenes are presented. Again, we use the precision and recall measures defined in Section 3.2.1.3 for the

Figure 3.11: Matching of scenes for the movie *The Mummy Returns*. It shows the key frames of the ground truth scenes that are obtained from the DVD chapters and the key frames of the detected scenes. The key frames of the ground truth scenes are accompanied by their titles. The matches scenes are shown with their key frames aligned. Pairs with blank spaces are the mis-matches, i.e., insertions and deletions.

performance evaluation. Detailed results for movie scene segmentation are shown in Table 3.3.

3.3 Discussions

The idea of the *central concept* is also applicable to other video domains. For example, in television talk shows, one major distinction between the commercials and the real TV talk

shows is that in the show itself there is often a repeating pattern between the host and the guest, which the commercials do not possess. The feature to distinguish this *central concept* involves the number of the repeating shots in the segment. Another example is the news video segmentation. In this task, each news segment is composed of the shots that are coherent to a certain news focus. Non-news segments include commercials, lead-in/out, reporter chit-chatting, etc. The textual information, closed captions (CC) and automatic speech recognition (ASR) output can be used as the features for constructing the posterior distribution. In this case, the semantic relations between the key words appearing in the shots can be analyzed. Shots that have the same news focus should possess similar distributions of key words. The MCMC framework can find the places where the distributions of key words change to detect the scene boundaries.

There is another temporal segmentation process on the lower-level video structure, which is commonly known as the shot boundary detection. Shot level segmentation and the scene segmentation have their similarities and differences. A shot is defined as a series of continuous frames with consistent background settings. This assumption naturally leads to the color consistency constraints, and it does not refer to any high level semantic meanings. On the other hand, scene segmentation involves more semantic coherence. For example, in home videos, shots within the same scene are coherent to each other in terms of the same events or the same physical sites. In feature films, shots in the same scene are related to the same sub-theme of the movie story line. In both the cases, the color similarity constraint is insufficient for segmentation. The high-level semantics are often bridged by analyzing the

patterns of other types of low-level features, like video pace and visual content in the films or the narration in the news programs.

3.4 Conclusions

In this chapter, we have presented a general statistical framework for the temporal scene segmentation of videos. We have solved the scene segmentation task by automatically determining the places where the *central concept* changes. A target distribution of the model parameters, including the number of scenes and their corresponding boundary locations, is constructed to model the probabilities of the video shots being declared as the scene boundaries, and the solution is achieved by performing the sampling from this target distribution using the Markov chain Monte Carlo (MCMC) technique. In the iterative process of MCMC, the posterior probability is computed based on the model prior, conditional prior and the data likelihood given the parameters, and updates are determined based on the posterior probabilities and the transition probabilities. We have applied the method to several home videos and three feature films, and we obtained high accuracy measures (Tables 3.1, 3.2 and 3.3).

CHAPTER 4

SEMANTIC LINKING OF VIDEOS

In this chapter, we present a framework for the semantic linking of news topics. Unlike the conventional video content linking methods, which are based only on video shots, the proposed framework links the news video across different sources at the story level. The rest of this chapter is organized as follows: Section 4.1 describes the proposed framework in detail, including the computation of the visual similarity and the textual correlation between stories; Section 4.2 presents the system evaluation results for the tasks of the story linking and news ranking; finally, Section 4.3 concludes our work.

4.1 Proposed Framework

Let us consider how humans link stories on the same topic and distinguish the ones that are different. Given two news stories, our visual system gives us a first impression about the common contents of the two. It could be the same person of interest, the same action/activity, or the same physical site. For example, when the Secretary-General of the United Nations

proposes a peace plan, all the news channels broadcast the same picture containing his face. On the other hand, when reporting on a riot, there may not be any particularly person of interest. Instead, the physical scene is emphasized. In this scenario, global matching between the images is needed. Besides the visual information, we also acquire the language information in the video, which provides the most direct semantic cue in the news videos. Often the form of the language information is speech and/or closed captioning (CC) on the screen. They contain the textual form of the spoken words in the video. The proposed method constructs the semantic linkage between news stories in a similar way using both visual and textual information. It computes the visual similarity based on both the facial and non-facial key frames of the stories, and establishes the textual correlation using automatic speech recognition (ASR) output.

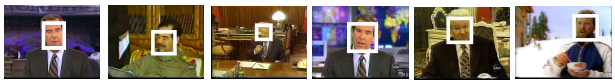
4.1.1 Visual Correlation

The first step in the computation of visual correlation is to detect faces in the key frames of the stories. If a face is detected in a key frame, that key frame is classified as a facial key frame; otherwise, it is classified as a non-facial key frame. Given a story S_i, we have a set of its key frames, which is composed of two disjoint sets, the facial key frames, $K = \{k_{(i,1)}, \cdots, k_{(i,m_i)}\}$, and the non-facial key frames, $\Phi = \{\phi_{(i,1)}, \cdots, \phi_{(i,n_i)}\}$, where m_i and n_i are the numbers of facial and non-facial key frames in S_i, respectively. Computation of the visual similarity between two stories is carried on K and Φ separately. Here, the video

shots containing anchor person(s) are not considered in the visual similarity computation. This is because it does not provide meaningful linkage, since no anchor person works for two stations, and stories broadcasted by the same anchor person do not imply they are similar. We use a graph-based method to remove the anchor shots [119]. The underlying mechanism for the anchor removal technique is to analyze the frequencies of the video shots in the news program. The shots are classified as the general anchor, anchor of special programs and the non-anchor shots, according to their frequencies based on the fact that anchors appear much more often than other non-anchor shots.

4.1.1.1 Facial Key-Frame Matching

Many times, the news networks broadcast events that involve a particular person or a group of persons. In these types of news stories, since the person is performing the action (e.g., a political leader giving speech), or the person constitutes the major part of the event (e.g., meeting of foreign leaders), he/she becomes the focus of the interest. The images often reveal the person's face. In these situations, the best linkage between stories is provided by the correlation of the persons by their facial information. Common face correlation methods are known to have some drawbacks, such as being sensitive to the pose of the face, lighting conditions, and the sizes of the faces. This is because the traditional face correlation methods use the local information of the face patch. To overcome the aforementioned problems, we utilize the global properties related to the detected faces. An extended region, "body", is



(a). Sample key-frames from the testing videos with faces labeled.

(b). The "body" regions extracted from the above detected faces.

Figure 4.1: (a). The sample key frames with the detected faces; (b). The body regions extended from the faces. Global feature comparison or face correlation fails to link the same person in these examples, while the comparison of the "body" regions provides the meaningful information.

used. The procedure for obtaining the "body" region is as follows: first, the face in the key frame is detected by the face detector [99]. The detected face region is then extended to cover the upper body of the corresponding person. The idea behind this is that in the news stories involving the important person, the person usually wears the same clothes. Therefore, this can be taken as the cue for the similarity. All the body regions in story S_i are collected to provide the body set, $B = \{b_{(i,1)}, \cdots, b_{(i,\beta_i)}\}$, where β_i represents the total number of body patches in the story. Note that $\beta_i \geq m_i$, because there might be multiple faces detected in a single key frame. Some of the facial key frames and their body patches are shown in Figure 4.1.

We compute the 3D color histogram, denoted by $h_{(i,j)}$, of each body patch $b_{(i,j)}$. The Bhattacharya distance between two histograms is used in the similarity measure. The similarity between two body patches $b_{(i,j)}$ and $b_{(p,q)}$ is defined as,

$$SimF(b_{(i,j)}, b_{(p,q)}) = e^{-d_B\left(b_{(i,j)}, b_{(p,q)}\right)}, \tag{4.1}$$

where $d_B\big(b_{(i,j)}, b_{(p,q)}\big) = -ln\big(\sum_{r \in allbins} \sqrt{b^r_{(i,j)} b^r_{(p,q)}}\big)$ is the Bhattacharya distance. The visual similarity between two stories S_i and S_p over the facial key frames is computed as follows,

$$\Gamma_F(i,p) = max\Big(SimF(b_{(i,j)}, b_{(p,q)}\Big), \tag{4.2}$$

where $j = \{1, \cdots, \beta_i\}$ and $q = \{1, \cdots, \beta_p\}$. Based on Eqns. 4.1 and 4.2, the range for Γ_F is bounded in $[0, 1]$.

4.1.1.2 Non-Facial Key-Frame Matching

Some stories do not contain human faces. For instance, in a report of a riot, no particular human face can be detected due to various reasons. Another case is the stories with special format, such as the weather forecast and the sport reporting. In these stories, only non-facial key-frames are available. The visual linkage here is defined by the similarity between the non-facial key-frames, which is computed based on the homography between the images. If the key-frames of two news stories are focusing on the same scene, the homography is able to successfully capture the transformation between the key-frames with small residual. Otherwise, the homography would provide high residual.

Homography models the planar transformation between two images. To estimate the homographies across images, interest-points and their correspondences (sparse optical flow) need to established. In our formulation, we use the Scale Invariant Feature Transform

(SIFT [59]) for the feature point detection and matching. Given a pair of corresponding points $\mathbf{x} = [x\ y\ t]$ and $\mathbf{x}' = [x'\ y'\ t']$ in their homogeneous coordinates from image I_1 and I_2, respectively, the homography is expresses as,

$$\begin{bmatrix} x' \\ y' \\ t' \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ t \end{bmatrix}, \tag{4.3}$$

where parameters $\{a_i, i = 1 \cdots 8\}$ capture the transformation between two matching planes, they can be estimated by providing at least four pairs of correspondences. For simplification purpose, the coordinates of $\mathbf{x}$ and $\mathbf{x}'$ are normalized, such that their third elements are 1. The above equation can be written in a shorter form $\mathbf{x}' = A\mathbf{x}$. With noise presented in the images, $\mathbf{x}$ may not match exactly with $\mathbf{x}'$. Rather, it maps to a projected point $\hat{\mathbf{x}}'$ by applying the homography A. Therefore, the goodness of the homography between two images is computed as in terms of the residual between the true matches and the projections,

$$\epsilon(I_1, I_2) = \frac{1}{k} \sum_{\forall(\mathbf{x}_i, \mathbf{x'}_i)} ||\hat{\mathbf{x}'}_i - \mathbf{x'}_i||, \tag{4.4}$$

where k is total number of the correspondences.

Two situations should be considered. Firstly, if two images are totally different, few correspondence would exist. Thus, the detected points in the images will be insufficient for the minimum criteria of the homography computation. Secondly, if there are multiple planes exist in the image, the homography computed using all correspondences would be invalid

(a) Examples of the matching image pairs.

(b). A pair of non-matching images from different scenes.

Figure 4.2: Point matching of images. (a). Two pairs of images, which were taken from the same scenes. The correspondences between feature points are shown. Figure (b) shows a pair of non-matching images from two different scenes.

for the matching purpose. Consider two non-facial key-frames $\phi_{(i,j)}$ and $\phi_{(p,q)}$ from stories S_i and S_p. Assume $\phi_{(i,j)}$ has k_1 points detected and $\phi_{(p,q)}$ has k_2 points. We apply the RANSAC technique on the point correspondences to extract the largest motion plane in the image, which contains k_m correspondences. The ratio of k_m and $min(k_1, k_2)$ is incorporated in the similarity computation. If a reasonable portion of the detected points contributes to the meaningful correspondences, the computed homography then would be valid. Otherwise, the computed homography should be rejected. The similarity between these two non-facial key-frames is defined as,

$$SimN(\phi_{(i,j)}, \phi_{(p,q)}) = I_{[\mu,1]}\Big(\frac{k_m}{min(k_1, k_2)}\Big) exp\Big(-\frac{\epsilon^2(\phi_{(i,j)}, \phi_{(p,q)})}{2\sigma^2}\Big), \tag{4.5}$$

where $I_{[b]}(a)$ is an indicator function, which gives 1 if a falls in range $[b]$, and 0 otherwise. Criteria μ controls the minimum valid ratio between the number of the interest points and the points that contribute to the correspondences, and σ is a scaling factor. In our experiment, we used $\sigma = 3$. Finally, the visual similarity between stories S_i and S_p based on the non-facial key-frames is defined as,

$$\Gamma_N(i,p) = max\ SimN(\phi_{(i,j)}, \phi_{(p,q)}), \tag{4.6}$$

where $j = \{1, \cdots, n_i\}$ and $q = \{1, \cdots, n_p\}$. The similarity values for Γ_N are also bounded in the range of $[0, 1]$. Examples of the image matching are shown in Figure 4.2. One pair of images is related to the same scene, and the images in the other group are unrelated.

4.1.2 Text Correlation

Sometimes, visual information is insufficient to distinguish differences. Consider the following story: Congress is passing a bill. One news source shows the debate among the senators, while another source shows comments from political activists. The content in each of these stories focuses on the same topic, but they are visually different. In this type of situation, textual information plays a more important role in the semantic linking process.

The textual information is obtained from the automatic speech recognition (ASR) output of the video. The ASR output contains the recognized words from the audio track of the

Starting Time: 11:35:59
Key-Words: federal, grand, jury, meet, investigate, allegation, president, clinton, sexual, relationship, former, white, house, interned, urged, lie, deputy, counsel, bruce, lindsey, chief, staff, john, ...
Ending Time: 11:36:34

Figure 4.3: The key frame of an example story in a video, accompanied by the key words extracted from that story. The starting and ending times are based on the analog version of the video (tape).

news programs with their starting time and duration. For each candidate news story S_i, we extract the key words between its time lines by applying a filter to prune the stop words, such as "the", "and", "or", etc. The story time line covers all the video shots, including both anchor and non-anchor shots. The extracted key words form the *sentence* of the story, which is denoted by Sen_i and has length of $L(Sen_i)$. One example of the story *sentence* is shown in Figure 4.3. If two stories focus on the same topic, there usually is a correlation in the narration of the video. In our approach, this textual linkage between stories S_i and S_p with *sentences* Sen_i and Sen_p is computed by the normalized textual similarity (NTS),

$$NTS(i,p) = \frac{M_{i\to p} + M_{p\to i}}{L(Sen_i) + L(Sen_p)}, \tag{4.7}$$

where $M_{i\to p}$ is the total number of key words in Sen_i that also appear in Sen_p, and $M_{p\to i}$ is the number of key words in Sen_p that also appear in Sen_i. The textual similarity Γ_T between stories S_i and S_p is defined as,

$$\Gamma_T(i,p) = NTS(i,p). \tag{4.8}$$

Based on this definition, it is easy to see that the range for the textual similarity value Γ_T is also $[0, 1]$.

4.1.3 Fusion of Visual and Textual Information

Up to this point, the visual and textual similarities have been determined. The semantic linkage between the news stories is the fusion of these similarities. To determine the form of the fusion, the relationship between the similarities must be defined. The final fusion of the visual and textual similarities is defined as

$$SSim(i,p) = \alpha_V \times \Psi\big(\Gamma_F(i,p), \Gamma_N(i,p)\big) + \alpha_T \times \Gamma_T(i,p), \tag{4.9}$$

where α_V and α_T are constants to balance the importance of the visual and textual similarities, respectively, and $\Psi(\cdot)$ is the fusion function between Γ_F and Γ_N.

The visual similarities Γ_F and Γ_N are computed from two disjoint sets: facial key frames K and non-facial key frames Φ, therefore, these two measures are independent of each other. Thus, the one that has a higher value is dominant over the other. In our formulation, the fusion function $\Psi(\Gamma_F, \Gamma_N)$ is defined as $max(\Gamma_F, \Gamma_N)$. On the other hand, no conclusion of independence can be drawn between the visual and textual similarities. Therefore, we use a linear fusion to combine these two measurements. The constants α_V and α_T balance the importance of the visual and textual effects. The simplest way to select them is to

let $\alpha_V = \alpha_T = 0.5$. However, based on our experience, we have observed that the textual information has a higher impact on the semantic correlation than the visual cues. It provides a better base-line compared to the visual information. Therefore, in our experiments, we set $\alpha_V = 0.35$ and $\alpha_T = 0.65$. Users can also tune them according to their preferences. If more effect is expected from the textual information, α_T can be increased, while α_V is decreased.

A few situations need special attention. Some news stories occur only in the anchor shots. Therefore, only textual information is available. Similarly, one of the visual similarities might be missing due to the absence of the facial or non-facial key frames. To deal with these cases, we have following rules:

- If the facial key frame set K is empty, and the non-facial key frame set Φ is not empty, set $\Gamma_F = \Gamma_N$;
- If Φ is empty, and K is not empty, set $\Gamma_N = \Gamma_F$;
- If both Φ and K are empty, replace $\Psi(\Gamma_F, \Gamma_N)$ by Γ_T. This means that if no visual information is available, the textual similarity plays the dominant role.

Given two news videos containing multiple stories, a story similarity map can be constructed. One example is shown in Figure 4.4. In this similarity map, brighter cells represent higher similarity values.

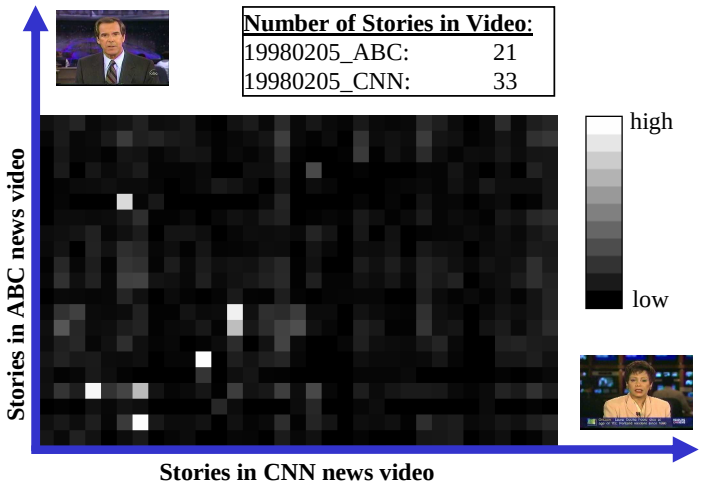


Figure 4.4: The similarity between two videos. The horizontal and vertical axis represent the stories from a CNN and an ABC video respectively. The axes are labelled by the selected anchor images. In this example, brighter cells correspond to higher story similarity values.

4.2 System Performance

We have tested our method on a large dataset from the TRECVID 2003 forum. This dataset is provided by the U.S. National Institute of Standards and Technologies (NIST). It is a open benchmark for the content extraction evaluation and topic search tasks. The dataset contains 100 videos in MPEG-1 format from two news sources: ABC *World News Tonight with Peter Jennings* and CNN *Headline News*. The videos are distributed over 50 days, with each day having a video from ABC and CNN. Each video is around 30 minutes long, covering both the regular news programs and the non-news segments in between the stories, and contains around 20-30 news stories. The TDT2 [4] has provided the ground truth for the news story boundaries generated by manual annotation. Accompanying with the MPEG-1 video data,

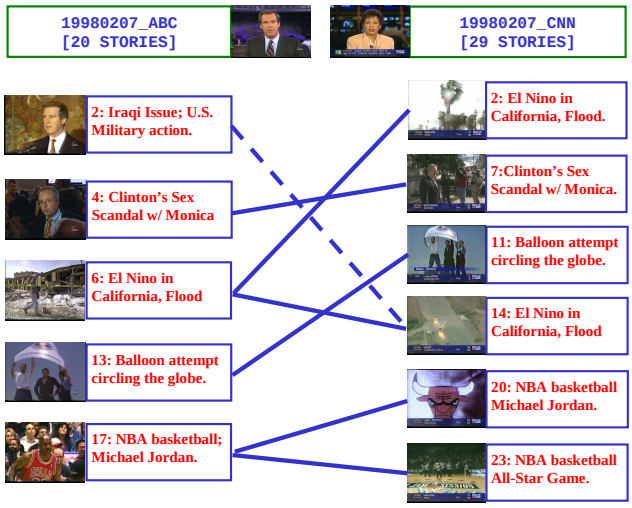


Figure 4.5: One example of story matching. Two news videos from ABC and CNN for the same date are used. In total, seven matches were detected, six of them are labelled as "Relevant" (solid lines), and one is labelled as "Irrelevant" (dashed line). The matched stories are displayed by their first key frame and brief summaries.

NIST also provided the ground truth data for the common shot boundaries, key frames and the automatic speech recognition (ASR) outputs by LDC [27].

Given two news videos from different sources, assume video 1 contains stories $\{S_{1,1}, \cdots, S_{1,n_1}\}$, and video 2 contains stories $\{S_{2,1}, \cdots, S_{2,n_2}\}$. We first compute their similarity map $SimMat$ based on Eqn. 4.9. To classify a match between $S_{1,i}$ and $S_{2,j}$, the value of cell $SimMat(i,j)$ is verified against the predefined threshold. In our experiment, only videos from the same day are matched with each other. One reason for that is because the news stories are interesting only to the audience in their proposed time periods. Stories that are apart in time do not tend to match. However, the proposed method has the capability to match stories across videos, regardless of their time difference. A full set of matches for a pair of example videos is shown in Figure 4.5, and one pair of the matched stories is demonstrated

Figure 4.6: Matched stories from two different sources. The left block contains the key frames and key words extracted from a story in video [19980204_ABC], and the right block contains the key frames and key words extracted from a story in video [19980204_CNN]. The key frames bounded by red boxes provide the visual similarity between these two stories, since both stories are captured at the same presidential palace. The key words in blue boldface are the common words that appear in both of the two stories. From the figure, the reader can easily draw the conclusion that both stories deal with the issue of weapons inspections of the Iraqi presidential palaces.

in detail in Figure 4.6. In Figure 4.6, the key frames and the extracted key words of the stories are shown. The key frames providing the visual similarity are boxed in red, and the common key words in both stories are in blue boldface.

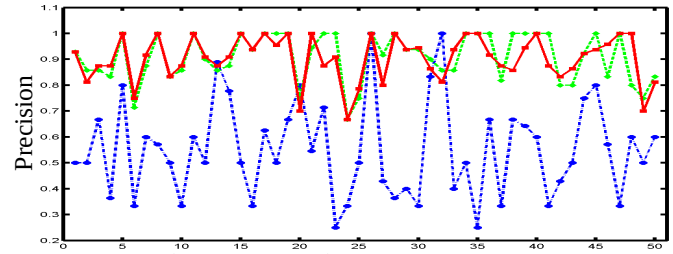
In our evaluation, there are three categories of matched stories: Relevant, Somehow Relevant and Irrelevant. A pair of matched stories is said to be "Relevant" if there is no ambiguity in content. For stories that are partially related, they are classified as "Somehow Relevant". This happens more in the miscellaneous stories, including commercials, where the same commercial is broadcasted by two station. It also happens when a story line actually contains multiple individual stories, which is often in the ground truth data. If the stories focus on completely different topics, "Irrelevant" is assigned as their label. Since there are three categories of matching, we assign different satisfactory scores $w_i \in \{1.0, 0.5, 0.0\}$ to

each of the matches. For each "Relevant" pair detected, it is assigned a satisfactory score of 1.0; for each matching pair with "Somehow Relevant", it is assigned a score of 0.5; finally, if a matching pair is "Irrelevant", a score of 0.0 is assigned. Higher overall scale indicates better satisfaction rate. Ideally, there should be an overall scale of 100% satisfaction. On the other hand, we should also examine how well the system recovers the ground truth matches, i.e., more true detected matches indicate better performance. In the areas of multimedia processing and information retrieval, these are expressed in the precision and recall forms, which are defined as follows for our application:

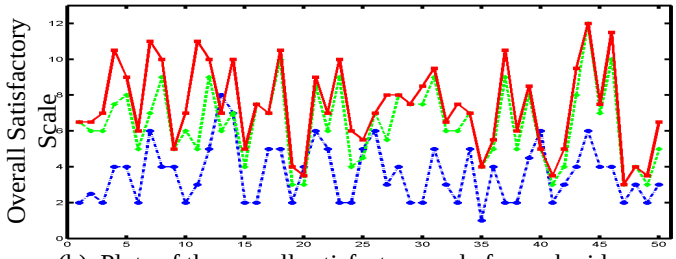
$$\begin{aligned} Precision &= \frac{\sum_{i=1}^{n} w_i}{\sum_{i=1}^{n} 1}, \\ Recall &= \frac{\sum_{i=1}^{n} w_i}{\sum_{j=1}^{m} 1}, \end{aligned} \tag{4.10}$$

where n is the total number of detected matches, and m is the total number of ground truth matches. The precision captures the overall satisfaction rate. However, in our application, due to the vast amount of data, it is very difficult to determine the number of the ground truth matches. In this situation, we have replaced the recall measure with another scale, accumulated satisfaction score, $OSat = \sum_i w_i$. This is equivalent to the recall measure with a fixed number of ground truth matches.

Based on the experimental observations, we have found that textual features often provide more semantic inferences and give a baseline of the overall matching. There is a question on how much impact the visual features can make. In the experiments, the visual features



(a). Plots of the precision for each video.



(b). Plots of the overall satisfactory scale for each video.

Figure 4.7: Comparison between the results obtained using the visually-based, text-based and combined methods. Part (a) shows the comparison of individual precisions, and (b) shows the comparison of the individual overall satisfaction scales. The solid plots are for the combined method, the dashed (–) plots are for the text-based correlation, and the dashed-dotted (-.) plots are for the visually-based correlation. The horizontal axes in both of the figures are the video pairs (date of the videos recorded).

are demonstrated to improve the satisfactory levels. In Figure 4.8, we present the results obtained based only on the visual cues, the results obtained based only on the textual cues, and the results based on both cues, separately. The visually-based method returns total of 332 matches, out of which 173 are relevant, 14 are somehow relevant and 147 are irrelevant. The text-based measure returns total of 351 matches, out of which, 312 are relevant, 12 are somehow relevant and 27 are irrelevant. With the combination of both visual and textual cues, there are 406 total returns, of which 351 are relevant, 31 are somehow relevant, and 24 are irrelevant. If each of the matches is considered equally important, the overall precisions for visually-based, text-based and combined results are 0.5422, 0.9060 and

0.9027, respectively. The combined results have slightly lower precision than the text-based results because the newly introduced wrong matches are more than the newly introduced correct matches. However, the combined results also provide significant positive returns than only using the textual cues. The overall satisfactory scales, which are equivalent to the recall measure, for the visually-based, the text-based and the combined results are 180, 318 and 366.5, respectively. One can observe that there is a significant improvement by incorporating the visual information in the linking process. This can be interpreted as, given a tradeoff of 1% in precision, a 15% gain is obtained in recall. Detailed comparisons between the three scores for each individual video pair is presented in Figure 4.7. The score for the visually-based results are fairly low. This is because that the same person appears in different stories, or the scene has strong parallax, such that Affine is not able to capture the similarity. Outliers in the text-based results are caused by repeating key words, even though the stories that carry these key words are not semantically related. For example, key words "war on drugs" refer to different topic than key words "war in Iraq". However, the textual correlation between them is large due to the common word "war". In this case, the visual information is able to lower the overall story similarity.

	Visual Results				Textual Results				Overall Results			
Date	Matches	Relevant	Somhow Relevant	Irrelevant	Matches	Relevant	Somehow Relevant	Irrelevant	Matches	Relevant	Somehow Relevant	Irrelevant
19980204	4	2	0	2	7	6	1	0	7	6	1	0
19980205	5	2	1	2	7	6	0	1	8	6	1	1
19980207	3	1	2	0	7	6	0	1	8	7	0	1
19980208	11	4	0	9	9	7	1	1	12	10	1	1
19980209	5	4	0	1	8	8	0	0	9	9	0	0
19980211	6	2	0	4	7	5	0	2	8	6	0	2
19980212	10	6	0	4	8	6	2	0	12	10	2	0
19980213	7	4	0	3	9	9	0	0	10	10	0	0
19980214	8	4	0	4	6	5	0	1	6	5	0	1
19980215	6	1	2	3	7	6	0	1	8	7	0	1
19980216	5	3	0	2	5	5	0	0	11	11	0	0
19980217	10	5	0	5	10	9	0	1	11	10	0	1
19980218	9	8	0	1	7	6	0	1	8	7	0	1
19980219	9	7	0	2	8	7	0	1	11	10	0	1
19980220	4	2	0	2	4	4	0	0	5	5	0	0
19980221	6	2	0	4	8	7	1	0	8	7	1	0
19980223	8	5	0	3	7	7	0	0	7	7	0	0
19980225	10	4	2	4	10	10	0	0	11	10	1	0
19980226	3	2	0	1	3	3	0	0	4	4	0	0
19980302	5	4	0	1	4	3	0	1	5	3	1	1
19980304	11	6	0	5	9	8	1	0	9	9	0	0
19980305	7	5	0	2	6	6	0	0	8	7	0	1
19980306	8	1	2	5	9	9	0	0	11	9	2	0
19980307	6	2	0	4	6	4	0	2	9	5	2	2
19980308	10	4	2	4	6	4	1	1	7	5	1	1
19980309	6	6	0	0	7	7	0	0	7	7	0	0
19980310	7	3	0	4	6	5	1	0	10	7	2	1
19980311	11	4	0	7	8	8	0	0	8	8	0	0
19980312	5	2	0	3	8	7	1	0	8	7	1	0
19980313	6	2	0	4	8	7	1	0	9	8	1	0
19980316	6	5	0	1	10	9	0	1	11	9	1	1
19980317	3	3	0	0	7	6	0	1	8	6	1	1
19980318	5	2	0	3	7	6	0	1	8	7	1	0
19980319	10	5	0	5	7	7	0	0	7	7	0	0
19980320	4	1	0	3	4	4	0	0	4	4	0	0
19980321	6	4	0	2	5	5	0	0	6	5	1	0
19980323	6	2	0	4	11	8	2	1	12	10	1	1
19980325	3	2	0	1	5	5	0	0	7	5	2	0
19980326	7	4	1	2	8	8	0	0	9	8	1	0
19980327	10	6	0	4	5	5	0	0	5	5	0	0
19980328	6	2	0	4	3	3	0	0	4	3	1	0
19980329	7	3	0	4	5	4	0	1	6	5	0	1
19980330	8	3	2	3	10	8	0	2	11	9	1	1
19980413	8	6	0	2	13	12	0	1	13	12	0	1
19980414	5	4	0	1	7	7	0	0	8	7	1	0
19980416	7	4	0	3	12	10	0	2	12	11	1	0
19980417	6	2	0	4	3	3	0	0	3	3	0	0
19980418	5	3	0	2	5	4	0	1	4	4	0	0
19980419	4	2	0	2	4	3	0	1	5	3	1	1
19980420	5	3	0	2	6	5	0	1	8	6	1	1

Figure 4.8: Table Summarizing the Story Linking Results. The left group presents the results obtained using only the visual information, the middle group shows the results based only on the textual correlation, and the right group shows the results using both the visual and textual information.

4.2.1 Story Ranking

The results computed from the proposed semantic linking method are further used in the news story ranking. The ranking is based on the repetition of the stories. In a general sense, more "interesting" or "hot" story topics appear more frequently and longer than other stories. In our formulation, we use the frequencies of the stories as the "interestingness" criteria. The story appearing the most is the most "interesting" story of that day.

Given two videos of the same date stamp, containing a and b stories, respectively, then a similarity matrix with size $[a] \times [b]$ is constructed (Figure 4.4). Treating each story as a node in a graph, the similarity map is considered as a weighted bipartite graph (Figure 4.5). To rank the stories, we apply a breadth-first traversal technique on the bipartite graphs to find the connected components of the stories. The stories are ranked by the sizes of their corresponding clusters. The cluster of the related stories can provide more coverage of the news topic, and it is better for further story summarization. The traversal algorithm is as follows:

1. Given videos with a and b stories, construct the semantic similarity matrix and the bi-partite graph;

2. Initialize the label $LL = 1$;

3. For node $i = 1$ to $(a + b)$, perform:

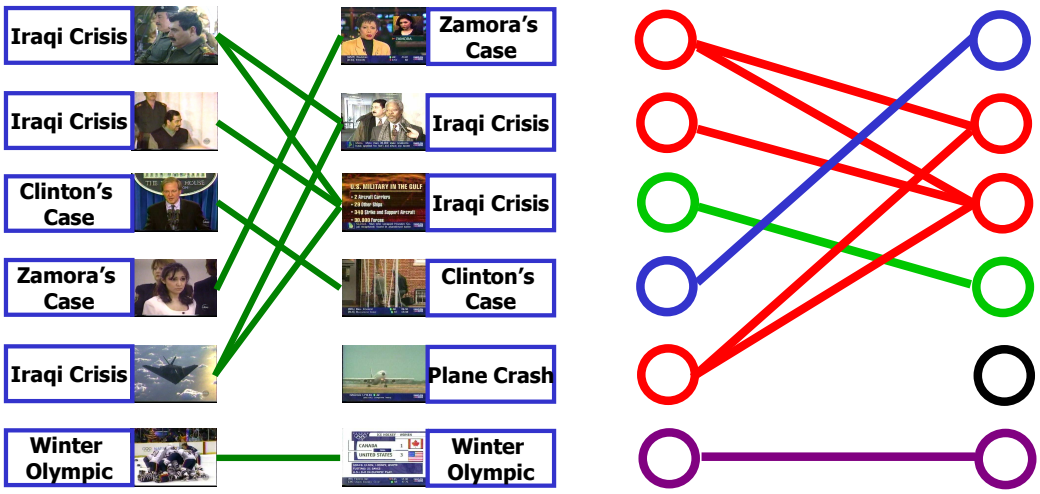


Figure 4.9: A demonstration of the story ranking application. It shows two videos with linked stories, and the story clusters are shown on the right side with different color labels. Based on the ranking results, the viewer can infer that the stories related to the "Iraqi Issue" are the most interested topic on that particular date.

- If the node is un-labelled, set $label(i) = LL$;

  Recursively label its connected neighbors with LL;

  Set $LL = LL + 1$.

4. Compute the sizes of the clusters with $label = \{1, \cdots, LL\}$;

5. Rank the story clusters based on their sizes, where larger size is assigned with higher ranking.

Figure 4.9 shows an examples of story ranking. Linked stories from two videos are presented, and the story clusters are labelled with different colors. We have applied the story-ranking method on the test set and extracted the three most "important" story topics for each day. The detailed results are shown in Figure 4.10. In the results, story clustering

was performed automatically, while the story summary on each cluster is generated manually for presentation purpose.

4.3 Conclusions

In this chapter, we have presented a semantic linking method for finding the similar news stories across different sources. The semantic correlation between two news stories is reflected by the visual similarity and the textual correlation. The key frames of the stories are analyzed. The "body" regions are extracted from the facial key frames for the persons of focus, and the non-facial key frames are globally aligned using the Affine model to detect the repeating events. The language correlation is computed based on the automatic speech recognition (ASR) output of the videos. The visual and textual similarities are fused to provide the overall semantic linkage between the news stories.

The output results of the semantic linking task are further utilized in a news ranking task. The matched stories from the linking process are modelled as the vertices in a bipartite graph. Sub-graphs are detected using the connected-component technique, and the ranking of the story clusters is performed by analyzing the component's size. Since more complete information is provided, the results of the story ranking task can be applied for better summarization of the stories.

Date	First Ranking	Second Ranking	Third Ranking
19980204	Crisis on Iraq.	Clinton's Investigation.	Military Sex Hurrassment Case.
19980205	Crisis on Iraq.	Clinton's Investigation.	El Nino in California.
19980207	El Nino in California.	NBA, Michael Jordan.	Clinton's Investigation.
19980208	El Nino in California.	Clinton's Investigation.	Crisis on Iraq.
19980209	El Nino in California.	Clinton's Investigation.	Crisis on Iraq.
19980211	Clinton's Investigation.	Crisis on Iraq.	Tragedy in Italy by US Airforce.
19980212	Winter Olympics Games.	Presidential Veto Law.	Clinton's Investigation.
19980213	Surgeon General Nomination.	Market and Stocks.	Valentine's Day.
19980214	War on Illegal Drugs.	Clinton's Investigation.	Crisis on Iraq.
19980215	El Nino in California.	Hari Carry in Hospital.	Winter Olympic Games.
19980216	Taiwan Plane Crashed.	Crisis on Iraq.	Winter Olympic Games.
19980217	Crisis on Iraq.	Zamora's Case.	Clinton's Investigation.
19980218	Crisis on Iraq.	Winter Olympic Games.	Clinton's Investigation.
19980219	Clinton's Investigation.	Crisis on Iraq.	US Trade Deficit.
19980220	Crisis on Iraq.	Crisis on Iraq.	American Wrestlers Visit Iran.
19980221	Crisis on Iraq.	Biological Weapon in Nevada.	El Nino in California.
19980223	Crisis on Iraq.	Tornado in Florida.	Union Worker Strick.
19980225	Clinton's Investigation.	Tornado in Florida.	Market and Stocks.
19980226	Crisis on Iraq.	Winfield Opera's Case.	Internet Sales Tax.
19980302	Crisis on Iraq.	Princess Dianna's Accident.	Uranian Bombs.
19980304	Sexual Harrassment for Same Sex.	Military Sex Hurrassment Case.	First Female Space Shuttle Pilot.
19980305	Clinton's Investigation.	Market and Stocks.	Blood Transfusion.
19980306	Unemployment Rate.	Shooting of Lottary Workers.	Clinton's Investigation.
19980307	White Superemist Suspects.	Clinton's Investigation.	Holicopter Crashed in California.
19980308	El Nino in California.	Cirsis on Kosovo.	Clinton's Investigation.
19980309	Woodward's case.	Winter Weather Across the Nation.	Clinton's Investigation.
19980310	Military Sex Hurrassment Case.	Winter Weather Across the Nation.	Clinton's Investigation.
19980311	Coffi Annan Visit US.	Clinton's Investigation.	Winter Weather Across the Nation.
19980312	Bi-Partison Legislation in Senate.	Asteroid 1997-AF-11.	Winter Weather Across the Nation.
19980313	Clinton's Investigation.	Market and Stocks.	Military Sex Hurrassment Case.
19980316	Clinton's Investigation.	Vatican Released WW2 Documents.	Separation of Sex in Military.
19980317	Clinton's Investigation.	Market and Stocks	House Construction.
19980318	Clinton's Investigation.	IRS Reform Plan.	Crisis on Kosovo.
19980319	Clinton's Investigation.	Murcoch's Sale.	US Trade Deficit.
19980320	Clinton's Investigation.	Breast Cancer Pill Approved.	American Policy to Cuba.
19980321	Social Security Issue.	Tornado in Southern States.	Pope Johe Paul II in Negiria.
19980323	President Clinton in Africa.	Oil Prices Up.	Prostate Cancer.
19980325	Arkensas School Shooting.	President Clinton in Africa.	Low Mortgage Rate.
19980326	President Clinton in Africa.	Arkensas School Shooting.	Crisis on Iraq.
19980327	President Clinton in Africa.	Market and Stocks.	Personal Incomes Increase.
19980328	Arkensas School Shooting.	Hospital Deaths in California.	Explosion in Arizona.
19980329	Clinton's Investigation.	Peru Plane Crashed.	Hospital Deaths in California.
19980330	Market and Stocks.	New Home Sale Increase.	President Clinton in Africa.
19980413	Bank Merging.	IRS Tax Return Filing.	Annual Parade in Northern Ireland.
19980414	President Clinton is visiting Taxes.	IRS Tax Return Filing.	South Africa President Mandella.
19980416	Tornado in Southern States.	Former Cambodian Dictator Died.	Violence on Television.
19980417	Tornado in Southern States.	Clinton's Speech in Chile.	US Trade Deficit.
19980418	Tornado in Southern States.	Clinton's Speech in Chile.	New Experiments in Space.
19980419	Wang Dan is Released.	Oklahoma City Bombing Anniversary.	Internet Help on Health.

Figure 4.10: Table Summarizing the Story Ranking Results. The three most "interesting" topics are shown for each day in the dataset.

CHAPTER 5

SPATIOTEMPORAL VIDEO ATTENTION

In this chapter, we present a novel bottom-up spatiotemporal video attention detection framework. The proposed method is able to provide potential locations of both prominent objects in images and interesting activities in video sequences. Video attention methods are generally classified into two categories: *top-down approaches* and *bottom-up approaches*. Methods in the first category, *top-down approaches*, are task-driven, where prior knowledge of the target region is known before the detection process. This is based on the cognitive knowledge of the human brain, and it is a spontaneous and voluntary process. Traditional rule-based or training-based object detection methods are the examples in this category. On the other hand, the second category, *bottom-up approaches*, are usually referred as the stimuli-driven techniques. This is based on the human reaction to external stimuli, such as bright color, distinctive shape or unusual motion, and it is a compulsory process.

The proposed video attention detection framework is in the bottom-up fashion. Here, the saliency maps based on the spatial and temporal features are generated separately and dynamically fused to produce the final spatiotemporal saliency map. The temporal attention

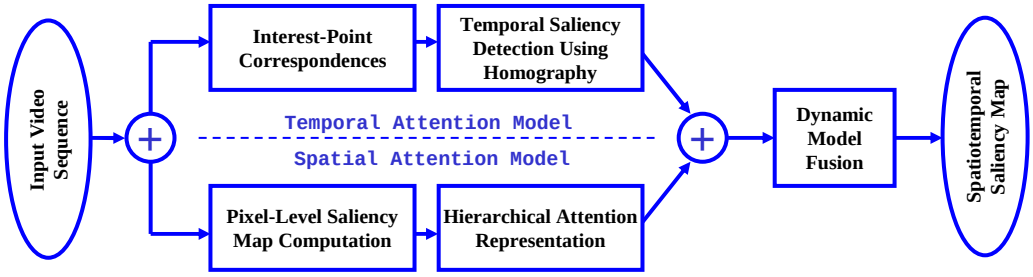
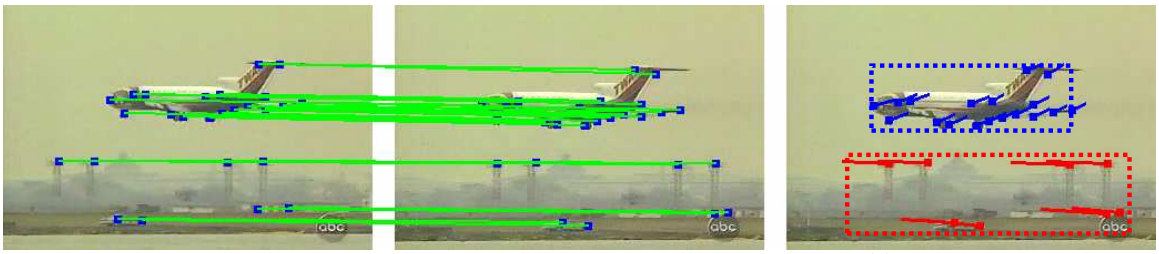


Figure 5.1: Work flow of the proposed spatiotemporal attention detection framework. It consists of two components, temporal attention model and spatial attention model. These two models are combined using a dynamic fusion technique to produce the overall spatiotemporal saliency maps.

model is based on the analysis of the planar motions in the scene, while the spatial attention model is based on the color contrast. The flow of the proposed framework is described in Figure 5.1. The organization of the rest of this chapter is as follows: The temporal and spatial attention models are presented in Sections 5.1 and 5.2, respectively. Section 5.3 describes the dynamic fusion method to combine the two individual attention models. Section 5.4 presents the system performance with extensive experimental results. Finally, Section 5.5 concludes our work.

5.1 Temporal Attention Model

In the temporal attention detection, saliency maps are often constructed by computing the motion contrast between image pixels. Most of the previously developed methods generate dense saliency maps based on pixel-wise computations, mostly dense optical flow fields (Section 2.3). However, it is well known that optical-flows at edge pixels are noisy if mul-



(a) Image 1 (b) Image 2 (c) Motion Regions

Figure 5.2: One example of the point matching and motion segmentation results. Figure (a) and figure (b) show two consecutive images. The interest points in both images and their correspondences are presented. The motion regions are shown in figure (c).

tiple motion layers exist in the scene. Furthermore, dense optical flows maybe erroneous in regions with less texture. In contrast, point correspondences (also known as the sparse optical flows) between images are comparatively accurate and stable. In this section, we propose a novel approach for computing the temporal saliency map using the point correspondences in video sequences. The proposed temporal saliency computation utilizes the geometric transformations between images, which model the planar motions of the moving segments.

Given images in a video sequence, feature points are localized in each image using the interest point detection method. Correspondences between the matching points in consecutive frames are further established by analyzing the properties of image patches around the feature points. In our framework, we have applied the Scale Invariant Feature Transformation (SIFT [59]) operator to find the interest points and compute the correspondences between them across video frames. One example of the interesting point matching is shown in Figure 5.2. Let $\mathbf{p}_m = (x_m, y_m)$ be the m-th point in the first image and $\mathbf{p}'_m = (x'_m, y'_m)$

be its correspondence in the second image. Given the point correspondences, the temporal saliency value $SalT(\mathbf{p}_i)$ of point $\mathbf{p}_i$ is computed by modelling the motion contrast between this target point and other points,

$$SalT(\mathbf{p}_i) = \sum_{j=1}^{n} DistT(\mathbf{p}_i, \mathbf{p}_j), \tag{5.1}$$

where n is the total number of correspondences. $DistT(\mathbf{p}_i, \mathbf{p}_j)$ is some distance function between $\mathbf{p}_i$ and $\mathbf{p}_j$. In our formulation, we analyze the geometric transformations between images. The motion model used is homography. Homography is used for modelling the planar transformations. The interesting point $\mathbf{p} = [x, y, 1]^T$ and its correspondence $\mathbf{p}' = [x', y', 1]^T$ can be associated by,

$$\begin{bmatrix} \hat{x}' \\ \hat{y}' \\ \hat{t}' \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}. \tag{5.2}$$

Here, $\hat{\mathbf{p}}' = [\hat{x}', \hat{y}', \hat{t}']^T$ is the projection of $\mathbf{p}$ in the form of homogeneous coordinates. Parameters $\{a_i, i = 1, \cdots, 8\}$ capture the transformation between two matching planes, and they can be estimated by providing at least four pairs of correspondences. For simplicity, we use $\mathbf{H}$ to represent the transformation matrix in the rest of the text. Also, we normalize $\hat{\mathbf{p}}_i$, such that its third element is 1. Ideally, $\hat{\mathbf{p}}'$ should be the same as $\mathbf{p}'$. With noise present in the imagery, a point $\hat{\mathbf{p}}'$ matches with $\mathbf{p}'$ with an error computed after applying $\mathbf{H}$, as,

$$\epsilon(\mathbf{p}_i, \mathbf{H}) = \| \hat{\mathbf{p}}'_i - \mathbf{p}'_i \| . \tag{5.3}$$

Motions of objects are only meaningful when certain reference is defined. For instance, a car is said "moving" only if visible background is present in the scene and disagrees with the car in terms of the motion direction. This fact indicates that multiple moving objects are in the scene to indicate local motion existence. In these types of situations, a single homography is insufficient to model all the correspondences in the imagery. To overcome this problem, we apply the RANSAC algorithm on the point correspondences to estimate multiple homographies that model different motion segments in the scene. The estimated homographies are later used in the temporal saliency computation process.

For each homography $\mathbf{H}_m$ estimated by RANSAC, a list of points $\mathbf{L}_m = \{\mathbf{p}_1^m, \cdots, \mathbf{p}_{n_m}^m\}$ are considered as its inliers, where n_m is the number of inliers for $\mathbf{H}_m$. Given the homographies and the projection error definition in Eqn. 5.3, we can define the motion contrast function in Eqn. 5.1 as,

$$DistT(\mathbf{q}_i, \mathbf{q}_j) = \epsilon(\mathbf{q}_i, \mathbf{H}_m), \tag{5.4}$$

where $\mathbf{q}_j \in \mathbf{L}_m$. The sizes of the inlier sets play dominant role in the current saliency computation. It is well known that the spatial distribution of the interest points is not uniform due to variance in the texture contents of image parts. Sometimes, relatively larger moving objects/regions may contribute less trajectories, while smaller regions but with richer

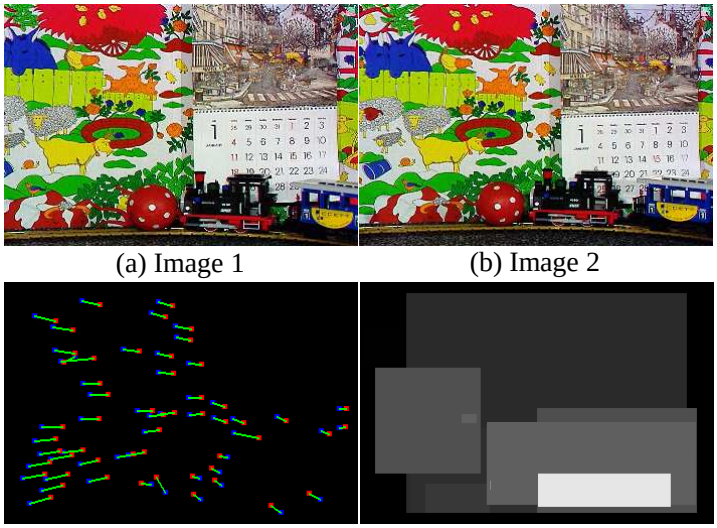
texture provide more trajectories. One example is shown in Figure 5.2. In these cases, the current temporal saliency definition is not realistic. Larger regions with less points, which often belong to the backgrounds, will be assigned with higher attention values. While foreground objects, which are supposed to be the true attended regions, will be assigned with lower attention values, if they possess more interest-points. To avoid this problem, we incorporate the spanning area information of the moving regions. The spanning area of a homography $\mathbf{H}_m$ is computed as,

$$\alpha_m = \Big(max(x_i^m) - min(x_i^m)\Big) \times \Big(max(y_i^m) - min(y_i^m)\Big), \tag{5.5}$$

where $\forall \mathbf{p}_i^m \in \mathbf{L}_m$, and α_i is normalized with respect to the image size, such that $\alpha_i \in [0, 1]$. In the extreme cases, where $max(x_i^m) = min(x_i^m)$ or $max(y_i^m) = min(y_i^m)$, to avoid zero values of α_m, the corresponding term in Eqn. 5.5 is replaced with a non-zero constant number (in the experiment, we use 0.1). The temporal saliency value of a target point $\mathbf{p}$ is finally computed as,

$$SalT(\mathbf{p}) = \sum_{j=1}^{M} \alpha_j \times \epsilon(\mathbf{p}, \mathbf{H}_j), \tag{5.6}$$

where M is the total number of homographies in the scene. In the degenerated cases, where some point correspondences do not belong to inlier sets of any of the estimated homographies, we apply a simplified form of the homography to each of these point correspondences. Suppose $\{\mathbf{p}_t, \mathbf{p}'_t\}$ is one of the "left-out" correspondences. The transformation is defined as



(a) Image 1 (b) Image 2

(c) Point Correspondences (d) Temporal Saliency Map

Figure 5.3: An example of the temporal attention model. (a) and (b) show two consecutive images of the input sequence. (c) shows the interest-point correspondences. (d) shows the detected temporal saliency map using the proposed homography-based method. In this example, the camera follows the moving toy train from right to left. Thus, intuitively, the attention region should correspond to the toy train. The saliency map also suggests that the second attended region corresponds to the moving calender. Brighter color represents higher saliency value.

a translation matrix $\mathbf{H}_t = [1\ 0\ d_x^t;\ 0\ 1\ d_y^t;\ 0\ 0\ 1]$, where $d_x^t = x_t' - x_t$ and $d_y^t = y_t' - y_t$, and the inlier set $\mathbf{L}_t = \mathbf{p}_t$.

Up to this point, we have the saliency values of individual points and the spanning regions of the homographies, which correspond to the moving objects in the scene. To achieve object-level attention for $\mathbf{H}_m$, the average of the saliency values of the inliers $\mathbf{L}_m$ is considered as the saliency value of the corresponding spanning region. All the image pixels in the same spanning region have the same saliency value. Since the resulting regions are rectangular, it is likely that an image pixel is covered by multiple spanning regions. In this case, the pixel is assigned with the highest saliency value possible. If the pixel is not covered by any spanning

region, its saliency value is set to zero. One example of the proposed temporal saliency map computation is demonstrated in Figure 5.3, where the camera follows a moving toy train from right to left, and apparently the attention region in the sequence corresponds to the moving toy train.

5.2 Spatial Attention Model

When viewers watch a video sequence, they are attracted not only by the interesting events, but also sometimes by the interesting objects in still images. This is referred to as the spatial attention. Based on the psychological studies, human perception system is sensitive to the contrast of visual signals, such as color, intensity and texture. With this underlying assumption, we propose an efficient method for computing the spatial saliency maps using the color statistics of images. The algorithm is designed with a linear computational complexity with respect to the number of image pixels. The saliency map of an image is built upon the color contrast between image pixels. The saliency value of a pixel I_k in an image I is defined as,

$$SalS(I_k) = \sum_{\forall I_i \in I} \parallel I_k - I_i \parallel, \tag{5.7}$$

where the value of I_i is in the range of $[0, 255]$, and $||\cdot||$ represents the color distance metric. This equation is expanded to have the following form,

$$SalS(I_k) = ||I_k - I_1|| + ||I_k - I_2|| + \cdots + ||I_k - I_N||, \tag{5.8}$$

where N is the total number of pixels in the image. Given an input image, the color value of each pixel I_i is known. Let $I_k = a_m$, and Eqn. 5.8 is further restructured, such that the terms with the same I_i are rearranged to be together,

$$\begin{aligned} SalS(I_k) &= ||a_m - a_0|| + \cdots + ||a_m - a_1|| + \cdots + \cdots, \\ SalS(a_m) &= \sum_{n=0}^{255} f_n ||a_m - a_n||, \end{aligned} \tag{5.9}$$

where f_n is the frequency of pixel value a_n in the image. The frequencies are expressed in the form of histograms, which can be computed in $O(N)$ time order. Since $a_n \in [0, 255]$, the color distance metric $\parallel a_m - a_n \parallel$ is also bounded in the range of $[0, 255]$. Since this is a fixed range, a distance map D can be constructed in constant time prior to the saliency map computation. In this map, element $D(x, y) = \parallel a_x - a_y \parallel$ is the color difference between a_x and a_y. One color difference map is shown in Figure 5.4. Given the histogram $f_{(\cdot)}$ and the color distance map $D(\cdot, \cdot)$, the saliency value for a pixel I_k is computed as,

$$SalS(I_k) = SalS(a_m) = \sum_{n=0}^{255} f_n D(m, n), \tag{5.10}$$

which executes in a constant time order. Thus, instead of computing the saliency values of all the image pixels using Eqn. 5.7, only the saliency values of colors $\{a_i, i = 0, \cdots, 255\}$ are

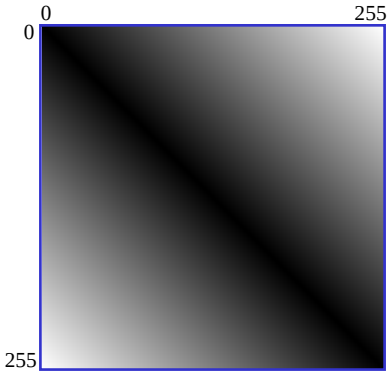


Figure 5.4: The distance map between the gray-level color values, which can be computed prior to the pixel-level saliency map computation. Brighter elements represent larger distance values.

necessary for the generation of the final saliency map. One example of the pixel-level spatial saliency computation is shown in Figure 5.5.

Greatly inspired by the work presented in [62], we propose a hierarchical representation for the spatial attention model based on the pixel-level saliency map computed previously. Two levels of attentions are achieved: attended points and attended regions. Attended points are analogous to the direct response of human perception system to external signals. They are computed as the image pixels with the locally maximum spatial saliency values. On the other hand, region-level attention representation provides attended objects in the scene. One simple way to achieve the attended regions is to apply the connected-component algorithm to find the bright regions. However, as shown in Figure 5.5, pixels with low attention values are embedded in high-value regions. Connected-component algorithm will fail to include these pixels in the attended region. Furthermore, connected-component method tends to generate over-detection of the attended regions. In this paper, we present a region growing technique for detecting the attended regions, which is able to resolve the above mentioned problems.

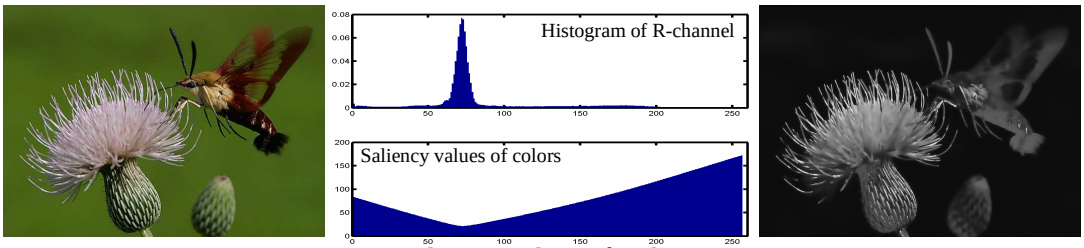


Figure 5.5: An example of the spatial saliency computation. The left figure shows the input image. The center-top figure shows the histogram of the R-channel of the image, while the center-bottom figure shows the saliency values of the colors. The horizontal axis represents the values of the colors, where $a_n \in [0, 255]$. The saliency values are close to what human expects, since higher frequency indicates repeating information in the image, and therefore, are relatively unattractive. The right figure shows the resulting spatial saliency map.

In our formulation, the attended regions are firstly initialized based on the attended points computed previously. Given an attended point $\mathbf{c}$, a rectangular box centered at $\mathbf{c}$ with the unit dimensions is created as the seed region $\mathbf{B_c}$. The seed region is then iteratively expanded by moving its sides outward by analyzing the energy around its sides. The attended region expansion algorithm is described as follows,

1. For each side $i \in \{1, 2, 3, 4\}$ of region $\mathbf{B}$ with length l_i, two energy terms $E(s_i)$ and $E(s_i')$ are computed for both its inner and outer sides s_i and s_i', respectively, as shown in Figure 5.6. The potential for expanding side i outward is defined as follows,

$$EP(i) = \frac{E(s_i)E(s_i')}{l_i^2}, \tag{5.11}$$

where l_i^2 is for the purpose of normalization.

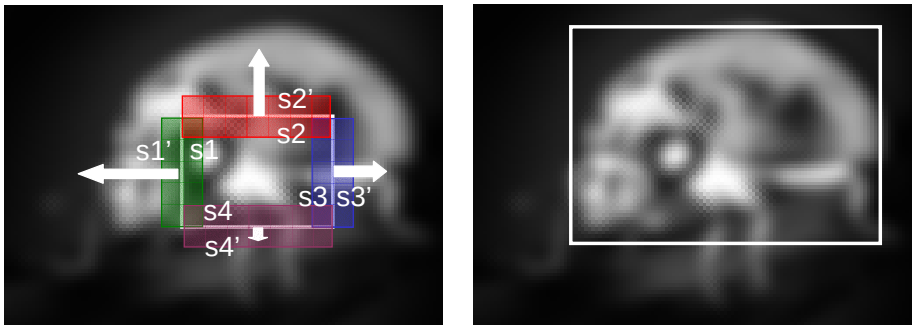


Figure 5.6: An example of the attended region expansion using the pixel-level saliency map. A seed region is created on the left. Expanding potentials on all four sides of the attended region are computed (shaded regions). The lengths of the arrows represent the strengths of the expansions on the sides. The final attended region is shown on the right.

2. Expand the region by moving side i outward with a unit length if $EP(i) > Th$, where Th is the stopping criteria for the expansion. In the experiment, the unit length is 1 pixel.

3. Repeat steps 1 and 2 until no more side of **B** can be further expanded, i.e., all the corresponding expansion potentials are below the defined threshold.

It should be noted that the expansion potential defined in Eqn. 5.11 is designed in such a way, that the attended region is expanded if and only if both the inner and outer sides have high attention values. The expansion stops at the boundary between the high value regions representing the interesting objects and the low value regions for the background. A demonstration of the expanding process is shown in Figure 5.6. It is possible that the attended regions initiated using different attended points eventually cover the same image region. In this case, a region merging technique is applied to merge the attended regions that cover the same target image region by analyzing the overlapping ratio between the regions.

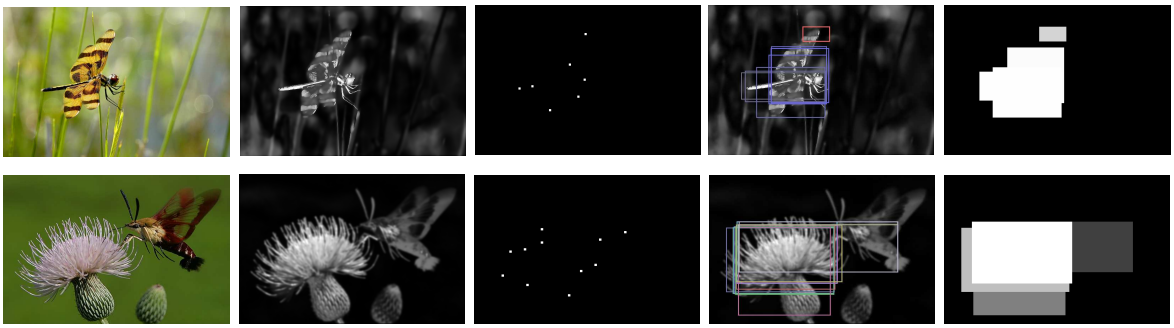


Figure 5.7: The results of spatial attention detection on two testing images. Column (a) shows the input images; column (b) shows the pixel-level spatial saliency maps; column (c) presents the detected attention points; column (d) shows the expanding boxes from the attention points in (c); finally, column (e) shows the region-level saliency maps of the images.

To be consistent with the temporal attention model, the final spatial saliency map reveals the attended regions in the rectangular shapes. Detailed results of the spatial attention detection on two images are shown in Figure 5.7.

5.3 Dynamic Model Fusion

In the previous sections, we have presented the temporal and spatial attention models separately. These two models need to collaborate in a meaningful way to produce the final spatiotemporal video saliency maps. Psychological studies reveal that, human vision system is more sensitive to motion contrast compared to other external signals. Consider a video sequence, in which the camera is following a person walking, while the background is moving in the opposite direction of the camera's movement. In general, humans are more interested

in the followed target, the walking person, instead of the his surrounding regions, the background. In this example, motion is the prominent cue for the attention detection compared to other cues, such as color, texture and intensity. On the other hand, if camera is being static or only scanning the scene, in which motion is relatively uniform, then the human perception system is attracted more by the contrasts caused by other visual stimuli, such as color and shape. In summary, we propose the following criteria for the fusion of temporal and spatial attention models,

1. If strong motion contrast is present in the sequence, temporal attention model should be more dominant over the spatial attention model.

2. On the other hand, if the motion contrast is low in the sequence, the fused spatiotemporal attention model should incorporate the spatial attention model more.

Based on these two criteria, simple linear combination with fixed weights between two individual models is not realistic and would produce unsatisfactory spatiotemporal saliency maps. Rather, we propose a dynamic fusion technique, which satisfies the aforementioned criteria. It gives a higher weight to the temporal attention model, if high contrast is present in the temporal saliency map. Similarly, it gives a higher weight to the spatial model, if the motion contrast is relatively low.

Finally, the spatiotemporal saliency map of an image I in the video sequence is constructed as,

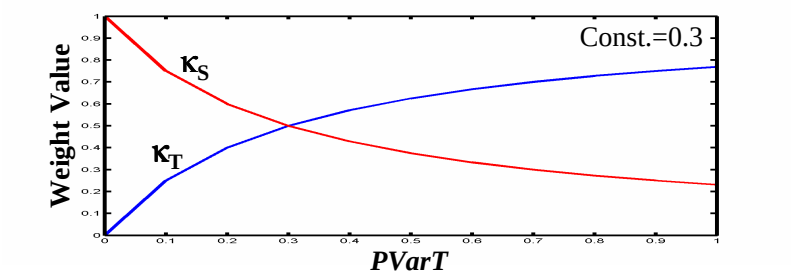


Figure 5.8: Plots of the dynamic weights, κ_T and κ_S, with respect to $PVarT$ ($Const = 0.3$). The fusion weight of the temporal attention model increases with $PVarT$.

$$Sal(I) = \kappa_T \times SalT(I) + \kappa_S \times SalS(I), \tag{5.12}$$

where κ_T and κ_S are the dynamic weights for the temporal and spatial attention models, respectively. These dynamic weights are determined in terms of the variance of $SalT(I)$. One special situation needs to be considered carefully. Consider one scene with a moving object whose size is relatively small compared to the background. The variance of the temporal saliency map in this case would be low by the overwhelming background saliency values and does not truly reflect the existence of the moving object. In this case, we compute a variance-like measure, *pseudo-variance*, which is defined as $PVarT = max(SalT(I)) - median(SalT(I))$. The weights κ_T and κ_S are then defined as,

$$\kappa_T = \frac{PVarT}{PVarT + Const}, \quad \kappa_S = \frac{Const}{PVarT + Const}, \tag{5.13}$$

where $Const$ is a constant number. From Eqn. 5.13, if the motion contrast is high in the temporal model, then the value of $PVarT$ increases. Consequently, fusion weight of the

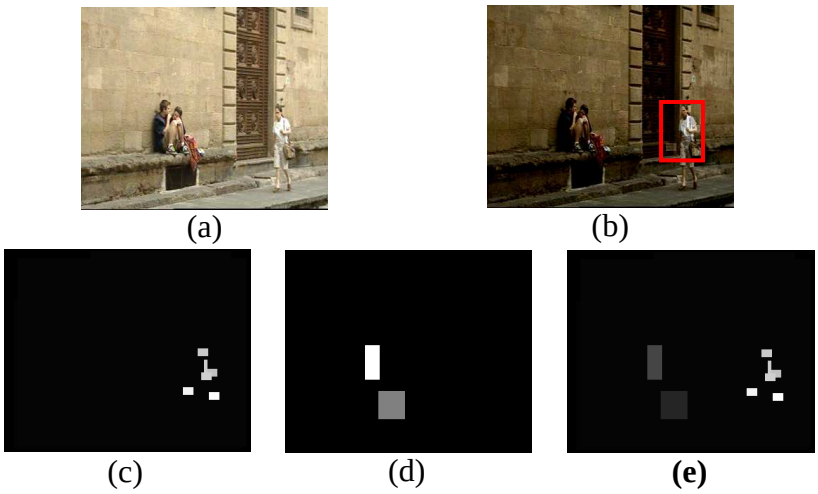


Figure 5.9: An example of model fusion. The video has two sitting people and one walking person. (a) is the key-frame of the video. (c) shows the temporal saliency map. (d) shows the region-level spatial saliency map. (e) is the combined spatiotemporal saliency map. Obviously, the moving object (the walking person) catches more attention than the still regions (sitting persons). Thus, it is assigned higher attention values. The attended region of the interesting action is shown in (b).

temporal model, κ_T, is also increased, while the fusion weight of the spatial model, κ_S, is decreased. The plots of κ_T and κ_S with respect to $PVarT$ are shown in Figure 5.8. One example of the spatiotemporal attention detection is shown in Figure 5.9, which shows a person is walking in front of the two sitting people. The moving object (walking person) is highlighted by the detected attention region.

5.4 Performance Evaluation

To demonstrate the effectiveness of the proposed spatiotemporal attention model, we have extensively applied the method on two types of video sequences, labelled Testing Set 1 and

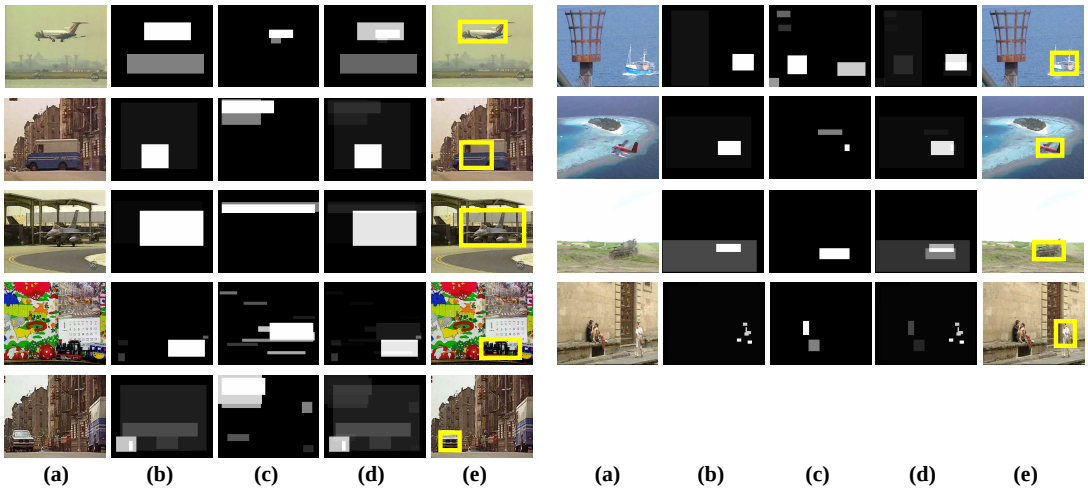


Figure 5.10: Spatiotemporal attention detection results for the testing videos in Testing Set 1. Column (a) shows the representative frames of the videos; column (b) shows the temporal saliency maps; column (c) shows the spatial saliency maps; column (d) shows the fused spatiotemporal saliency maps; and column (e) shows the regions that correspond to potential interesting actions in clips. It should be noted that when rich texture exists in the scene, temporal attention model is able to detect the attended regions using motion information, while the spatial model fails.

Testing Set 2. The testing sequences are obtained from feature films and television programs. Testing Set 1 contains nine video sequences, each of which has one object moving in the scene, such as moving cars and flying airplanes. The detailed results of Testing Set 1 are shown in Figure 5.10. The following information is presented: the representative frames of the testing videos (Figure 5.10(a)), the temporal saliency maps of the representative frames (Figure 5.10(b)), the spatial saliency maps of the representative frames (Figure 5.10(c)), the final spatiotemporal saliency maps (Figure 5.10(d)) and the detected regions that correspond to the prominent actions in the videos (Figure 5.10(e)). It should be noted that, for those videos that have richer texture, the spatial attention model generates less meaningful saliency maps. However, with the help of the proposed dynamic model fusion technique, the temporal

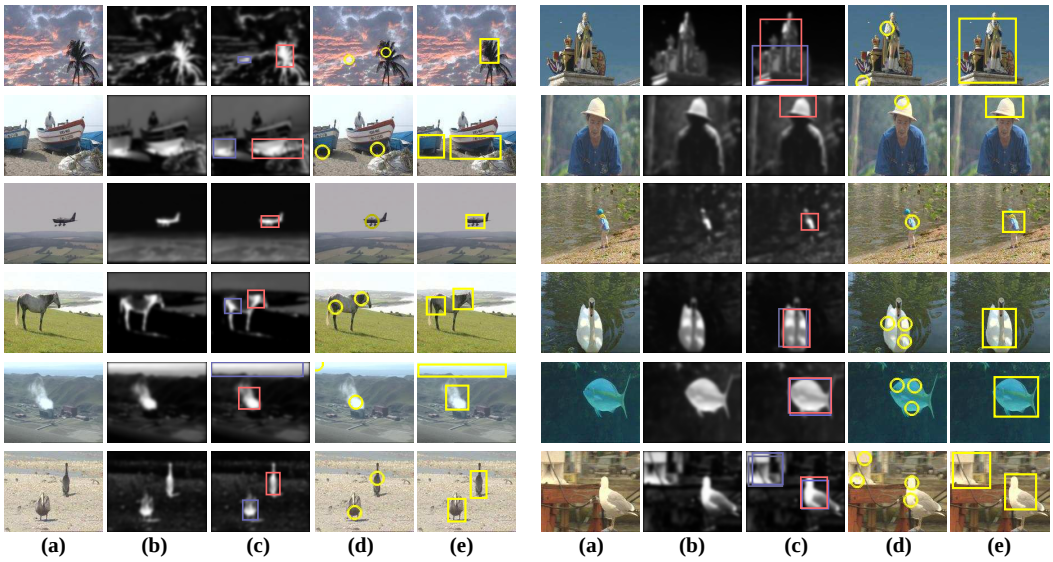


Figure 5.11: Spatiotemporal attention detection results for Testing Set 2. Column (a) shows the representative frames of the videos; column (b) shows the pixel-level spatial saliency maps; column (c) shows the extended bounding boxes using the proposed expansion method; column (d) shows the detected attended points; finally, column (e) shows the detected attended regions. Note that column (e) shows different information from column (c). If the extended bounding boxes overlaps with great amount, they are merged to produce a single attended region. Small bounding boxes are also removed.

attention model becomes dominant and is able to detect the regions where interesting actions happen (as shown in Figure 5.10(e)). This exactly fits human perceptual reactions to motion contrast in these types of situations regardless of visual texture in the scene.

The second testing set, Testing Set 2, contains video sequences without prominent motions. The videos are mainly focusing on the static scene settings or with uniform global motions, i.e., there is no motion contrast in the scene. In this case, the spatial attention model should be dominant over the temporal model. Some results on the testing sequences in Testing Set 2 are shown in Figure 5.11. The presented results include the following: the representative key-frames of videos (Figure 5.11(a)), the pixel-level spatial saliency maps

(Figure 5.11(b)), the expanding regions (Figure 5.11(c)), the attended points in the representative frames (Figure 5.11(d)) and the attended regions in the representative frames (Figure 5.11(e)). Temporal saliency maps are not shown since they are uniform and carry less information.

Assessing the effectiveness of a visual attention detection method is a very subjective task. Therefore, manual evaluation by humans is an important and inevitable element in the performance analysis. In our experiments, we have invited five assessors with both computer science and non-computer science backgrounds to evaluate the performance of the proposed spatiotemporal attention detection framework. Adopting the evaluation ideas from [62], each assessor is asked to give a vote on how satisfactory he or she thinks the detected attended region is for each testing sequence. There are three types of satisfactions, *good*, *acceptable* and *failed*. *Good* represents the situations where the detected attended regions/actions exactly match what the assessor thinks. As pointed out by [62], it is somehow difficult to define the *acceptable* cases. The reason is that different assessors have different views even for the same video sequence. One attended region considered inappropriate by one assessor may be considered perfect to another. In our experimental setup, if the detected attended regions in a video sequence do not cover the most attractive regions, but instead cover less interesting regions, the results are considered *acceptable*. As described by this definition, being *acceptable* is subjective to individual assessors. For instance, in the last example in Figure 5.10, one assessor considers the walking person is more interesting than the other two sitting people. Then, the current results shown is considered *good* to this assessor. However,

System Performance Evaluation			
Data Set	Good	Acceptable	Failed
Testing Set 1 (Moving Objects)	0.82	0.16	0.02
Testing Set 2 (Attended Points)	0.70	0.12	0.18
Testing Set 2 (Attended Regions)	0.80	0.12	0.08

Figure 5.12: System performance evaluation for three categories, *Testing Set 1 with moving objects*, *Testing Set 2: attended point detection* and *Testing Set 2: attended region detection*.

another assessor may be attracted by the sitting people the most, then by the walking person. In this case, the current result is considered *acceptable* to the second assessor.

We have performed the evaluation on both testing sets with three categories: (1) Testing Set 1 with moving objects in the scene; (2) Testing Set 2 with detected attended points; and (3) Testing Set 2 with detected attended regions in the scene. Figure 5.12 shows the assessment of all three categories. In this result table, element in row M and column N represents the proportion of the votes on category M with satisfactory level N. The assessment shown in the table demonstrates that the proposed spatiotemporal attention detection framework is able to discover the interesting objects and actions with more than 90% satisfaction rates. The results of attended point detection have a lower satisfaction rate than the other two region-level attention representations. This is due to the fact that, the attended regions possess contextual information among image pixels, and therefore, have richer contents in terms of semantic meanings than image pixels. On the other hand, attended points are isolated from each other, and human perception system responds to them very differently for differ-

ent persons. Due to the lack of semantic meanings, more disagreements between assessors emerged for the detected attended points, and therefore, has lowered the satisfactory score.

Another interesting observation from the experiments is that, as the texture content in the imagery becomes richer, the attention detection performs with lower satisfactory rate. This is clearly shown in the results (Figure 5.11). For videos that have prominent objects with relatively plain background settings, the proposed attention detection method performs well and produces very satisfied attended regions. On the other hand, if the background settings are much richer, some false detections are generated. This is actually a good simulation to the human vision system. As pointed by the psychological studies in [22], human vision system is sensitive to the difference or contrast between the target region and its neighborhood. In the situations where the background settings are relatively uniform, the contrast between the object and the background is larger. Thus, human vision system is able to pick up the target region very easily. On the other hand, if rich background settings are present in the scene, the contrast between the object and the background is less comparing to the former cases, human vision system is distracted by other regions in the scene and less capable to find the target object.

5.5 Conclusions

In this paper, we have presented a spatiotemporal attention detection framework for detecting both attention regions and interesting actions in video sequences. The saliency maps are

computed separately for the temporal and spatial information of the videos. In the temporal attention model, interest-point correspondences and geometric transformations between images are used to compute the motion contrast in the scene. The areas of the spanning regions of the motion groups are incorporated in the motion contrast computation. In the spatial attention model, we have presented a fast algorithm for computing the pixel-level saliency map using the color histograms. A hierarchical attention representation is established. Rectangular attended regions are initialized based on the attended points. They are further iteratively expanded by analyzing the expansion potentials along their sides. To achieve the spatiotemporal attention model, a dynamic fusion technique is applied to combine the temporal and spatial models. The dynamic weights of the two individual models are controlled by the pseudo-variance of the temporal saliency values. Extensive testing has been performed on numerous video sequences to demonstrate the effectiveness of the proposed framework, and very satisfactory results have been obtained.

CHAPTER 6

CONCLUSIONS AND FUTURE DIRECTIONS

In this dissertation, we have presented three new video processing and understanding frameworks. These techniques are the necessities to the understanding of video contents, such as video summarization, video classification and video clustering. In Chapter 2, we have reviewed the previous accomplishments in the following three areas, temporal video scene segmentation, video linking and matching, and visual attention detection. We described our approaches for these three tasks in details in Chapters 3, 4 and 5, respectively.

In order to correctly analyze the video content, videos need to be firstly segmented into meaningful units. These units are generally the collections of video shots, which are coherent to certain aspects, such as sub-themes in movies, stories in broadcast news and family events in home videos. Common approaches for scene segmentation include detecting significant changes in video features and utilizing prior knowledge of a specific domain, which have apparent limitations (Chapter 2). In Chapter 3, we have proposed a novel framework, which is designed in a statistical fashion using the Markov chain Monte Carlo (MCMC) technique with proposal updates including diffusions and jumps. The contribution of our proposed

work is two-fold. First, it is not limited to fixed thresholds, i.e., it is able to detect weak boundaries as well as strong ones. Second, it is general-purpose. We have applied the framework to two domains, home videos and feature films, representing categories of raw videos and produced videos, respectively. High precision and recall scores obtained in both domains indicate the effectiveness and generality of the proposed technique. The proposed work is accepted by the IEEE Transactions on Multimedia for publication (Zhai and Shah [113]).

Once videos are segmented into meaningful units, users can further perform scene understanding tasks to organize the videos. In Chapter 4, we have presented a semantic linking framework for correlating stories in broadcast news videos. Both visual and speech information of the videos are used to compute the semantic similarity between videos. Homographies are estimated using sparse interest-points to detect visually similar key-frames, such as near duplicates. Automatic speech recognition outputs are analyzed to reveal the keyword co-occurrence in the matching stories. The proposed story linking technique is also applied to the story ranking task, which generates the interestingness of the stories. We have obtained very satisfactory performance in both the story linking and ranking tasks. This work was published in the ACM International Conference on Multimedia in 2005 (Zhai and Shah [114]).

One of the video understanding tasks is the object/activity detection and recognition. In order to solve this problem, the target objects/activities need to be firstly localized in the scene. Visual attention detection provides a hierarchical saliency representation of the videos,

which guides the search tasks. In Chapter 5, we have presented an effective and efficient spatiotemporal visual attention detection technique. The temporal and spatial saliency maps are generated separately. Temporal saliency map is computed based on the differences between moving planes in the scene. The spatial saliency is computed based on the contrasts in pixel colors. A dynamic fusion method is designed to combine both the temporal and spatial saliency maps to emphasize the effects of motion contrasts. The proposed attention detection framework has been extensively applied on several types of videos. Both point-level and region-level attentions have been achieved with high user satisfactory rates. The proposed work is accepted by the ACM International Conference on Multimedia in 2006 (Zhai and Shah [112]).

6.1 Future Directions

Scene level video representation provides more information of the video and many times possesses complete story lines of the video. Currently, our proposed framework for linking news stories uses low-level video features, such as speech and interest-points. The high-level semantic similarity is expressed in terms of the similarities computed using these features. It is sometimes difficult to bridge the gap between the low-level video features and high-level video correlation. In our future work, one promising direction is to incorporate high-level semantic concepts of the video, such as government-leader, outdoor scene, people walking, etc. In this new plan, the high-level semantic concepts are firstly detected using audiovisual

features of the video sequences. The semantic similarity between videos is then computed by comparing their semantic concepts. One possible similarity measurement could be the Dice metric, which models the co-occurrence of the semantic concepts in the matching videos.

It should be noted that not any semantic concept is meaningful or applicable in the task of matching videos. For instance, concepts with high frequencies are less discriminative, such as people walking, male speech and face. They may cause false matching due to their relatively high occurrences. On the other hand, concepts that occurs very rarely are too few to be used. Therefore, it is narrowed down to the mid-frequency concepts. There are several ways to determine which concepts fall into this category. Often, when there is the luxury of training data, the frequencies of target semantic concepts can be computed from the training videos and estimated for the testing set. Another common approach is to utilize public available resources, one of which is the well-known WordNet [25].

It comes down to the problem of how to effectively and efficiently classify videos into the semantic concepts. Many times, the concepts describe particular objects or activities, or combinations of these two, and the target objects/activities must be localized before classification. In this case, it is time consuming and erroneous to analyze the global information of the scene. In our future work, we plan to apply our proposed spatiotemporal visual attention detection framework in the high-level semantic concept detection task. This will provides a hierarchical of processing priorities on the image parts. It is expected to limit the search range and eliminate irrelevant image information in the decision making process. To strengthen the effectiveness of the proposed visual attention framework, we also plan

to incorporate other cues in the model, including intensity, texture and orientation. This will make the proposed method able to detect not only the color difference in the image, but also the differences in other signal domains. The bottom-up attention approach detects prominent regions in the scene based on the raw signals. This can also be combined with the top-down approach, which utilizes the prior knowledge of the target objects/activities, to achieve more accurate classification performance.

LIST OF REFERENCES

[1] B. Adams, C. Dorai, S. Venkatesh, "Novel Approach to Determining Tempo and Dramatic Story Sections in Motion Pictures", *International Conference on Image Processing*, 2000.

[2] D.A. Adjeroh, M.C. Lee and I. King, "A Distance Measure for Video Sequences Similarity Matching", *Conference on Multi-Media Database Management Systems*, 1998.

[3] O. Alatas, O. Javed and M. Shah, "Compressed Spatio-temporal Descriptors for Video Matching and Retrieval", *IEEE International Conference on Image Processing*, 2004.

[4] J. Allan, J.G. Carbonell, G. Doddington, J. Yamron and Y. Yang, "Topic Detection and Tracking Pilot Study Final Report", *Broadcast News Transcription and Understanding Workshop*, 1998.

[5] A. Amir, M. Berg, S.F. Chang, G. Iyengar, C.Y. Lin, A Natsev, C. Neti, H. Nock, M. Naphade, W. Hsu, J. Smith, B. Tseng, Y. Wu, and DQ Zhang, "IBM Research TRECVID 2003 Video Retrieval System", *TREC Video Retrieval Evaluation Forum*, 2003.

[6] A. Aner and J.R. Kender, "Video Summaries Through Mosaic-Based Shot and Scene Clustering", *European Conference on Computer Vision*, 2002.

[7] Y.A. Aslandogan and C.T. Yu, "Techniques and Systems for Image and Video Retrieval", *IEEE Transactions on Knowledge and Data Engineering*, vol.11, no.1, pages.56–63, 1999.

[8] J.C. Baccon, L. Hafemeister and P. Gaussier, "A Context and Task Dependent Visual Attention System to Control A Mobile Robot ", *Internationl Conference on Intelligent Robots and System*, 2002.

[9] S. Basu, "Conversational Scene Analysis", *Thesis*, 2002.

[10] O. Boiman and M. Irani, "Detecting Irregularities in Images and in Video", *IEEE International Conference on Computer Vision*, 2005.

[11] L. Chaisorn, T-S. Chua and C-H. Lee, "The Segmentation of News Video Into Story Units", *International Conference on Multimedia and Expo*, 2002.

[12] S.F. Chang, W. Chen, H. Meng, H. Sundaram and D. Zhong, "Video Q: An Automated Content-Based Video Search System Using Visual Ques", *ACM International Conference on Multimedia*, 1997.

[13] L-Q. Chen, X. Xie, W-Y. Ma, H.J. Zhang and H-Q. Zhou, "Image Adaptation Based on Attention Model for Small-Form-Factor Devices", *International Conference on Multimedia Modelling*, 2003.

[14] Y.X. Chen and J.Z. Wang, "A Region-Based Fuzzy Feature Matching Approach to Content-Based Image Retrieval", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(9):1252–1267, 2002.

[15] W-H. Cheng, W-T. Chu, J-H. Kuo and J-L. Wu, "Automatic Video Region-of-Interest Determination Based on User Attention Model", *International Symposium on Circuits and Systems*, 2005.

[16] W.G. Cheng and D. Xu, "Content-Based Video Retrieval Using Shot Cluster Tree", *International Conference on Machine Learning and Cybermetics*, 2003.

[17] S.S. Cheung and A. Zakhor, "Efficient Video Similarity Measurement with Video Signature", *IEEE Transactions on Circuits and Systems for Video Technology*, vol.13, pages59–74, 2003.

[18] T.S. Chua and L.Q. Ruan, "A Video Retrieval and Sequencing System", *ACM Transactions on Information Systems*, vol.13, no.4, pages.373–407, 1995.

[19] F. Dellaert, S.M. Seitz, C.E. Thorpe and S. Thrun, "EM, MCMC, and Chain Flipping for Structure from Motion with Unknown Correspondence", *Machine Learning, special issue on Markov chain Monte Carlo methods*, 50: 45–71, 2003.

[20] M.E. Donderler, O. Ulusoy and U. Gudukbay, "Rule-Based Spatial-Temporal Query Processing for Video Databases", *Very Large Data Bases Journal*, Vol. 13, No. 1, 88–103, January 2004.

[21] J.A. Driscoll, R.A. Peters II and K.R. Cave, "A Visual Attention Network for A Humanoid Robot", *International Conference on Intelligent Robots and Systems*, 1998.

[22] J. Duncan and G.W. Humphreys, "Visual Search and Stimulus Similarity", *Psychological Review*, 96(3):433–458, 1989.

[23] B. Erol and F. Kossentini, "Similarity Matching of Arbitrary Shaped Video by Still Shape Features and Shape Deformations", *IEEE International Conference on Image Processing*, 2001.

[24] W.E. Farag and H.A. Wahab, "Video Content-Based Retrieval Techniques", *Multimedia Systems and Content-Based Image Retrieval*, Chapter VI, Idea Group Publishing, 2004.

[25] C. Fellbaum, "WordNet: An Electronic Lexical Database", *MIT Press*, May, 1998.

[26] L.L. Galdino and D.L. Borges, "A Visual Attention Model for Tracking Regions Based on Color Correlograms", *Brazilian Symposium on Computer Graphics and Image Processing*, 2000.

[27] J.L. Gauvain, L. Lamel and G. Adda, "The LIMSI Broadcast News Transcription System", *Speech Communication*, 37(1–2):89–108, 2002.

[28] S. Geman and D. Geman, "Stochastic Relaxation, Gibbs Distributions, and Bayesian Restoration of Images", *IEEE Transations on Pattern Analysis and Machine Intelligence*, Vol. 6, 721–741, 1984.

[29] P. Green, "Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination", *Biometrika*, 82, 711–732, 1995.

[30] U. Grenander and M.I. Miller, "Representation of Knowledge in Complex Systems", *Journal of Royal Statistical Society*, Series B, Vol. 56, No. 4, 1994.

[31] B. Gunsel, A. Ferman and A. Tekalp, "Temporal Video Segmentation Using Unsupervised Clustering and Semantic Object Tracking", *Journal of Electronic Imageing*, Vol. 7, No. 3, 1998.

[32] A. Hampapur and R.M. Bolle, "Comparison of Distance Measures for Video Copy Detection", *IEEE International Conference on Multimedia and Expo*, 2001.

[33] F. Han, Z.W. Tu and S.C. Zhu, "Range Image Segmentation by an Effective Jump-Diffusion Method", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 26, No. 9, 2004.

[34] J. Han, K.N. Ngan, M. Li and H.J. Zhang, "Towards Unsupervised Attention Object Extraction by Integrating Visual Attention and Object Growing", *International Conference on Image Processing*, 2004.

[35] A. Hanjalic, R.L. Lagendijk, and J. Biemond, "Automated High-Level Movie Segmentation for Advanced Video-Retrieval Systems", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 9, Issue 4, 1999.

[36] W.K. Hasting, "Monte Carlo Sampling Methods Using Markov Chains and Their Applications", *Biometrika*, 57:97-109, 1970.

[37] T.C. Hoad and J. Zobel, "Fast Video Matching with Signature Alignment", *ACM SIGMM International Workshop on Multimedia Information Retrieval*, 2003.

[38] K. Hoashi, M. Sugano, M. Naito, K. Matsumoto, F. Sugaya and Y. Nakajima, "Shot Boundary Determination on MPEG Compressed Domain and Story Segmentation Experiments for TRECVID 2004", *TREC Video Retrieval Evaluation Forum*, 2004.

[39] W. Hsu and S.F. Chang, "Generative, Discriminative, and Ensemble Learning on Multi-Model Perceptual Fusion Toward News Video Story Segmentation", *International Conference on Multimedia and Expo*, 2004.

[40] Y. Hu, D. Rajan and L-T. Chia, "Adaptive Local Context Suppression of Multiple Cues for Salient Visual Attention Detection", *IEEE International Conference on Multimedia and Expo*, 2005.

[41] I. Ide, H. Mo, N. Katayama and S. Satoh, "Topic Threading for Structuring a Large-Scale News Video Archive", *International Conference on Image and Video Retrieval*, 2004.

[42] L. Itti, C. Koch and E. Niebur, "A Model of Saliency-Based Visual Attention for Rapid Scene Analysis", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 11, 1254–1259, November, 1998.

[43] O. Javed, S. Khan, Z. Rasheed, and M. Shah, "Visual Content Based Segmentation of Talk and Game Shows", *International Journal on Computer Applications*, 2002.

[44] F. Jing, M.J. Li, H-J Zhang, and B. Zhang, "An Effective Region-Based Image Retrieval Framework", *IEEE Transactions on Image Processing*, 13(5):699–709, 2004

[45] K. Kashino, T. Kurozumi and H. Murase, "A Quick Search Method for Audio and Video Signals Based on Histogram Prunning", *IEEE Transactions on Multimedia*, vol.5, no.3, pages.348–357, 2003.

[46] Z. Khan, T. Balch and F. Dellaert, "An MCMC-Based Particle Filter for Tracking Multiple Interacting Targets", *European Conference on Computer Vision*, 2004.

[47] J. Kender and M. Naphade, "Visual Concept for News Story Tracking: Analyzing and Exploiting the NIST TRECVID Video Annotation Experiment", *International Conference on Computer Vision and Pattern Recognition*, 2005.

[48] J.R. Kender and B.L. Yeo, "Video Scene Segmentation Via Continuous Video Coherence", *International Conference on Computer Vision and Pattern Recognition*, 1998.

[49] Y.T. Kim and T.S. Chua, "Retrieval of News Video Using Video Sequence Matching", *International Conference on Multimedia Modelling*, 2005.

[50] S.H. Kim and R.H. Park, "An Efficient Algorithm for Video Sequence Matching Using the Modified Hausdoff Distance and The Directed Divergence", *IEEE Transactions on Circuits and Systems for Video Technology*, vol.12, no.7, pages.592–596, 2002.

[51] C. Kim and B. Vasudev, "Spatiotemporal Sequence Matching for Efficient Video Copy Detection", *IEEE Transactions on Circuits and Systems for Video Technology*, vol.15, no.1, 2005.

[52] T. Kohonen, " A Computational Model of Visual Attention", *International Joint Conference on Neural Networks*, 2003.

[53] R. Lancini, F. Mapelli and A. Mucedero, "Automatic Identification of Compressed Video", *International Conference on Acoustics, Speech and Signal Processing*, 2004.

[54] Y. Li, S. Narayanan and C.C. Jay Kuo, "Movie Content Analysis Indexing and Skimming", *Video Mining*, Kluwer Academic Publishers, 2003.

[55] R. Lienhart, S. Pfeiffer, and W. Effelsberg, "Scene Determination Based on Video and Audio Features", *International Conference on Multimedia Computing and Systems*, 1999.

[56] T. Lin, C.W. Ngo, H.J. Zhang and Q.Y. Shi, "Integrating Color and Spatial Feature for Content-Based Video Retrieval", *International Conference on Image Processing*, 2001.

[57] J.G. Liu, Y. Zhai and M. Shah, "PEGASUS: An Information Mining System for TV News Videos", *SPIE Defense and Security Symposium*, 2006.

[58] X. Liu, Y. Zhuang and Y. Pan, "A NewApproach to Retrieve Video by Example Video Clip", *ACM International Conference on Multimedia*, 1999.

[59] D.G. Lowe, "Distinctive Image Features From Scale-Invariant Keypoints", *International Journal of Computer Vision*, 60(2): 91–110. 2004.

[60] Z. Lu, W. Lin, X. Yang, E.P. Ong and S. Yao, "Modeling Visual Attentions Modulatory Aftereffects on Visual Sensitivity and Quality Evaluation", *IEEE Transactions on Image Processing*, Vol. 14, No. 11, November, 2005.

[61] Y. Luo, J-N. Hwang and T-D. Wu, "Object-Based Video Analysis and Interpretation", *Multimedia Systems and Content-Based Image Retrieval*, Chapter VIII, Idea Group Publishing, 2004.

[62] Y.F. Ma and H.J. Zhang, "Contrast-Based Image Attention Analysis by Using Fuzzy Growing", *ACM Conference on Multimedia*, 2003.

[63] O. Marques and B. Furht, "Content-Based Image and Video Retrieval", *Kluwer Academic Publisher*, 2002.

[64] O. Le Meur, D. Thoreau, P. Le Callet and D. Barba, "A Spatio-Temporal Model of the Selective Human Visual Attention", *International Conference on Image Processing*, 2005.

[65] R. Milanese, H. Wechsler, S. Gill, J.M. Bost and T. Pun, "Integration of Bottom-Up and Top-Down Cues for Visual Attention Using Non-Linear Relaxation ", *International Conference on Computer Vision and Pattern Recognition*, 1994.

[66] R. Mohan, "Video Sequence Matching", *International Conference on Acoustics, Speech and Signal Processing*, 1998.

[67] M. Naphade, M. Yeung and W. Xiong, "A Novel Scheme for Fast and Efficient Video Sequence Matching Using Compact Signatures", *SPIE Conference on Storage and Retrieval for Media Databases*, 2002.

[68] A.Y. Ng, M.I. Jordan and Y. Weiss, "On Spectral Clustering: Analysis and an Algorithm", *Annual Conference on Neural Information Processing Systems*, 2002.

[69] C.W. Ngo, H.J. Zhang, R.T. Chin, and T.C. Pong, "Motion-Based Video Representation for Scene Change Detection", *International Journal on Computer Vision*, 2001.

[70] C.W. Ngo, T.C. Pong and H.J. Zhang, "On Clustering and Retrieval of Video Shots Through Temporal Slices Analysis", *IEEE Transactions on Multimedia*, Vol. 4, No. 4, 2002.

[71] A. Nguyen, V. Chandrun and S. Sridharan, "Visual Attention Based ROI Maps From Gaze Tracking Data", *International Conference on Image Processing*, 2004.

[72] People's Daily, http://english.people.com.cn/other/archive.html.

[73] J.M. Odobez, D.G. Perez and M. Guillemot, "Video Shot Clustering Using Spectral Methods", *International Workshop on Content-Based Multimedia Indexing*, 2003.

[74] A. Oliva, A. Torralba, M.S. Castelhano and J.M. Henderson, "Top-Down Control of Visual Attention in Object Detection", *International Conference on Image Processing*, 2003.

[75] N. Ouerhani and H. Hugli, "Computing Visual Attention from Scene Depth", *International Conference on Pattern Recognition*, 2000.

[76] N. Ouerhani, J. Bracamonte, H. Hugli, M. Ansorge and F. Pellandini, "Adaptive Color Image Compression Based on Visual Attention", *International Conference on Image Analysis and Processing*, 2001.

[77] O. Oyekoya and F. Stentiford, "Exploring Human Eye Behaviour Using A Model of Visual Attention", *International Conference on Pattern Recognition*, 2004.

[78] Y. Peng and C.W. Ngo, "Clip-Based Similarity Measure for Hierarchical Video Retrieval", *ACM SIGMM International Workshop on Multimedia Information Retrieval*, 2004.

[79] C. Peters and C.O. Sullivan, "Bottom-Up Visual Attention for Virtual Human Animation", *International Conference on Computer Animation and Social Agents*, 2003.

[80] M. Petkovic and W. Jonker, "Content-Based Video Retrieval: A Database Perspective", *Kluwer Academic Publisher*, 2004.

[81] S. Pfeiffer and U. Srinivasan, "Scene Determination Using Auditive Segmentation", *Media Computing: Computational Media Aesthetics*, Kluwer Academic Publisher, 2002.

[82] D.B. Phillips and A.F.M. Smith, "Bayesian Model Comparison via Jump Diffusion", *Markov Chain Monte Carlo in Practice*, Ch. 13, Chapman and Hall, 1995.

[83] K. Rapantzikos, N. Tsapatsoulis and Y. Avrithis, "Spatiotemporal Visual Attention Architecture for Video Analysis", *IEEE Workshop on Multimedia Signal Processing*, 2004.

[84] Z. Rasheed, M. Shah, "Scene Detection In Hollywood Movies and TV Shows", *International Conference on Computer Vision and Pattern Recognition*, 2003.

[85] Z. Rasheed, M. Shah, "Movie Genre Classification by Exploiting Audio-Visual Features of Previews", *International Conference on Pattern Recognition*, 2002.

[86] P. Sand and S. Teller, "Video Matching", *ACM Transactions on Graphics*, 2004.

[87] F. Schaffalitzky and A. Zisserman, "Automated Scene Matching in Movies", *International Conference on Image and Video Retrieval*, 2002.

[88] J. Senegas, "A Markov Chain Monte Carlo Approach to Stereovision", *European Conference on Computer Vision*, 2002.

[89] Y. Sheikh, Y. Zhai and M. Shah, "An Accumulative Framework for the Alignment of An Image Sequence", *Asian Conference on Computer Vision*, 2004.

[90] Y. Sheikh, K. Shafique, Y. Zhai and M. Shah, "Visual Monitoring of Railroad Grade Crossings", *SPIE Defense and Security Symposium*, 2004.

[91] J. Sivic, F. Schaffalitzky and A. Zisserman, "Object Level Grouping for Video Shots", *European Conference on Computer Vision*, 2003.

[92] H. Sundaram and S.F. Chang, "Video Scene Segmentation Using Video and Audio Features", *IEEE International Conference on Multimedia and Expo*, 2000.

[93] F. Stentiford, "A Visual Attention Estimator Applied to Image Subject Enhancement and Colour and Grey Level Compression", *International Conference on Pattern Recognition*, 2004.

[94] Y. Tan, S. Kulkarni and P. Ramadge, "A Fraemwork for Measuring Video Similarity and Its Application to Video Query by Example", *International Conference on Image Processing*, 1999.

[95] W. Tavanapong and J.Y. Zhou, "Shot Clustering Techniques for Story Browsing", *IEEE Transactions on Multimedia*, 2004.

[96] TRECVID 2004 Guidelines, http://www-nlpir.nist.gov/projects/tv2004/tv2004.html.

[97] A. Treisman and G. Gelade, "A Feature-Integration Theory of Attention", *Cognitive Psychology*, 12:97–136, 1980.

[98] Z.W. Tu and S.C. Zhu, "Image Segmentation by Data Driven Markov Chain Monte Carlo", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 5, May 2002.

[99] P. Viola and M. Jones, "Robust Real-Time Object Detection", *International Journal on Computer Vision*, 2001.

[100] Webster Online, http://www.m-w.com.

[101] M. Yeung, B. Yeo, and B. Liu, "Segmentation of Videos by Clustering and Graph Analysis", *Computer Vision and Image Understanding*, vol. 71, no. 1, 94–109, July 1998.

[102] M.M. Yeung, B.L. Yeo, W. Wolf and B. Liu, "Video Browsing Using Clustering and Scene Transitions on Compressed Sequences", *SPIE Conference on Multimedia Computing and Networking*, 1995.

[103] A. Yoshitaka, T. Ishii, M. Hirakawa and T. Ichikawa, "Content-Based Retrieval of Video Data by the Grammar of Film", *IEEE Symposium on Visual Languages*, 1997.

[104] Y. Zhai, X.C. Cao, Y.J. Zhang, O. Javed, F. Rafi, S. Ali, O. Alatas, S. Khan and M. Shah, "University of Central Florida at TRECVID 2004", *TREC Video Retrieval Evaluation Forum*, 2004.

[105] Y. Zhai, J.G. Liu and M. Shah, "Automatic Query Expansion in News Video Retrieval", *International Conference on Multimedia and Expo*, Toronto, July 9-12, 2006.

[106] Y. Zhai, J. Liu, X.C. Cao, A. Basharat, A. Hakeem, S. Ali, M. Shah, C. Grana and R. Cucchiarra, "Video Understanding and Content-Based Retrieval", *TREC Video Retrieval Evaluation Forum*, 2005.

[107] Y. Zhai, Z. Rasheed and M. Shah, "Semantic Classification of Movie Scenes Using Finite State Machines", *IEE Proceedings on Vision, Image and Signal Processing*, Vol.152, No.6, pp.896-901, 2005.

[108] Y. Zhai, Z. Rasheed and M. Shah, "A Framework for Semantic Classification of Scenes Using Finite State Machines", *International Conference on Image and Video Retrieval*, 2004.

[109] Y. Zhai, Z. Rasheed and M. Shah, "Conversation Detection in Feature Films Using Finite State Machines", *International Conference on Pattern Recognition*, 2004.

[110] Y. Zhai, Z. Rasheed and M. Shah, "University of Central Florida at TRECVID 2003", *TREC Video Retreival Evaluation Forum*, 2003.

[111] Y. Zhai, K. Shafique, A. Vartak, P. Berkowitz and M. Shah, "Multiple Vehicle Tracking in Surveillance Videos", *CLEAR'06 Evaluation Campaign and Workshop*, 2006.

[112] Y. Zhai and M. Shah, "Visual Attention Detection in Video Sequences Using Spatiotemporal Cues", *ACM International Conference on Multimedia*, 2006.

[113] Y. Zhai and M. Shah, "Video Scene Segmentation Using Markov Chain Monte Carlo", *IEEE Transactions on Multimedia*, 2006.

[114] Y. Zhai and M. Shah, "Tracking News Stories Across Different Sources", *ACM International Conference on Multimedia*, 2005.

[115] Y. Zhai and M. Shah, "Determining Structure in Continuously Recorded Videos", *ACM Annual Conference on Multimedia*, 2005.

[116] Y. Zhai and M. Shah, "A General Framework for Temporal Video Scene Segmentation", *International Conference on Computer Vision*, 2005.

[117] Y. Zhai and M. Shah, "Automatic Segmentation of Home Videos", *International Conference on Multimedia and Expo*, 2005.

[118] Y. Zhai and M. Shah, "A Multi-Level Framework for Video Shot Structuring", *International Conference on Image Analysis and Recognition*, 2005.

[119] Y. Zhai, A. Yilmaz and M. Shah, "Story Segmentation in News Videos Using Visual and Text Cues", *International Conference on Image and Video Retrieval*, 2005.

[120] D-Q. Zhang, C-Y. Lin, S-F Chang and J.R. Smith, "Semantic Video Clustering Across Sources Using Bipartitie Spectral Clustering", *IEEE International Conference on Multimedia and Expo*, 2004.

[121] Y.J. Zhang, Y.Y Gao and Y. Luo, "Object-Based Techniques for Image Retrieval", *Multimedia Systems and Content-Based Image Retrieval*, Chapter VII, Idea Group Publishing, 2004.

[122] H.J. Zhang, J. Wu, D. Zhong and S.W. Somaliar, "An Integrated System for Content-Based Video Retrieval and Browsing", *Pattern Recognition*, Vol. 30, No. 4, 1997.

[123] L. Zhao, W. Qi, S.Z. Li, S.Q. Yang and H.J. Zhang, "Content-Based Retrieval of Video Shot Using the Improved Nearest Feature Line Method", *International Conference on Acoustics, Speech and Signal Processing*, 2001.

[124] J. Zhou and X.P. Zhang, "Automatic Identification of Digital Video Based on Shot Level Sequence Similarity", *ACM Conference on Multimedia*, 2005.

[125] S.C. Zhu and A.L. Yuille, "Region Competition: Unifying Snakes, Region Growing, and Bayes/MDL for Multiband Image Segmentation", *IEEE Transations on Pattern Analysis and Machine Intelligence*, Vol.18, No.9, pp.884-900, 1996.