A FRAMEWORK FOR ASSOCIATED NEWS STORY RETRIEVAL

EHSAN YOUNESSIAN

School of Computer Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of

Doctor of Philosophy

2013

Abstract

Video retrieval – searching and retrieving videos relevant to a given query – is one of the most popular topics in both real life applications and multimedia research. Finding relevant video content is important for producers of television news, documentaries and commercials. Particularly, in news domain, hundreds of news stories in many different languages are being published everyday by the numerous news agencies and media houses. The huge number of published news stories brings enormous challenges in developing techniques for their efficient retrieval. In particular, there is the challenge of identifying two news clips that discuss the same story. Here, the visual information need not be similar enough for simple near-duplicate video detection algorithms to work. Although, visually two news stories might be different, they might be addressing the same main topic. We call such news stories as associated new stories and the main objective in this thesis is to identify such stories. Therefore, it is imperative that we resort to other modalities such as speech and text for robust retrieval of associated news stories.

In the visual domain, associated news stories can be seen as duplicate, near-duplicate, partially near-duplicate videos or in more challenging cases as videos sharing specific visual concepts (e.g. fire, storm, strike, etc). We study Near-Duplicate Keyframe (NDK) identification task as the main core of the visual analysis using different global and local features such as Scale-Invariant Feature Transformation (SIFT). We propose the Constraint Symmetric Matching scheme to match SIFT descriptors between two keyframes and also incorporate other features such as color to tackle the NDK detection task. Next, we

cluster keyframes within a news story if they are NDKs and generate a novel scene-level video signature, called scene signature, for each NDK cluster. A scene signature is essentially a Bag-of-SIFT containing both common and distinct visual cues within an NDK cluster and is more compact and discriminative compared to the keyframe-level local feature representation. In addition to scene signature, we generate a visual semantic signature for a news video which is a 374-dimensional feature indicating the probability of the presence of the predefined visual concepts in a news story. We integrate these two sources of visual knowledge (i.e. scene signature and semantic signature) to determine enhanced visual content similarity between two stories.

In the textual domain, associated news stories usually have common spoken words (by anchor or reporter) and/or displayed words (appear as a closed caption) which can be extracted through Automatic Speech Recognition (ASR) and Optical Character Recognition (OCR), respectively. Since OCR transcripts usually have high error rate, we propose a novel post-processing approach based on the local dictionary idea to recover the erroneous OCR output and identify more informative words, called keywords. We generate an enhanced textual content representation using ASR transcript and OCR keywords through an early fusion scheme. We also employ textual semantic similarity to measure the relatedness of the textual features.

Finally, we incorporate all enhanced textual and visual representations/similarities through an early/late fusion scheme, respectively, to investigate their complementary role in the associated news story retrieval task. In the proposed early fusion, we retrieve visual semantics, determined as the visual semantic signature, using textual information provided by ASR and OCR. In the late fusion, we combine enhanced textual and visual content similarities and early fusion similarity through a learning process to boost the retrieval performance.

We evaluate the proposed NDK retrieval, detection and clustering approaches in extensive experiments on standard datasets. We also assess the effectiveness and compactness of the proposed scene signature to represent a video compared to other local and global video signatures using a web video dataset. Finally, we show the usefulness of multi-modal approaches using different textual and visual modalities to retrieve associated news stories.

To my lovely family

Acknowledgements

Foremost, I would like to express my sincere thanks and appreciation to my supervisor, Dr. Deepu Rajan. I have benefited tremendously from his invaluable advice. His guidance helped me in all the time of research and writing of this thesis. Beside my advisor, I would like to thank Prof. Alexander Hauptmann for offering me the exchange program opportunity at Informedia lab at Carnegie Mellon University and leading me working on diverse exciting projects. I am very grateful for his constant encouragement and his perspectives on life at large. My sincere thanks also goes to Dr. Xavier Anguera and Dr. Tomasz Adamek for offering me the internship program at Telefónica I+D company and providing a stimulating and fun environment in which to learn and grow. Their assistance allowed me to pursue my interests and explore new research areas.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Video retrieval – searching and retrieving videos relevant to a given query – is one of the most popular topics in both multimedia research and real life applications. Finding relevant video content is important for content producers (e.g. producers of commercials, documentaries, and television news), content distributors and consumers. For instance, detecting video copy to control the copyright of the large number of video clips uploaded everyday is a critical issue for the owner of popular video sharing websites. For news and documentary producers who are dealing with a vast amount of professional video archives, video retrieval/search tools play a critical role since it costs less to re-use content than to reshoot; in fact when dealing with content of a historical nature it might be impossible to reshoot. On the other hand video sharing on web video servers is exponentially growing which creates perhaps the most diverse and the largest publicly available video archive [5]. Finding the videos of interest is becoming harder and harder everyday for the users. Research on video retrieval aims to address this task.

In the early nineties research on content-based image retrieval began to develop algorithms facilitating the automatic search of image corpora by their con-

tent without using costly manual annotation [44]. Later scholars began studying content-based video retrieval systems with some of the early image retrieval techniques being adapted and other new video retrieval systems being developed [86]. The early video retrieval research lacked the solid and controlled retrieval experiments on standard test collections and retrieval/search tasks that was the foundation of most of the research in the text-based information retrieval domain. Early research on the visual retrieval methods was often conducted on collections of images from homogeneous categories which does not fit video retrieval problem well. In the last couple of years there has been a collaborative effort in the video retrieval research community through the TRECVID competition to facilitate controlled video retrieval experiments on reasonable sized common video collections [14, 52, 102].

In this research, we particularly focus on broadcast news videos and aim to study associated news story retrieval task. Following the video retrieval research community, we also assess the quality of the proposed approaches using TRECVID video collections.

1.1 Associated News Story Retrieval

Since the launch of 24-hour news channels, there has been an explosion of television news content all over the world in numerous languages broadcasting many news stories each day. These stories could be in the form of an interview, breaking news, pre-recorded segments, or live broadcast of an event. Among these huge volumes of news stories from various channels, there exists a great deal of overlap in the semantic sense, i.e., they address the same main topic. We refer to such

news stories as *associated news stories*. Examples of associated news stories are shown in Figure 1.1. Figure 1.1(a) shows two stories about "Trial of Saddam Hussein" published by CNN and MSNBC channels. In Figure 1.1(b), there are three news stories from ABC, CCTV and CNN channels discussing the same topic of "Bush press conference". Figure 1.1(c) depicts two news stories from NBC and MSNBC referring to the "Fire in Oklahoma" event. It is apparent that associated news stories are different from video copies, which may not be the exact duplicate of each other but may be transformed (e.g. with different encoding parameters) or are modified versions of the original document [121]. It is also different from (partially) near-duplicate videos, which could be close to exact duplicates of each other while differing in other aspects like editing operations, camera setting or different length and temporal order [108, 110]. In fact, video copies and near-duplicate videos belong to a class of associated news stories, as we explain later in Section 1.3.

In associated news story retrieval, we address the problem of unconstrained video similarity which can be applied as a prior stage for other tasks like event-based information organization, topic detection and tracking (TDT), news story summarization, news story clustering, novelty and redundancy detection in news stories, event threading, and auto-documentary. In all mentioned tasks, an effective video representation, containing low/mid/high-level multi-modal features, and a proper similarity measure are foundations to designing a multimedia system. In this thesis, we investigate these two issues in the context of associated news story retrieval.

Figure 1.1: Associated news story categories — Example stories for each category, indicating (a) trail of Saddam Hussein, (b) Bush press conference, and (c) fire in Oklahoma.

1.2 Multiple Modalities in News Videos

The news program is one of the most popular and viewed programs on television. Media houses are moving from traditional newspapers and posters towards electronic means for news dissemination. Most news agencies have their websites to broadcast news products in different formats like text, audio or video articles. They also have established active accounts in multimedia sharing web sites like YouTube [5] and Twitter [3] and use their popularity to attract more audiences surfing the World Wide Web.

Broadcast video stories (articles) can be written in *packages*, *readers*, *voice overs*, and *sound on tape* [19]. A *package* is an edited set of video clips for a news story and is common on television. It is narrated typically by a reporter and has audio, video, graphics and video effects. The anchor usually reads a "lead in" (introduction) before the package is aired and may conclude the story with additional information, called a tag. A *reader* is an article read without accompanying video or sound. Sometimes an "over the shoulder graphic" is added. A *voice over*, or *VO*, is a video article narrated by an anchor. *Sound on tape*, or *SOT*, is sound and/or video, usually recorded in the field. It is usually an interview or *soundbite* which is a short piece of actual sound from the event reported on. The term is also used for the section of video that accompanies the audio.

These broadcast news story types include enriched auditory, textual and visual cues which can be utilized for news story retrieval. All kinds of news stories (specially *reader* and *SOT*) are basically accompanied with words, uttered by anchor persons, reporters, commentators, interviewees etc. that carry the signif-

icant part of semantics. Accordingly, applying ASR and retrieving spoken words would be essential for the story retrieval task. In *soundbite*, we may take advantage of powerful acoustic features through which one can determine different acoustic labels like music, singing, human or animal sounds etc. for acoustic segmentation of a news video. Further, most of the news stories also come with some textual information which mainly appears as the caption on the bottom of the screen indicating the headline of the news story and some briefs, called *kicker*, to grab the reader's attention, or as the *cutline* [19] to name and describe pictures or person(s) speaking. These valuable textual cues, which are highly related to the news story, can be extracted through Optical Character Recognition (OCR) and can be used for news story retrieval. In addition to ASR and OCR transcripts, visual elements play a critical role (mainly for *package* or *voice over*), especially since humans receive much of their information of the world through their sense of vision. In addition, metadata like time duration, tag words, thumbnails etc. have also been used for video retrieval and re-ranking tasks [113].

It is clear that multiple modalities must be exploited in order to develop an effective video retrieval system. This is all the more important for news videos in particular, since they are more structured vis-a-vis the different modalities which leads to a robust associated news story retrieval algorithm.

1.3 Categories of Associated News Stories

We categorize associated news stories into three categories according to the visual difference in their video footage.

The first category includes associated news stories sharing the same video

footage from a few big press agencies. They are referred to as co-derivative videos [55]. Co-derivative videos are considered to be unique versions of a single video production (e.g. television, theatrical, and trailer versions of a film would be considered co-derivative). In the domain of news, a news story may typically be lengthened or shortened by inserting or deleting part of the video material. Although the same material is presented, specific post-processing like inserting logo, banners or graphic layout, showing the items in the picture-in-picture format, screen windowing, gamma correction, cropping and other transformations may be applied to the raw footage so as to customize the look for a specific channel. Figure 1.1(a) shows two such associated news stories S1 and S2. Finding this type of associated news story has been investigated extensively under the domain of "video copy detection" or "near-duplicate video detection", some of which we will review in the next chapter.

The second category of associated news stories has different video footages but they cover the same news object in usually the same venue and within the same context. For instance, news stories addressing a pseudo-event like a press conference belong to this category (Figure 1.1(b)). A pseudo-event is an event or activity that exists for the sole purpose of media publicity and serves little or no other function in real life. Without the press, nothing meaningful actually occurs at the event [19]. As we can see from Figure 1.1(b), finding Near-Duplicate Keyframes (NDK) across associated news stories from this category is more problematic than from the first category due to the high degree of variation of camera angles, significant object displacement, and variations of camera lens settings.

The third category consists of associated news stories which address an event which is not visually related to specific objects or occurs at a specific venue, e.g.

news story addressing "Katrina hurricane" or "Fire in Oklahoma" in Figure 1.1(c). Visually, the hurricane or fire will be similar irrespective of where it happens. It could be the case that the same news story might be reported from different locations so that there is very little visual similarity between them. This type of associated stories cannot be usually detected solely based on common and low-level visual cues. Using high-level visual features like semantic visual concepts can be beneficial in this case. Moreover, other modalities like textual and auditory information could possibly be more informative and discriminative.

1.4 Challenges

To develop an effective video retrieval system, two fundamental problems of video representation and video similarity measure must be addressed. In the former, there are different challenges to represent a video using different modalities. In visual modality, an effective video representation should be robust against possible variations in lighting, object displacement, edit setting, camera setting, etc. It should also be capable of capturing the semantic visual concepts depicted in videos since the connection between some similar videos is beyond the specific object or place and is about the mutual visual concepts they share. On the other hand, in the textual domain, the main challenge to represent textual information of a video is the accurate extraction of spoken words from audio channel and written words from video. The quality of ASR output is highly related to the speaker and recording environment. The complex and transparent backgrounds, and different fonts and sizes of the letters, displayed on the screen, can decrease the accuracy of the OCR transcript.

Given video representations, an effective video similarity measure is required to distinguish semantically similar videos. However, there are some semantic gaps in different modalities which should be bridged. In the textual domain, there is a significant number of incorrectly recognized words in the extracted ASR/OCR transcripts. Moreover, different news broadcasters could consistently introduce bias in reporting political and social issues because producers, editors, and reporters collectively make similar decisions based on shared values and beliefs. This bias can highly affect many decisions,e.g., what to cover, what to show on a screen, and what and how to say. That explains why different news broadcasters do not necessarily use the same words to report an identical event. Therefore, their original textual representation can be significantly different, particularly in non-English news where machine translation is used to generate the English version of the extracted ASR transcript. In this case, the textual semantic similarity metric can play a critical role to measure the textual similarity between stories more effectively. In the visual domain, the main challenge is how to integrate different global and local visual similarities to obtain a more semantic similarity measure which is close to human perception. A single visual similarity measure may not perform well on all associated news stories, while by fusing different visual representations/similarities, retrieval performance can be improved. The final problem in determining video similarity measure is how to fuse visual and textual similarities in a way that they can play a complementary role to each other so that we get meaningful visual similarity when textual similarity measure fails to capture the semantic relations between videos, and vice-versa.

1.5 Objectives

In this research, we investigate different modalities to retrieve all three categories of associated news stories described in Section 1.3. Note that the usefulness and capabilities of different modalities differ for different categories. However, it is clear that a practical news video retrieval method will benefit from considering all different modalities into account. The objectives of this research are:

• To investigate visual similarity across news stories through semantic near-duplicate analysis between keyframes/scenes.

• To investigate textual similarity via textual features derived from ASR/OCR transcripts.

• To incorporate the concept-based indexing and retrieval method to improve news video retrieval performance.

• To explore effective early and late fusion schemes to integrate different modalities in feature-level or decision-level, respectively.

1.6 Contributions

The main contributions of this research are as follows:

• Near-Duplicate Keyframe retrieval/detection — we develop an algorithm for NDK retrieval and detection despite the slight to moderate degree of variations caused by lighting, viewpoint, acquisition time, motion, and editing effects. For retrieval, keyframes similar to the query keyframe are ranked while for detection, we determine a threshold based on which keyframe pairs are classified as NDK or non-NDK.

• Scene signature — we propose a compact and effective video signature, called

scene signature, extracted at the scene-level using mutual and distinct visual cues inherent within NDK clusters in a news story.

• Textual semantic similarity — we propose a novel method to measure textual semantic similarity between enhanced textual representations of two news stories using WordNet-based and Wikipedia-based semantic similarities. The enhanced textual representation is obtained by refining the OCR output of video text using a local dictionary that is generated from the corresponding ASR transcript.

• Visual semantic signature — In addition to scene signature, we incorporate another visual representation, called semantic signature. It is essentially the probability of the presence of predefined visual concepts in a news story. We integrate scene and semantic signature similarities to boost the retrieval performance especially when scene signature fails to capture visual similarity.

• Visual concept signature — we utilize textual information of a news story, including ASR and refined OCR transcripts, to generate the visual concept signature which is basically the projection of the enhanced textual representation to a predefined visual concept list using textual semantic similarity.

1.7 Framework for Associated News Story Retrieval

We illustrate our framework for associated news story retrieval in Figure 1.2. It utilizes audio, visual and textual modalities. However, audio is converted to text through ASR for further processing, instead of processing directly in the audio domain. Hence, our framework broadly uses the visual and textual modalities each of which is briefly described below.

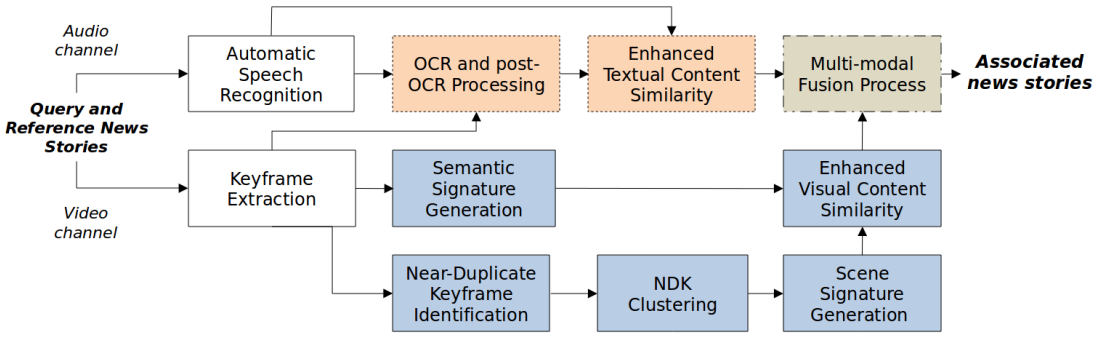

Figure 1.2: The general framework for associated news story retrieval — White blocks refer to the sampling and feature extraction modules. Pink and blue blocks refer to the textual and visual analyses, respectively. The gray block indicates multi-modal fusion of enhanced textual and visual similarities.

1.7.1 Visual Modalities

We develop two content-based representations for broadcast news story called scene signature and semantic signature. A scene signature is obtained by first clustering keyframes within a news story using our proposed NDK detection algorithm. Then, for each cluster of keyframes, we generate a scene signature, with respect to common as well as distinct visual information in the keyframes. Our purpose is to develop a scene signature that is more compact as well as more semantically representative than keyframe-level representation. The semantic signature is based on the probabilities of the presence of predefined visual concepts in a news story. We integrate the scene and semantic signature similarities to obtain a measure of visual content similarity between news stories.

As seen in Figure 1.2, the study of visual modalities consists of four parts — NDK identification, NDK clustering, scene signature generation, and enhanced visual content similarity.

1.7.1.1 NDK Identification

We develop a novel NDK identification method to detect all NDK pairs between two stories or within a story. Our proposed method can handle extreme zooming and significant object motion which are major challenges in identifying near-duplicate keyframes. To do so, we utilize the number of matching keypoints between keyframes as the main feature and also consider new color-based features. For NDK detection, we investigate the unbalanced classification problem to obtain the best separating boundary since existing datasets include much more negative data samples than positive samples. Particularly, for each NDK pair in a dataset with n keyframes, we can have $2 \times (n-1)$ non-NDK pairs. In such a case, a classifier can detect all pairs as non-NDk and still obtain a high accuracy, which is not desired. To avoid that, we need to moderate the role of non-NDKs, when we learn a classifier.

1.7.1.2 NDK Clustering

We propose an effective content-based clustering method for keyframes of news video stories using the developed NDK identification module. The set of keyframes is represented in a graph structure where each keyframe is a vertex and the edge weight between a pair of keyframes is the probability of the pair being an NDK. We adopt spectral clustering scheme to remove outlier keyframes and group the remaining keyframes. We also choose a proper number of clusters based on two criteria — within-cluster similarity and the eigengap heuristic — through which we try to obtain non-singleton and well-separated NDK clusters.

1.7.1.3 Scene Signature Generation

We propose a novel video signature called scene signature which can be applicable for variety of tasks in unconstrained news video domain. The same news stories that originate from different channels can appear with different layouts, lengths, temporal order, additional visual content and so on. This can significantly affect the effectiveness and robustness of existing video signatures. We aim to represent the visual cues appearing in a news story scene in a compact and comprehensive manner. To this end, an initial scene signature is obtained for each NDK cluster within a news story using the most informative common and distinct visual cues. A scene signature is actually a collection of SIFT descriptors. Compared to conventional keypoint-trajectory-based signatures, we take the co-occurrence of SIFT keypoints into account. This is beneficial especially when we deal with picture-in-picture, split screen, or long shots with significant object/camera movements. Next, through three steps of refinements on the initial scene signature we reduce the semantic gap to obtain the final scene signature which is more compact and semantically meaningful.

1.7.1.4 Enhanced Visual Content Similarity

The visual similarity between associated news stories with slight to moderate degree of variation like in Figure 1.1(a) and (b) can be measured based on global and local features. However, high degree of variation such as significant object displacement or change in camera setting can fail with low/mid-level visual features. Moreover, global and local signatures fail to link the associated news stories from the third category like in Figure 1.1(c) where the similarity between stories can be captured better through shared visual concepts like fire, firefighter, natural

disaster and so on rather than a specific object or location. To address this issue, we incorporate another high-level visual feature, called semantic signature, which is the probabilities of the presence of predefined visual concepts within a news video. Although semantic signature suffers from limited number of visual concept detectors trained and their relatively low distinguishing ability, it can be useful as a complementary signature in addition to other global and local signatures. Accordingly, we investigate failures of local signature similarity and incorporate semantic signature similarity with higher weight in such cases to improve retrieval performance.

1.7.2 Textual Modalities

We develop a method to determine story-level textual content similarity using two textual sources — one is the ASR transcript of the speech signal and the other is the OCR transcript of the video text. Video OCR systems play a key role in video content analysis [54, 90]. Especially in broadcast news retrieval, taking overlay text into account helps to improve the search results [35, 54]. The OCR output is obtained from an OCR machine which takes in pre-processed video frames.

We carry out a simple but fast pre-processing procedure followed by a novel OCR post-processing method using the concept of local dictionary to effectively correct erroneous OCR outputs and also to weight more informative OCR outputs. The modified OCR data is used together with the ASR transcript to generate keywords that are specific to a particular story.

Next, we use textual semantic similarity to measure the similarity between enhanced textual representations, which are the union of the modified OCR transcripts and ASR transcripts for each news story. Using the developed textual se-

mantic similarity, we can capture relatedness of words like "fire" and "firefighter", or "taxi" and "cab" more effectively, compared to the conventional Vector Space Model (VSM) method [41] in which only identical words are matched. We utilize the well-known WordNet-based [80] and Wikipedia-based [4] semantic similarities and also investigate how to refine them to avoid noisy connections. For example, some general terms like "make" or "act" can be connected to a wide range of words and degrade the quality of the semantic similarities.

1.7.3 Multi-modal Fusion Process

First, we investigate the early fusion approach for news story retrieval where we aim to retrieve visual semantic signatures using textual information. Although most news stories contain significant amount of textual information extracted via ASR and OCR, it often happens that some stories do not have representative visual cues which can be extracted using local and semantic signatures. Especially in *reader*, which is a type of news story read without accompanying video or sound, there is no relevant visual cue to the topic of interest. This fact leads to poor retrieval results using visual representation only. This observation motivates us to generate a visual concept signature, by mapping enhanced textual representation (i.e. the ASR and refined OCR transcripts) to the visual concept list, and use it to retrieve semantic signatures of associated news stories. We adopt two methods, namely Semantic Similarity Mapping (SSM) and Canonical Correlation Analysis (CCA), for direct and indirect mapping of textual representation to visual concept list, respectively. In SSM method, we use the developed textual semantic similarity, explained in Section 1.7.2, to map the enhanced textual representation to the visual concept list. In the CCA method, we learn the

co-occurrence of the words and visual concepts in the training data to determine the projection functions through which we can project the textual representation of a query story and visual semantic signatures of other stories to a third feature space wherein they are comparable. We determine the final early fusion result by integrating SSM and CCA similarity scores through a linear late fusion scheme with equal weights.

Finally, we also employ different late fusion strategies such as Ranked List and SVM-based fusion methods to integrate textual, visual and early fusion similarity scores and determine the final decision to retrieve associated news stories. In other words, we investigate whether textual/visual modalities and their early fusion can provide complementary information for associated news story retrieval task.

1.7.4 Datasets

Here, we briefly describe the datasets used to evaluate the performance of the algorithms developed in the thesis. For NDK identification, we use Columbia [125] and NTU [117] datasets each of which includes 300 NDKs and 300 non-NDKs. For NDK clustering and associated news story retrieval, we use TRECVID 2006 [2] dataset that includes 30 successive days of news videos. It contains English, Chinese and Arabic daily news from MSNBC, CNN, NBC, PHOENIX, NTDTV, CCTV and LBC channels covering both domestic and global events. We utilize ASR, provided by TRECVID organization [2], and OCR transcripts, extracted using JOCR engine [8], for English news and translated ASR transcripts, provided by TRECVID organization [2], for Chinese and Arabic news.

Story boundaries are manually marked for each video. Each story is labeled based on its main topic to create the baseline for evaluation. Overall, there are

830 news stories out of which there are 296 associated news story pairs. For scene signature studies, we prepare a dataset containing 100 news stories from different channels downloaded from YouTube [5] in February 2011. They vary in duration from 2 to 5 minutes and cover world events.

1.8 Organization of the Thesis

The thesis is organized as follows. A literature review covering various aspect of visual modality, textual modality and multimodal fusion in the context of multimedia retrieval is presented in Chapter 2. Chapter 3 presents our proposed NDK identification and clustering methods. We introduce a novel method to address the keypoint matching problem. We also describe our spectral-clustering-based method to group keyframes within a news story into different clusters. Chapter 4 describes the method to generate a scene signature. The scene-to-scene similarity measure is developed and used to calculate similarity between stories. Chapter 5 explains how we refine the OCR outputs through an early fusion scheme using ASR transcript. We also employ the textual semantic similarity to measure the textual relatedness between stories more effectively. In Chapter 6, first we explain how to integrate local signature and semantic signature similarities effectively, to obtain the enhanced visual content similarity. Then, we study the early fusion approach to retrieve visual semantic signature of news stories using textual information. We report our system performance on associated news story retrieval task using enhanced visual and textual content similarities and early fusion similarity scores and assess their contributions in the final performance. Finally, conclusions and directions for future work are suggested in Chapter 7.

Chapter 2

Literature Review

In this chapter, we explain related works in content-based multimedia retrieval using visual and textual modalities, respectively. We also review recent research in multi-modal fusion.

2.1 Visual Modality

2.1.1 Near-Duplicate Keyframe Identification

Near-Duplicate Keyframes (NDK) are pairs of keyframes that are very similar to each other despite the slight to moderate degree of variations caused by lighting, camera viewpoint, acquisition time, motion, and editing effects. There are mainly three different types of near-duplicates, studied in the literature [97]. (i) Strict near-duplicates, which are from the same video footage used in different programs. In the image domain, it is often referred as Image Exact Detection (IED). (ii) Object duplicates, which are from different footages of the same objects or the same background. (iii) Scene duplicates, which is composed of different footages taking the same scene, the same event, at the same time, but from the different viewpoints, e.g., by different cameras, and possibly with temporal offsets.

NDKs, particularly scene duplicates, are commonly found in broadcast programs. The task of NDK detection involves identifying NDK pairs while that of NDK retrieval involves ranking all near-duplicates to an input query image. In general, most of the existing approaches for NDK identification can be divided into two categories: appearance-based methods and local feature based methods which are explained separately below.

2.1.1.1 Appearance-based Methods

The variations in motion, illumination, view point and editing effects bring about great challenges for the development of automatic NDK detection methods. Conventional image matching methods using low level features (e.g., color moment or histogram, texture, and edge) lack the discriminative power for NDK detection due to its incapacity to capture scene semantics and composition. These methods will likely fail because of the out-sized variations within the NDKs and also due to the high similarity among non-NDKs. These methods perform well in Image Exact-Duplicate (IED) detection [25, 50]. The semantic representation, such as the model vector system [31, 103] could be a promising technique for NDK detection. However, its drawbacks can be summarized in one point: the vector representation of concepts fails to capture spatial/attributed relations of individual concepts. Moreover, most content-based detectors require a large number of examples for learning. Part-based image similarity has been previously pursued using 2-D string [25] and composite region templates (CRTs) [104]. However string representation is difficult to accurately represent visual scenes with complicated visual features, while region-based representation is sensitive to the region segmentation error. Part-based approaches have also been used in object/scene

recognition [42] and scene generative model [61].

2.1.1.2 Local Feature-based Methods

Compared to global features, local features appear to be useful for NDK identification due to their distinctiveness and robustness to changes due to different transformations and operations. Local feature based methods are based on extracting interesting keypoints from the images and matching them to identify near-duplicate keyframes. In [115], generic image categorization approaches using "Bag-of-Words" model is applied to Image Near-Duplicate (IND) detection and retrieval. Under this model, images are treated as documents by assigning descriptors of local covariant regions to "visual words". Each image is then represented by a histogram of word frequency. The most critical problem of such methods is that the ambiguous visual words will introduce large number of false matches when each region is matched independently to others. Several methods have been proposed to improve this problem by capturing the spatial arrangement of visual words. Lazebnik et al. [67] extended the Pyramid Matching Kernel (PMK) [47] by incorporating the spatial information of regions. Two images are partitioned into increasingly fine sub-regions and PMK is used to compare corresponding sub-regions. This method implicitly assumes the correspondences between sub-regions, which is not scale or rotation invariant. Also by utilizing spatial information of local regions, Savarese et al. [98] used correlogram to measure the distribution of the proximities between all pairs of visual words and used it for category classification, but the method is not scale invariant. In [26], a multi-level spatial matching framework with two-stage matching is proposed to deal with spatial shifts and scale variation for the image near-duplicate

identification task. The block-based distance between image pairs is used and multiple alignment hypotheses are explored. In [28], the local dependencies in spatial-scale space are analyzed. It integrates both appearance, spatial and scale co-occurrence information to handle cases with spatial transformation and scale changes. In [115], authors explored both visual and textual modalities using visual vocabulary and semantic context, respectively, for NDK retrieval purpose. They investigated each modality individually and jointly.

In [125], a stochastic attributed relational graph (ARG) matching with part-based representation is proposed for NDK identification. ARG is a fully connected graph with detected Smallest Unvalued Segment Assimilating Nucleus (SUSAN) corners as vertices, and the matching of ARGs is constrained by the spatial relation imposed by corner points. To reduce computational load, a distribution-based similarity model is learned locally and globally in the vertex and graph levels, respectively. The learning is feasible since the matching (or transformation) of NDK pairs often follow certain geometric arrangement; otherwise they should not be categorized as near-duplicate. Although interesting, this method suffers from the limitations of slow matching speed and the requirement of heuristic parameters for learning.

In contrast to [125], the approach in [63] utilizes the PCA-SIFT descriptors of keypoints for direct point set matching. To accelerate matching speed, an efficient index structure based on locality sensitive hashing (LSH) is proposed to facilitate fast search. LSH, nevertheless, requires several user-defined parameters which affect the distortion and granularity of search. In addition, authors apply their approach to high-resolution art images, and its robustness remains unclear when the target database consists of keyframes with low-resolution, motion-blur and com-

pression artifacts. Point set matching is usually followed by post-processing such as RANdom Sample Consensus (RANSAC) [78], Hough transform, or homography constraint checking to overtly remove false matches. However, RANSAC works well when correct matches dominate false matches. Hough transform, on the other hand, is only applicable on keyframes with certain regular shapes like lines and circles. In news videos, applying the aforementioned post-processing techniques has certain limitations where objects can be embedded in highly cluttered background. In [83], similar to [63], authors adopt point set matching, and propose a new index structure called keypoint-IS to support fast filtering. Keypoint-IS is theoretically and practically more desirable than LSH for keypoint-based NDK identification. It is more capable of locating nearest neighbors in approximate search and requires less parameters than LSH. The proposed NDK detector is based on a supervised learning like in [125], but it learns the patterns of matching, rather than the statistical parameters of pre-defined distributions. Since the matching of keypoints are supposed to be spatially coherent for NDK pairs, it may not be necessary to explicitly encode the spatial relation with ARG to constrain matching. As an alternative, this approach performs point set matching with one-to-one symmetric constraint. The outcomes of matching form either spatially regular or random patterns ready for NDK learning and detection but it suffers from the requirement of a large number of appropriate training samples for learning.

Generally speaking, most of the local feature based methods use the number of matching keypoints as the dominant feature. However, there is a group of relevant keyframes with low number of keypoints, which cannot be detected as NDKs through local feature based approaches. These keyframe pairs are smooth

(e.g. sky or sea scenes) according to which low number of keypoints are detected through the feature extraction step. Consequently, low number of matching keypoints can be detected across the keyframe pair which misclassify them as non-NDK pairs. Therefore using global features, such as color-based features, in addition to the local features can boost the performance as explained in Chapter 3.

2.1.2 NDK Clustering

In the literature, various supervised and unsupervised shot-based clustering methods have been reported. A comprehensive overview of video shot clustering methods is presented in [9]. In unsupervised clustering category, various approaches used different clustering frameworks like *k-means*[46], *Ant-Tree* [34], and lately spectral clustering [34, 84, 126]. Due to its effectiveness to reflect the perceptual organization in videos, spectral clustering method performs well on unconstrained videos [9]. The main challenges in unsupervised spectral clustering approaches are determination of a good similarity measure and the number of clusters. For instance, Odobez et al. [84] form a similarity matrix across keyframes according to some global features and temporal information of keyframes. Spectral clustering is adapted to group the keyframes based on the feature distance. Using low-level visual features, this approach has low computational cost and performs well on videos which have long shots in which the similarity (based on the used features) of visual content is high, e.g., videos like documentaries and home videos. Since this study is done on news videos, which generally have a short shot length, and also possess large variations in layouts, context (e.g., object displacement and camera angle variations) together with editing effects (e.g. zooming and occlusion), we

argue that instead of global fingerprints, visual similarity based on local features can be more helpful to handle these variations more effectively. In addition, the temporal order of keyframes is not meaningful in news videos since there might be anchor or reporter shots interspersed in the story stream or there might be repetition of some shots due to lack of raw material. Another spectral approach by Zhang et al. [126] utilizes temporal information along with color-based information through a frame-based framework to model each shot as a GMM based on which cross-shot similarity is determined and corresponding affinity matrix is computed. Using all frames in a video shot, they could not achieve significant improvement compared to the conventional method proposed by Odobez et al. [84] (less than 2% improvement in both precision and recall scores). In [88], the similarity across keyframes is determined based on facial features like number, location and size of existing faces in the keyframes. The number of clusters is adaptively determined based on the cluster validity score. This approach cannot perform well in the news video domain where a large portion of keyframes might include no face. Furthermore, the used face detection algorithm is only capable of detecting frontal faces.

2.1.3 Video Retrieval

The literature on video retrieval consists of approaches addressing duplicate, near-duplicate and partially near-duplicate video detection tasks with the main focus on the accuracy and/or the efficiency of the proposed approach. These methods can be grouped into four broad approaches — signature matching, sequence matching, distribution matching, and context-based approaches. We review some of the important video retrieval techniques below.

2.1.3.1 Signature Matching

Signature matching based approaches can be divided into two groups of global and local signatures. Examples of global signature matching include the use of global color histogram [29] or using color information to average frames in a video as a tiny fingerprint [110]. In [39], authors propose a random histogram to project low-level features and embed them into a high dimensional space using locality sensitive hashing. In [30], a randomized algorithm is proposed to select a number of seed frames and a small collection of closest frames, called Video Signatures (ViSig), is assigned to this set of seed frames. However, depending on the relative positions of the seed frames and ViSigs, this randomized algorithm may sample non-similar frames from almost-identical videos. Although extracting global features can be considered straightforward and they perform well for videos that are exact copies and copies with little variation in the global features, they will fail under intense photometric variations such as the brightness and contrast changes. Moreover, since they take the whole frame as input, it may lead to poor distinguishing power when we deal with keyframes with fixed logo, banner or closed captions.

In local signature matching methods [110, 113, 114, 127, 132], a bag of keyframe local features is determined to represent the whole video. Using a keyframe to represent a shot is a well-known method in the area of video retrieval [21]. This basis is more desirable than per frame basis since a shot is a natural way to segment a video and each of these segments may represent a higher-level concept. Moreover, since video similarity measure involves quadratic computational complexity, the summarization of frames to keyframes can improve the efficiency of

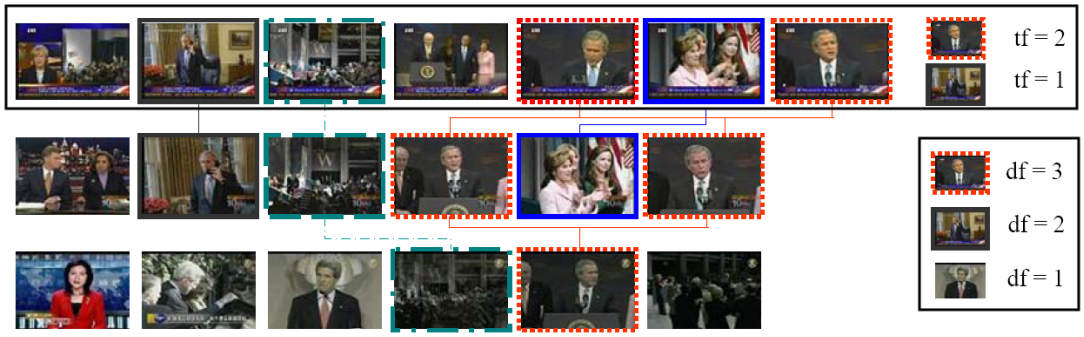

Figure 2.1: Example of stories with common NDKs — Near-duplicate keyframes appeared in three news stories of different channels (tf: term frequency of keyframe, df: document frequency of keyframe) [111].

retrieval. Generally, a video shot is represented by one or a set of keyframes, each of which is described by a set of global descriptors based on color, edge or texture, and/or local descriptors like SIFT [75] or SURF [15]. In [112], authors introduce *must-link* approach based on which they link news stories together if one NDK pair could be detected across them. In some other cases, one might prefer to consider NDK as a *soft-link* constraint and take into account other factors such as similarity level of keyframes and number of NDK links across two stories [56]. In [111], authors explored novelty and redundancy detection with visual duplicates and speech transcripts for cross-lingual news stories. Similar to documents that are treated as a bag of words, they treat a news story as composed of a bag of keyframes in the visual track. The keyframes are further classified as NDK when they appear multiple times in the corpus and as non-near duplicate keyframes (non-NDK) when they appear only once. Figure 2.1 shows an example of stories with common NDKs where NDKs from different stories are connected via colorful lines and their corresponding tf and df are calculated based on their term frequency and document frequency, respectively.

The bag of keyframe approach suffers from some limitations. Firstly, we need to conduct $(m + n)^2$ NDK comparisons, including both within- and between-story pairwise keyframe comparison to form bag of keywords representation for two stories having m and n keyframes. Secondly, they simply identified the connections between NDKs and they did not emphasize which NDK is more representative, e.g., in Figure 2.1 which Bush keyframes are more representative of the story. Thirdly, in cases that have transitivity characteristic between NDKs, the bag of words approach may fail to capture the semantics well. For instance, when keyframes A and B are NDK and keyframes B and C are NDK, bag of words approach does not necessarily consider A and C as NDK pairs.

In general, the above-mentioned methods are robust against a wide range of transformations like lighting changes, object occlusion, cropping and view point changes. However, as mentioned in [132] local feature based approaches have the following drawbacks: first, the stability of local interest points may be unsatisfactory when working with unconstrained video content since the detection rate of near-duplicates in keyframe-based approaches is dependent on the result of keyframe selection to certain extent. Second, the number of extracted local interest points between keyframes may be very different which can lead to poor quality of video matching. Third, the number of local interest points that needs to be determined for a video frame/keyframe is typically high, which results in an expensive computational cost when a large video database is in use. To cope with these challenges, authors in [132] propose shot-based interest points for effective and efficient near-duplicate video retrieval. They assume that the variation of the keypoints in a shot frame is diverse. They choose the shot-based approach instead of keyframe-based since the successive frames in a single shot cannot perfectly

match with each other by their local descriptors due to the possible viewpoint and illumination change, and also the partial object inclusions during the object motions which can negatively affect the similarity matching between keyframes.

Trajectory-based approaches track keypoints along the video sequence to enrich keypoint features with spatio-temporal information. In [114], the whole shot is represented using a bag of trajectories where each trajectory in turn is described as temporal patterns of discontinuities. In general, the extraction of trajectories is an extremely costly operation. Moreover, trajectory features are sensitive to camera motion and therefore their robustness is restricted to copy detection, but not necessarily robust for general near-duplicate detection especially those involving viewpoint changes. The method presented in [66] introduces an approach to consider the temporal behaviors of local features. It tracks the trajectories of keypoints and assigns high level descriptions to each interest point according to trajectory parameters.

2.1.3.2 Sequence Matching

In the sequence matching approaches, the temporal structure of videos plays an important role in near-duplicate detection. These methods facilitate fast retrieval of duplicate videos through the use of global features. However, localization of duplicate video segments is often carried out in a heuristic manner. Typical examples include voting scheme [32] which counts the number of duplicates within different time stamps of a video to locate duplicate segments. Such heuristics do not perform well when we deal with partial near-duplicate videos where interesting parts of a video are copied, edited, and then pasted in random positions on another video. In [108], authors address the issue of partial near-duplicate detec-

tion and localization where the connections across videos are determined using partial alignment of video content. They treat partial alignment problem as a network flow problem. In [57], authors introduce a compact yet effective video signature called video distance trajectory (VDT). This method suffers from severe information loss since it reduces the n-dimensional feature vector, which is basically a color histogram, into a one-dimensional vector.

In [81], authors index every keyframe based on whether they include specific predefined concepts. They treat video copy detection as a semantic concept sequence matching problem. Although the presence of the semantic concepts is robust to spatial and temporal video transformation, many video sequences have similar semantic concepts. Moreover the number of semantic concepts is limited. In [131], authors propose a novel representation, a variable-size video cuboid signature, to describe a video segment. To capture the local spatio-temporal information of video subclips, they utilize the Earth Movers Distance (EMD) to measure the similarity between two signatures. They propose Locality Sensitive Multi-leveled Approximation (LSMA) method to optimize the near-duplicate video similarity matching over streams based on locality sensitive hashing under EMD metric. In [71], authors introduce a compact spatio-temporal feature to represent videos and construct an efficient data structure to index the feature to achieve real-time retrieving performance. Since similar news stories have generally short shots and different lengths and even different temporal orders, sequence matching based approaches do not work well in this domain.

2.1.3.3 Distribution Matching

Distribution matching based approaches take the global data distribution of video clips into account instead of the visual features of each single frame. Shen et al. [101] project each video as a bounded coordinate system called BCS using principal component analysis. In [72] authors present a SIFT-Bag based generative-to-discriminative framework to build Gaussian Mixture Models (GMM) for each event in unconstrained news videos. In [100], each video is summarized into a set of clusters, each of which is modeled as a hypersphere called Video Triplet described by its position, radius, and density. Each video is represented by a much smaller number of hyperspheres. Video similarity is then approximated by the total volume of intersections between two hyperspheres. Although these approaches facilitate a highly efficient computing scheme by constructing a single representation for a video clip, it may lose a large amount of discriminative information. Since news stories generally consist of diverse and heterogeneous visual content with significant amount of irrelevant information, using a single distribution-based signature for the entire story does not provide sufficient discrimination.

2.1.3.4 Context-based Approaches

In context-based approaches, signatures are derived from the context related to the video. For instance, in [113] authors utilize web context such as thumbnail, view count and time duration of a video for real-time web video re-ranking. In [135], the compact signatures are obtained by capturing the color shift and centroid shift of neighboring frames, and detecting the length of each shot in videos. While these methods perform efficiently for the large-scale duplicate

videos, they suffer from lack of discrimination for partially near-duplicate videos.

2.2 Textual Modality

In this section, we briefly describe the related work in refining OCR output and in designing a semantic similarity measure in the textual domain.

2.2.1 OCR Output Refinement Methods

OCR output errors are corrected either by pre-processing or post-processing [62]. In the video domain, text-based video retrieval has been studied by many researchers. In [20], authors explored closed captions, extracted from DVD, and discrete cosine transform of the video frames as two features for classifying movies by genre and learning user preference. In [90], a news story browsing and searching system has been developed using ASR closed captions of stories as well as video OCR. They use a simple pre-processing method to identify candidate text regions and use conventional OCR engine as the text recognizer. Using late fusion framework in the textual domain, authors in [35] use an advanced pre-processing approach for each individual frame to provide more qualified inputs to the OCR engine. For a given frame, a set of various filters is used to detect a candidate text region. A text localization step is adopted to extract potential blocks of text on an image with reasonable precision and recall. The output of OCR is coupled with ASR transcript for the video search task. This approach is computationally expensive due to its inefficient text box verification step. In [54], authors compare *n-gram* analysis [18] and dictionary look up techniques to correct OCR error for the text-based video retrieval. The former generates a new set of n-gram strings, which are contiguous sequences of n letters from given sequences of text, to match

the unedited OCR outputs. Note these n-gram strings include strings with an edit distance of one character and all substrings with at least three characters. The second method looks up the global dictionary, which is a collection of widely used words in English, to find the most similar words to the OCR outputs, to correct spelling errors.

In [129], authors propose the keywords expansion method, which expands the noisy ASR and OCR transcripts to include as many relevant variations as possible, to address the TV commercial classification task. They also use the encyclopedia and English dictionary to correct misspelled terms and obtain keywords. Note that in contrast to TV commercial where background music degrades the ASR transcript quality, in the news domain, ASR transcripts usually possess a higher accuracy. However, in the news domain the quality of OCR transcript varies depending on the video resolution, font size and the complexity of the background of text region.

2.2.2 Textual Semantic Similarity Methods

Different news broadcasters could consistently introduce bias in reporting political and social issues because producers, editors, and reporters collectively make similar decisions based on shared values and beliefs. This bias can highly affect many decisions, e.g., what to cover, what to show on a screen, and what and how to say. Regarding what to show on a screen, authors in [73] proposed a method to automatically detect highly biased news stories which enables audiences to follow news stories from contrasting perspectives. To do so, a statistical model for empathic pattern of visual concepts is introduced. The main idea behind the empathic pattern of visual concepts is that news broadcasters holding

contrasting ideological beliefs appear to highlight different subsets of visual concepts. Regarding what to say, different news broadcaster do not necessarily use the same words to report an identical event. Therefore, their original textual representation can be significantly different, particularly in non-English news where machine translation is used to generate the English version of the extracted ASR transcript. Considering different viewpoints and cultures, authors in [85] proposed a cross-lingual retrieval method for detecting identical news events that exploits text information together with image information.

In such cases, one can also employ a textual semantic similarity metric to measure the textual similarity between stories more effectively. In the literature, there are two main groups of methods addressing semantic similarity between words. The first group is the lexical-based approach [68, 74, 93, 99] in which similarity between words is calculated based on their semantic relations in a predetermined word/concept hierarchy such as WordNet [80] or ConceptNet [74]. There are three main categories within WordNet-based methods: (1) edge counting measured based on the path length between terms [68]. (2) information content measured as a function of the probability of occurrence of the terms in the corpus [93] and (3) combination of the aboves [99]. The major problem with the WordNet-based similarity is that many words (e.g. 'Obama', 'IBM' etc.) are missing from the hierarchy. Moreover, it relies mainly on hyponym information which is good for nouns but not so for adjectives and even verbs.

The second group is the distributional approach [37, 65, 69] in which the semantic similarity/relatedness of two words is calculated based on their co-occurrence in a large-scale collection like Wikipedia [4]. The main idea behind distributional similarity is that similar terms appear in similar contexts. The

first step is to represent the target words as $w = (f_1, f_2, .., f_N)$ where f_i denotes if word v_i occurs in the neighborhood of w and N is the number of words in the lexicon. For instance, if w='tesguino', v_1='bottle', v_2='drunk', v_3='matrix' then $w = (1, 1, 0)$. We can calculate semantic similarity between two words simply as the cosine similarity between their representations.

2.3 Multi-modal Fusion

Multi-modal fusion research has attracted much attention due to the benefit it provides for various multimedia analysis tasks. It is not the intention here to provide an exhaustive survey of the vast literature in this topic. A comprehensive survey article on multi-modal fusion is available in [13]. Our purpose is to focus on multi-modal strategies that we employ in this research. We can categorize them into early and late fusion methods. The integration of multiple media features is referred to as early fusion while the integration of the intermediate decisions is referred to as late fusion [13]. Neither of these fusion methods is perfect [107]. We briefly review related works in the followings.

2.3.1 Early Fusion Approach

Early fusion combines different features into a long feature through which it can implicitly model the correlations between them. For instance, in a multi-modal early fusion approach, authors in [96] study audio-visual fusion using Canonical Correlation Analysis (CCA). They use the co-occurrence of auditory and visual features in the training data to determine the projection functions through which the visual and auditory features can be mapped to another feature space wherein they are comparable and the corresponding features are close to each other. As

another application of early fusion, authors in [14] exploit visual concepts using text query for image retrieval task. They optionally use WordNet [80] to match detected visual concept names and the given text query. By mapping the text query of interest onto the visual concept list, they could retrieve images with the related visual concepts.

2.3.2 Late Fusion Approach

Early fusion does not perform well, if the feature of different modalities is too heterogeneous with skewed length distribution and significantly different scales. In contrast, in late fusion, this is not a concern since the features from each modality will not compare with each other before the final fusion step. In addition, we can employ various detection techniques and classifiers according to specific feature types in late fusion framework. Moreover, the late fusion methods are usually less computationally expensive compared with the early fusion approaches. Therefore, late fusion techniques have become more popular and more extensively studied than early fusion techniques in the literature [107]. Here we mainly focus on the late fusion strategies where the textual and visual similarity modules provide the uni-modal decision scores which later are combined through a decision fusion module to obtain the final decision score.

The late fusion strategies can be categorized into two major groups of (i) Rule-based (e.g MIN, MAX, Ranked List, query (in)dependent weighting fusion etc.) and (ii) Classification-based fusion (e.g. SVM, Bayesian inference etc.) as comprehensively discussed in [13]. We briefly review some related late fusion approaches below.

2.3.2.1 Rule-based Late Fusion Approach

As an exemplary of the rule-based approach in the context of video retrieval, authors in [38] adopt a linear weighted late fusion technique to integrate the normalized scores and ranks of the retrieval results. The normalization was conducted using max-min method. The video shots were retrieved using different modalities such as text and different visual features (color, edge and texture). Authors obtain the best performance in TRECVID type searches by combining scores determined by textual and visual unites with different weights. Combining scores and ranks with equal weights, they obtain the best performance in a single query image.

Query-dependent or query-class weighting can be considered an evolution of query independent weighting since the former tackles many of the latter failures. The focal point of this approach is that given training references and an appropriate set of training queries, query clusters (i.e. query-classes) can be found such that queries within each cluster share some similar properties which differentiate them from other queries in the collection. The properties may be artifacts such as semantic similarity, performance similarity, distance etc. By partitioning a set of training queries into discrete query classes, it is then possible to optimize for each query-class a different set of weights for local decisions. The query-class concepts was introduced by Yan et al. [120] for content-based video retrieval where four classes of Named person, Named object, General object, and Scene were defined, based on which they assign different weights to different low-level classifiers. Later in [119], they developed probabilistic latent query analysis (pLQA) to retrieve latent query classes automatically. A query is soft-assigned to a mixture

of query classes, and adopt the number of query classes under a model selection principle. Xie et al. [116] propose a query-class fusion method that dynamically builds a query class using training queries that are the most similar to the testing query.

2.3.2.2 Classification-based Late Fusion Approach

Among various classification-based late fusion approaches, we focus on the SVM-based late fusion approach since it has been extensively used in many existing multimedia analysis and retrieval methods [13]. In the context of multimodal fusion, SVM [22] is employed to build a separating hyperplane by using scores given by the individual classifiers. Using the kernel concept, the basic SVM method can be extended to create a non-linear classifier, where every dot product in the basic SVM formulation is replaced with a non-linear kernel function.

In the area of image classification, authors in [134] propose a multimodal fusion framework to classify the images, which have embedded text within their spatial coordinates, through a two-step fusion process. First, a bag-of-words model [70] is adopted to classify the given image using the low-level visual features. In the textual domain, the text detector localizes the textual region in an image using text color, size, location, edge density etc. Second, a pair-wise SVM classifier is trained for fusing the visual and textual features together. Authors in [10] employ a late fusion approach for semantic visual concept detections (e.g. sky, fire-smoke) in videos using visual, audio and textual modalities. They develop a discriminate learning approach while fusing different modalities at the semantic level. Particularly, the scores of all intermediate concept classifiers are integrated to construct a vector that is passed as the semantic feature to SVM. In

[130], authors introduce a fusion scheme, called double fusion, which integrates early fusion and late fusion together, to address the Multimedia Event Detection (MED) task. They use multiple visual and textual representations and treat the early fusion result as an individual classifier. Then an SVM-based fusion scheme is adopted using all developed uni-modal classifiers together with the early fusion classifier. Promising results are reported on TRECVID MED 2010 and 2011 datasets which shows significant improvement compared to the sate-of-the-art performance.

Chapter 3

Near-Duplicate Keyframe Identification and Clustering

Near-Duplicate Keyframes (NDK) are keyframes that are very similar to each other despite the slight to moderate degree of variations caused by lighting, camera viewpoint, acquisition time, motion, and editing effects. NDKs are commonly found in broadcast video streams. It is well known that the real challenge in identifying NDKs is due to the variation in motion, illumination, view point and editing effects between potential near duplicate keyframes [125, 128, 133]. Figure 3.1 shows examples of challenging NDKs due to zooming in (a) and (c), object/camera motion in (b) and (d) and low number of keypoints in (e).

In the NDK analysis, global features are making way for local features invariant to the kind of transformations mentioned above. Local features include keypoints and their associated descriptors extracted from local patches in an image. The most popular among them is the space invariant feature transform (SIFT) descriptor [75] and their variants such as PCA-SIFT. The former is shown to be scale invariant as well as tolerant to a certain degree of affine transformation. Similarly, the latter is known to be robust to color and photometric changes

Figure 3.1: Challenging NDK examples — Examples of NDKs with zooming in (a) and (c), object/camera motion in (b) and (d), low number of keypoints in (e).

[63].

In this chapter, first we develop an algorithm for NDK identification which includes retrieval and detection. The task of NDK retrieval involves ranking all near duplicates to an input query image while that of NDK detection involves identifying NDK pairs. The former is useful for copyright infringement detection [63] and query by example applications, while the latter finds applications in linking news stories and grouping them into threads [125] and in multimedia search [26]. Next, we investigate the application of the proposed keyframe-level similarity measure to tackle keyframe clustering problem within a news story.

The rest of this chapter is organized as follows. In Section 3.1, we explain our proposed NDK retrieval algorithm based on keypoints matched between two keyframes. In Section 3.2, we propose a novel NDK detection method using different global and local features. Section 3.3 describes our proposed content-based keyframe clustering method. In Section 3.4, we assess the proposed NDK

retrieval, detection, and clustering algorithms using standard datasets.

3.1 NDK Retrieval

Our work is motivated by [128] in which one of the main issues addressed is that of reducing the large number of keypoints generated so that matching between two images can be done, efficiently. The authors propose a one-to-one symmetric (OOS) matching scheme between keypoints in two images. The speed and search effectiveness of the matching scheme is enhanced through an indexing structure and a decision rule. Finally, a support vector machine (SVM) is trained to learn a histogram of matching orientations of the lines that join the matched keypoints. In [83], they used entropy of matching patterns in vertical and horizontal directions as a measure for NDK detection instead of the SVM-based learning algorithm. For that purpose, they capture the matching patterns of keypoints with two histograms of matching orientations. Figure 3.2 shows vertical and horizontal alignments of two keypoints in a pair of keyframe. Keypoint A and B from keyframe 1 are matched to keypoint A' and B' in keyframe 2, respectively.

Considering the horizontal and the vertical alignments of keypoints between two keyframes, we construct histograms G_h and G_v, respectively. A histogram is composed of the quantized angles, θ_h or θ_v, formed by the matching lines and horizontal or vertical axis. Let h be the height of the upper keyframe (Keyframe 1 in Figure 3.2). The coordinates of keypoint A in Keyframe 1 and keypoint A' in Keyframe 2 are (x_0, y_0) and (x_1, y_1), respectively. The angle θ_v of the line joining

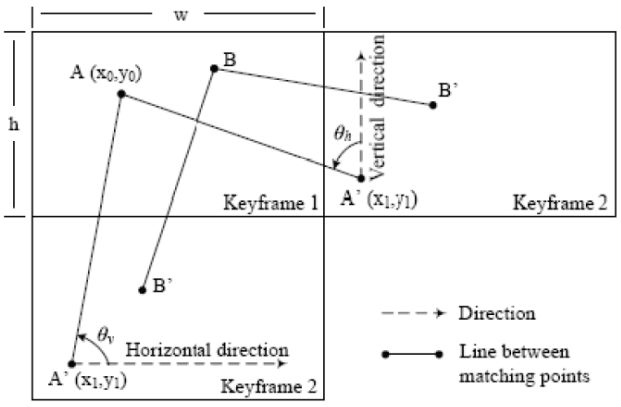

Figure 3.2: Keypoint alignment of a pair of keyframe — Keypoint A and B from keyframe 1 are matched to keypoint A' and B' in keyframe 2 [83].

A and A' is computed as

$$\theta_v = \cos^{-1}\left(\frac{x_1 - x_0}{\sqrt{(x_1 - x_0)^2 + (y_1 + h - y_0)^2}}\right). \tag{3.1}$$

The histogram G_v is formed by computing θ_v of all matching lines and then accumulating the count to the corresponding bins. The histogram G_h is computed in a similar manner by the angle θ_h. Let w be the width of the left keyframe (Keyframe 1 in Figure 3.2), the angle θ_h is computed as

$$\theta_h = \cos^{-1}\left(\frac{x_1 - x_0}{\sqrt{(x_1 + w - x_0)^2 + (y_1 - y_0)^2}}\right). \tag{3.2}$$

Histograms are quantized into 36 bins ranging from 0° to 180°. Ideally, the parallel or parallel-like lines should fall in the same bin of histograms. Histograms G_h and G_v intuitively suggest the spatial coherency of matching lines in the horizontal and vertical directions and depict the different partitions of orientation for the same set of matched keypoints. For an NDK pair, both histograms should be correlated. Specifically, whenever a peak in G_h is found, there should exist a corresponding

peak of the same keypoints in G_v. To reveal the mutual information between G_h and G_v, the authors in [83] use entropy to measure the homogeneity of histogram patterns. They call their measure Pattern Entropy (PE). Denote N as the number of bins in a histogram, and $P = [p_1, p_2, ..., p_m]$ and $Q = [q_1, q_2, ..., q_n]$, where $m \leq n \leq N$. The notation p_i (similarly for q_i) is a non-empty set of keypoint pairs that fall in an identical bin of the histogram. Physically, P corresponds to one of the histograms (G_h or G_v) with less non-empty bins, while Q corresponds to the other histogram. P is more compact than Q since fewer bins are used to accommodate the matched keypoints. In pattern entropy, the degree of points in p_i being distributed in Q is defined as [83]:

$$\mathrm{PE}(Q, P) = \frac{1}{S} \sum_{q_i \in Q} \mathrm{Entropy}(q_i, P), \tag{3.3}$$

where

$$\mathrm{Entropy}(q_i, P) = -\frac{1}{\log m} \sum_{p_j \in P} \frac{|q_i \bigcap p_j|}{|q_i|} . \log \frac{|q_i \bigcap p_j|}{|q_i|}, \tag{3.4}$$

$$S = \sum_{q_i \in Q} |q_i| = \sum_{p_i \in P} |p_i|, \tag{3.5}$$

and $|q_i \bigcap p_j|$ is the cardinality of intersection between two sets p_i and q_j. Basically, Entropy $\in [0, 1]$ measures the extent of dispersing a set p_i across the bins of another histogram in the orthogonal direction. An Entropy value of 0 indicates the keypoints in p_i are found exactly in another set q_i of Q. A value of 1 indicates that the keypoints in p_j are evenly distributed in some sets of Q. PE always lies in [0, 1]. The extreme value of PE=0 indicates a perfect coherent match in both horizontal and vertical directions. Conversely, the value of PE equals to 1 basically shows a random match across space. In the evaluation, keyframes with

(a) 0.73 (b) 0.76 (c) 0.72 (d) 0.68

Figure 3.3: Example NDKs where Pattern Entropy fails — The number below each sub-figure shows the corresponding PE score [133].

PE ≤ 0.5 are declared as NDK pairs. But this approach does not perform well in some cases which can be categorized into three major groups of illumination variation, occlusion, and zooming. Figure 3.3 shows 4 NDK pairs and matching lines between them calculated using One-to-One Symmetric (OOS) matching method [83]. PE scores, shown below sub-figures, are more than half which leads to the failure of PE method to detect these pairs of keyframes as NDK.

Figure 3.4 shows the framework of the proposed NDK retrieval algorithm. We propose a new measure for keypoint matching that considers the ratio of the distance of the nearest neighbor and the second nearest neighbor to a keypoint in the query image. Next, we propose a keypoint clustering scheme to assign a score (PC score) to each potential near duplicate keyframe based on the geometric consistency of matching keypoints. Finally, the above two measures are combined through linear discriminant analysis (LDA) and the decision boundary is trained by an SVM classifier.

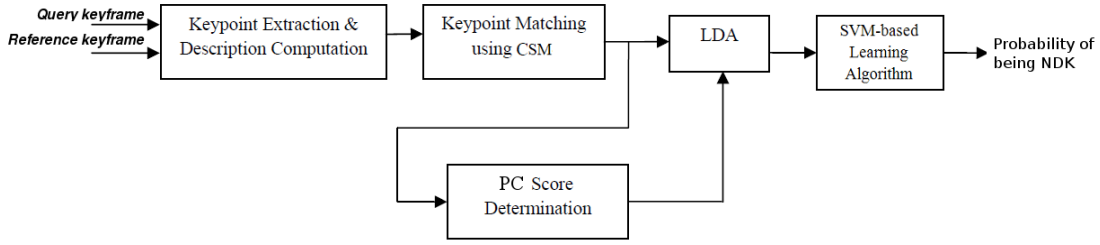

Figure 3.4: NDK retrieval algorithm overview — Input: query and reference keyframes. Output: Probability of being NDK.

3.1.1 Keypoint Extraction

Since we develop our algorithm based on local features, first of all we need to extract keypoints. The keypoint detectors basically locate stable keypoints (and their support regions) which are invariant to kinds of transformations introduced by geometric and photometric changes. Popular detectors include Harris-Affine [79], Hessian-Affine [79], and Difference of Gaussian (DoG) [75]. The keyframes are converted from RGB to gray-scale before applying detectors. We choose the Hessian detector because of its better performance [75].

The descriptors of keypoints are invariant to certain transformations that exist in different images. SIFT (Scale-Invariant Feature Transform) is shown to be one of the best descriptors for keypoints [75]. It is a 128-dimensional feature vector that captures the spatial structure and the local orientation distribution of a patch surrounding a keypoint. PCA-SIFT (36-dimensional vector) is a compact version of SIFT with principal component analysis on gradient field of local patch. However, the PCA-SIFT has been empirically shown to be less distinctive than original SIFT [79] and is also slower than the original SIFT in feature computation. Hence, we use the original SIFT descriptors.

Figure 3.5: Logo and subtitle matching keypoints in a non-NDK pair — This example shows the importance of removing keypoints located in the textual region or fixed logos/banners.

3.1.2 Logo and Subtitle Keypoint Removal

Before finding matching patterns between a keyframe pair, we need to remove keypoints located in a logo or subtitle area, since this group of keypoints may incorporate in meaningless matchings. Figure 3.5 shows a pair of keyframe which are not NDK, but there are few matching lines between their keypoints located at the logo and subtitle areas. We use TRECVID 2006 dataset which contains news videos from different channels. All keyframes from the same channel may have the same logo which can be matched. Similarly, matching lines between the letters in subtitles should be removed since they are not meaningful and reliable in our task.

Since the location of the logo and the subtitle are fixed for a particular channel, we use this information to spatially filter out the keypoints located in these areas. An example of the location of logo and subtitle for NBC channel is shown in Figure 3.5 by yellow rectangles.

3.1.3 Constrained Symmetric Matching (CSM)

Given keypoints extracted separately from two keyframes, we need to align them to facilitate the similarity measurement. Keypoint matching is considered as a bipartite graph matching problem. There are numerous algorithms for point set matching. Depending on the mapping constraint being imposed, we can categorize them as many-to-many (M2M), many-to-one (M2O), one-to-many (O2M), and one-to-one (O2O) matching [83]. The factors that affect the choice of matching strategy include noise tolerance, similarity measure, matching effectiveness and efficiency. In videos, frames suffer from low-resolution, motion-blur, and compression artifacts. Noise becomes a crucial factor in selecting matching algorithm, particularly when the matching decision is made on a small local patch. In NDK identification, keyframe transformation introduces noise, which affects the performance of keypoint detection. Consequently, it becomes very common that a keypoint fails to find its corresponding keypoint in another keyframe, and on the other extreme, a keypoint can simply match to many other keypoints. O2O matching could suppress faulty matches although some correct matches may be missed. Although M2M is applicable for many retrieval problems, there exists no effective mechanism to restrict false matches.

Our proposed matching algorithm which has a point to point basis has two stages. The first stage is to identify the matching set of keypoints for a particular keypoint in one image from the other image. While any distance measure in the feature space can be used, we employ the cosine distance between keypoints for its simplicity, i.e., the cosine of the angle between two SIFT vectors. Although nearest neighbor matching schemes might result in missing out true matches, it

is essential that as many false matches are eliminated. Moreover, the matching keypoints must be meaningful in the sense that they belong to the same object or a similar region. In order to discard keypoints that do not have a good match, Lowe suggests the use of the ratio of the nearest neighbor to that of the second-closest neighbor [75]. Our proposed matching algorithm begins by defining the Ratio of Distances (RoD) between keypoint i in frame A and keypoint j in frame B as:

$$\text{RoD}(k_i^A, k_j^B) = \frac{\text{MAX}_{l=1,..,n_B}(< \mathbf{D_i^A}, \mathbf{D_l^B} >)}{< \mathbf{D_i^A}, \mathbf{D_j^B} >}, l \neq j \tag{3.6}$$

where $\mathbf{D_i^A}$ and $\mathbf{D_j^B}$ are the SIFT vectors of the i-th keypoint of keyframe A and the j-th keypoint of keyframe B, respectively, n_B is the number of keypoints in keyframe B and $<,>$ is the cosine distance between two vectors. Next, we define an association rule between the keypoints in frame A and B as:

$$k_i^A \Rightarrow k_j^B \quad if \quad \text{RoD}(k_i^A, k_j^B) < \tau \tag{3.7}$$

where τ is a threshold ($0 < \tau \leq 1$). As shown in the experimental results, τ is determined from its probability density function for correct and incorrect matches. It gives an idea of how much of the true and false matches are eliminated for each value of τ. As τ is decreased from 1, it imposes more restriction on the association of two keypoints since their similarity must be much farther away from the similarity to other keypoints. Based on Equation (3.7), we define a matrix $\mathbf{R_{A \to B}} = [r_{ij}]_{n_A \times n_B}$ where

$$r_{ij} = \begin{cases} 1 & if \quad k_i^A \Rightarrow k_j^B \\ 0 & \text{otherwise} \end{cases} \tag{3.8}$$

Similarly, we define the matrix $\mathbf{R_{B \to A}}$.

In the second stage, we combine the symmetric property of keypoints in NDKs with the SIFT keypoint matching scheme. We propose a new symmetric pairing mechanism for the keypoints through the matrix $\mathbf{R_S}$ defined as:

$$\mathbf{R_S} = \mathbf{R_{A \to B}} \circ \mathbf{R_{B \to A}}, \tag{3.9}$$

where $\circ$ indicates element-by-element multiplication. The i-th keypoint from frame A and the j-th keypoint from frame B are paired if and only if $\mathbf{R_S}(i, j)$ is 1. We call this keypoint matching scheme as constrained symmetric matching (CSM) and evaluate it for different values of τ in Section 3.4.2. The effectiveness of CSM is also illustrated in Section 3.4.2 by an NDK retrieval task where the numbers of correct retrievals in top-1 are ascertained. If numbers of matching keypoints of keyframe pairs are the same, the keyframes are further ranked according to their color-based similarity. Specifically, the keyframe is divided into blocks of size 5×5 and the mean and variance of each color channel in each block is determined. Note that the color feature is used only when there is a conflict arising due to equal number of matching keypoints in two or more pairs.

3.1.4 Pattern Coherency

The next step in the proposed framework is to assign a score to each pair of keyframe based on the keypoints that are matched. In [83], Ngo et al. recognize that the matching of keypoints across keyframes results in a pattern formed by the lines joining the keypoint from one keyframe to the matching keypoint in the other keyframe. Their pattern entropy is derived from the angles that the matching lines make with the vertical and horizontal direction. Any pair of images with PE ≤ 0.5 is considered near duplicates. In [133], it is shown that the PE

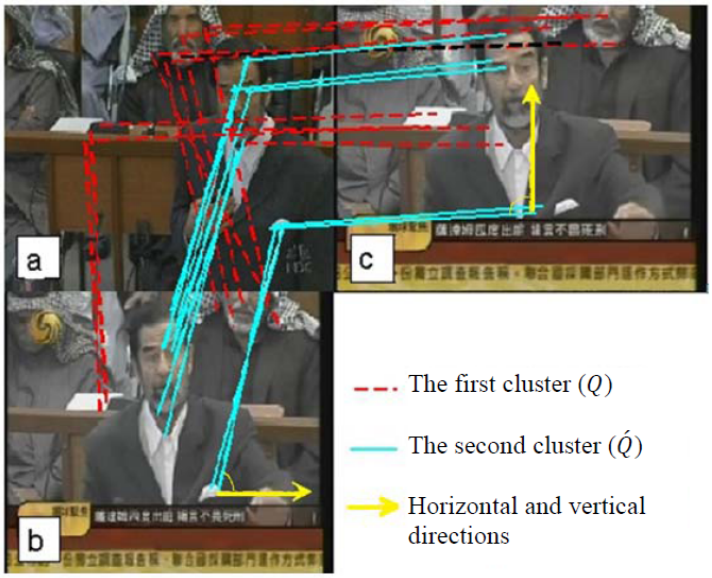

Figure 3.6: Pattern Coherency determination — we use both vertical and horizontal alignments.

fails in cases where there is a change in illumination, object motion or zooming.

In this section, we propose a more effective method to recognize the pattern of matching lines for NDK pairs. The key idea is that in NDK pairs, the matching lines joining the keypoints of identical object in two frames are relatively coherent. We propose a robust score called Pattern Coherency (PC) to measure the total coherency of groups of matching lines that join objects in pairs of keyframes. We explain the procedure with the aid of an example shown in Figure 3.6. First, we split the matching lines between Figure 3.6(a) and Figure 3.6(b) into two clusters (Q and Q') based on the angles that they make with the horizontal directions. Our choice of two as the number of clusters comes from the observation that, generally, objects located in the background or the foreground have roughly the same matching direction in NDK pairs. A simple k-means algorithm is used to

perform the clustering. The mean distance of the angles in the cluster Q to its centroid, Q_c, is obtained as

$$\Omega_Q = \frac{1}{|S_Q|} \sum_{i=1}^{|S_Q|} (\theta_{Qi} - Q_c)^2, \tag{3.10}$$

where $|S_Q|$ is the number of points in cluster Q and θ_{Qi} is the angle of the i-th matching line in Q with the horizontal direction. Similarly, we determine $\Omega_{Q'}$ using Equation (3.10). In Figure 3.6, Q and Q' are shown as blue solid and red dashed lines, respectively, between (a) and (b). Next, we compute PC_v, pattern coherency of vertical alignment, as

$$\mathrm{PC}_v = \Omega_Q + \Omega_{Q'}. \tag{3.11}$$

The advantage of the clustering method lies in the fact that matching lines as a result of significant object displacement and extreme zooming can be divided into two components and analyzed independently (i.e. cluster of lines between Figure 3.6(a) and Figure 3.6(b)) and the PC_v score will be determined by summation of their individual pattern coherency as mentioned in Equation (3.11). Considering them as a single set, as in [83], would result in a combined contribution of horizontal and vertical angles in cases of motion/zooming/occlusion which leads to a high value of PE score. However, in Figure 3.6(a) and Figure 3.6(b), due to within coherency of two clusters computed by Equation (3.10), the determined PC_v based on Equation (3.11) will be low.

We determine PC_h similarly using matching lines in horizontal alignment (i.e. matching lines between (a) and (c) in Figure 3.6). Although the object displacement cannot be observed in this alignment, PC_h computed by Equation (3.10) and (3.11) is still low. We conclude that PC score works well either in the presence or in the absence of object displacement.

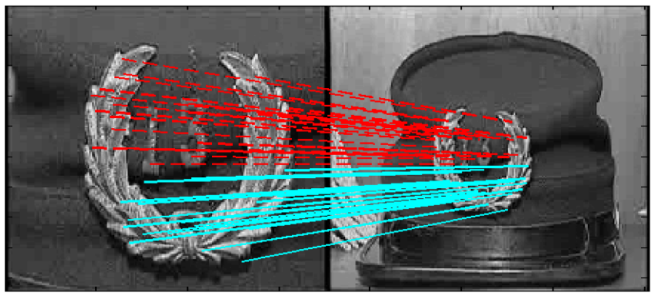

Figure 3.7: An example of extreme zooming and matching keypoints — Blue solid lines and red dash lines form the first (Q) and the second cluster (Q'), respectively.

Since we claim that the low value of either PC_h or PC_v is sufficient for being NDK, we impose a weaker constraint by taking the minimum of PC scores in vertical and horizontal direction as $\mathrm{PC} = \min(\mathrm{PC}_h, \mathrm{PC}_v)$. Later in Section 3.4.3, we will justify the choice of min based on experimental results.

Figure 3.6 is a good example of such cases which were misclassified as non-NDK by [83] due to significant object displacement while our proposed method retrieves it correctly as an NDK pair. Figure 3.7 illustrates an example of extreme zooming whose pattern coherency is determined relatively low by using the proposed clustering technique while its PE score is high which leads to its misclassification based on [83]. This confirms that when compared to PE, PC can handle extreme zooming cases as well. At the same time, other variations due to illumination and small regional changes are also handled well by PC.

3.1.5 Training

In this section, we consider the number of matching keypoints obtained in Section 3.1.3 and the PC score of Section 3.1.4 as features to be trained using SVM. Figure 3.8 shows the scatter diagram of these features for NDK and non-NDK samples. It is

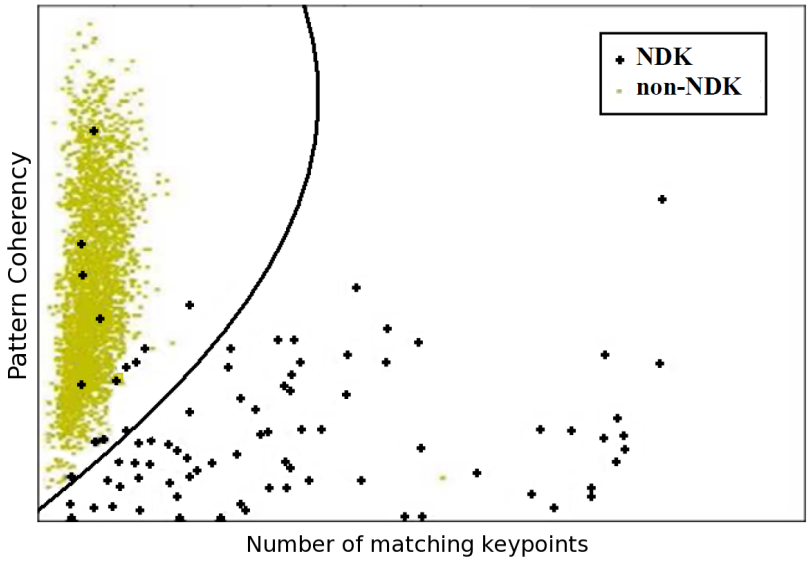

Figure 3.8: Scatter diagram of NDK and non-NDK samples in Columbia dataset [125] — The solid line shows the trained separating boundary.

evident that the non-NDK samples are clustered along the PC axis while the NDK clusters are distributed along the number-of-keypoints axis. The SVM boundary separating the two classes indicates that the features are discriminative enough for training the SVM to detect NDK and non-NDK pairs. We use linear discriminant analysis (LDA) to determine the linear combination of these two features that best separates the two classes of data. In addition, we also treat the two features independently and train the SVM using a 2-D vector of these features. The output of the SVM in either case is the probabilities that the features have been derived from an NDK or a non-NDK pair. An RBF kernel is used for the SVM with the penalty parameter C set to 10.

3.2 NDK Detection

Although NDK retrieval and detection have the same basis, there are two more issues that needs to be addressed in the context of NDK detection. First, as shown in Figure 3.1, there are groups of challenging NDK which can not be detected by using only keypoint matching approaches. In addition to keyframe pairs with significant object displacement (e.g. Figure 3.1 (b)) and/or extreme zooming (e.g. Figure 3.1 (a)), there is also another group of relevant keyframes with low number of keypoints which cannot be detected as NDK by local feature based approaches. These keyframe pairs have mostly smooth regions (e.g. Figure 3.1 (e)) according to which low number of keypoints are detected in the feature extraction step. Consequently, low number of matching keypoints can be detected between the keyframe pair which classifies them as non-NDK pairs.

Second, for NDK detection task we need to specify a threshold for near duplicate score, with respect to which we can discriminate NDKs from non-NDKs. This fact shows the importance of the separating boundary trained by the SVM. Particularly, since in the available datasets there are much more negative than positive samples of NDKs, suppressing the effect of unbalanced data on the trained separating boundary becomes critical.

According to the above explanations, in addition to the number of matching keypoints explained in Section 3.1.3, we determine a color-based similarity score and difference of complexity measure as two more discriminative features to be fed to an SVM classifier (Figure 3.9). We explain the last two features and their usefulness in the following. We also propose a solution to cope with the imbalance of the used datasets by selecting a subset of the dataset for training

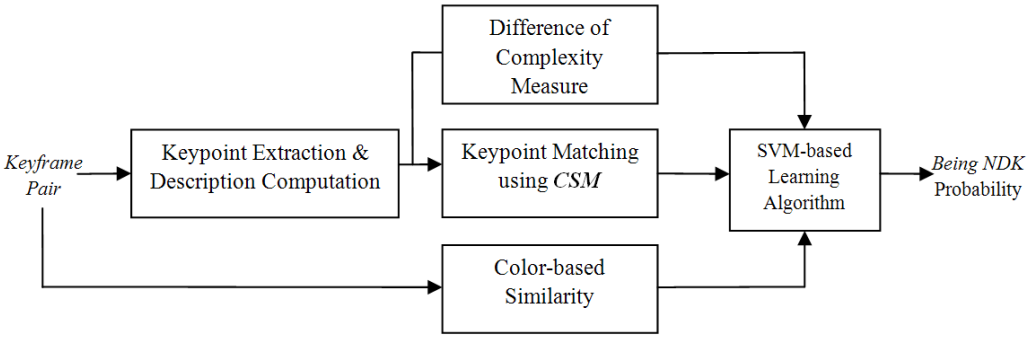

Figure 3.9: Our proposed NDK detection framework — Input: a keyframe pair. Output: Probability of being NDK.

an SVM classifier. The output is an SVM kernel based on which we can detect NDKs within a news story at a proper level of precision and recall. We also evaluate the proposed NDK detection algorithm on Columbia and NTU datasets.

3.2.1 Color-based Similarity

As shown in [117, 124, 128], the number of matching keypoints is the most discriminative feature in a NDK identification task. In Figure 3.10, we show the scatter diagram of NDKs and non-NDKs for two datasets. We observe that NDKs with significant number of matching keypoints are easily separable from non-NDKs. Most of the local feature based approaches use this feature alone for the NDK identification task due to its high distinguishing power. Using this feature alone, we obtain high precision but not a desired recall. This means that there is a group of (less strict) NDKs, particularly duplicate scenes, which can not be identified using only number of matching keypoints. The challenging issue here is how to detect actual NDK pairs between which the number of matched keypoints

is not substantial like in Figure 3.1(e) or when they do have adequate number of keypoints but due to intense variations and drawbacks of the matching scheme, there is a low number of matching keypoints between them like in Figure 3.1(b). This motivates us to determine a color-based feature for the keyframe pair as a complementary feature.

In [133], the global feature of 5-by-5 grid color moment and edge properties have been used as the first layer of a Multi-Level Ranking (MLR) framework to filter out more distant pairs of keyframes, efficiently. The remaining keyframes are passed to the next layer of local-feature-based analysis which is more accurate but computationally expensive. Although the Multi-Level Ranking (MLR) framework improves the efficiency and scalability, some NDKs are missed since the first ranking component in this method is based on global features. Moreover, technically the NDK pair that differs due to extreme zooming or significant object motion, e.g., in Figure 3.1(a), may have very different color moment and edge properties. These observations motivate us to develop a color-based similarity measure that is not block-based and incorporate it as an auxiliary feature along with the number of matching keypoints feature to detect NDKs (not as an individual feature through MLR framework). For this purpose, keyframes are characterized by a color histogram h_i in RGB space (uniformly quantized to $8 \times 8 \times 8$ bins). The color-based similarity is computed by the metric based on Bhattacharyya coefficient, shown in Equation (3.12), which has proven to be robust to compare color distributions [33] and outperforms the Mahalanobis and Euclidean distance measures in general image retrieval system [91]. Thus, the color-based similarity (CS) between two keyframes K_A and K_B is expressed in

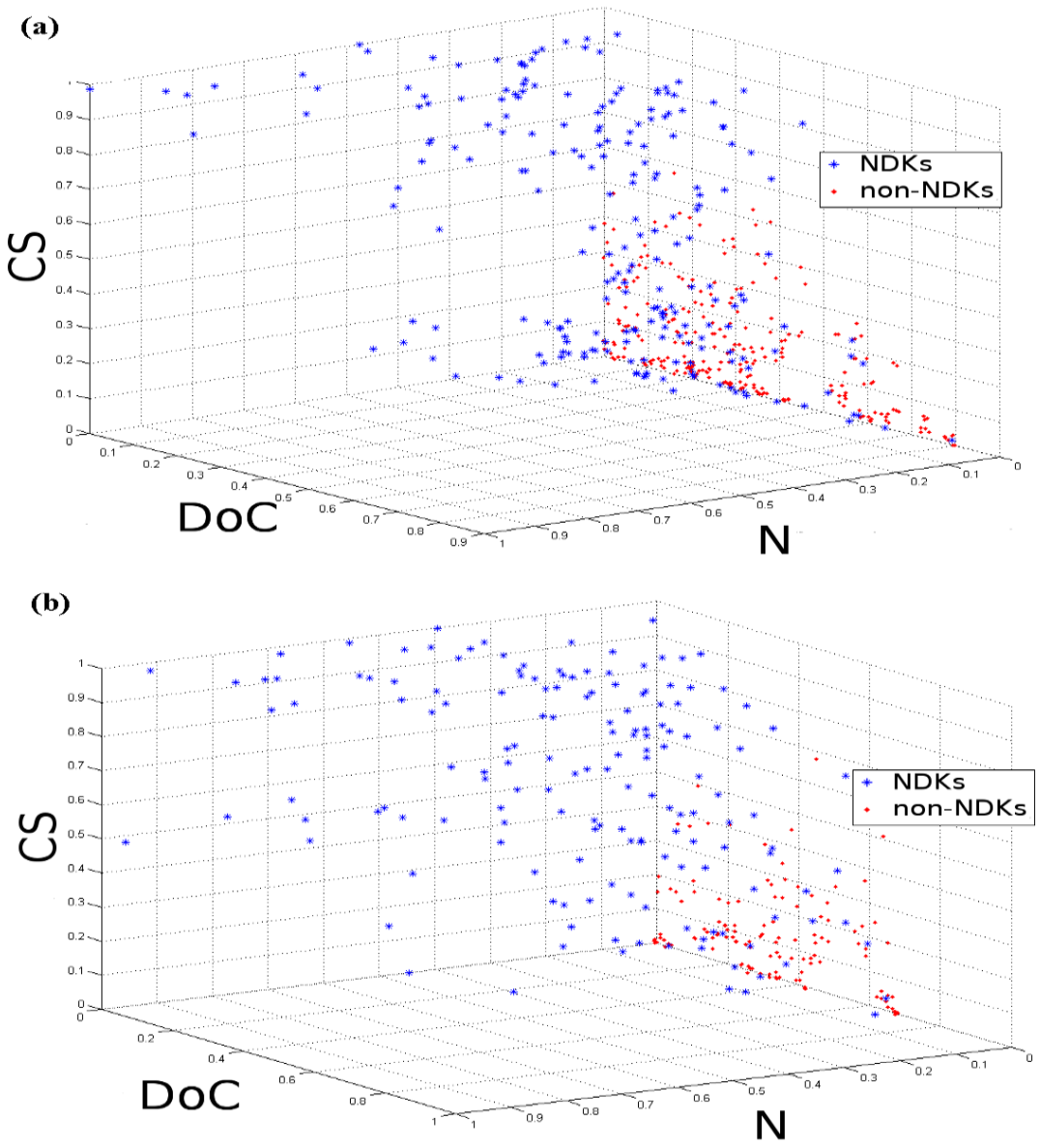

Figure 3.10: Scatter diagram of NDKs and non-NDKs based on three features — (a) on Columbia dataset, (b) on NTU dataset. N, CS and DoC refer to the Number of matching keypoints, Color-based Similarity and Difference of Complexity scores, respectively.

terms of color histograms C_A and C_B as

$$d_c(C_A, C_B) = \exp(-\frac{1 - \rho_{\mathrm{BT}}(h_A, h_B)}{2\sigma^2}) \tag{3.12}$$

where ρ_{BT} denotes the Bhattacharyya coefficient, defined by $\sum_k (h_{A_k}, h_{B_k})^{\frac{1}{2}}$, the sum running over all bins in the histograms. We fix the σ value to 0.25 [84].

3.2.2 Difference of Complexity Measure

In this section, we introduce another feature called Difference of Complexity (DoC) measure, to improve NDK detection performance especially when we deal with NDK pairs without adequate matching keypoints. The basic assumption here is that NDKs are expected to possess similar amount of information. In the literature, this information gain is often determined as the image entropy. But here we aim to study image complexity from a different point of view.

Image complexity in the context of NDK detection is a measure of inherent difficulty of finding a pair of NDKs. The difficulty of NDK detection task depends not only on the input keyframes, but also on the type of information it extracts and the method employed to extract it. To address this difficulty, first we simply determine image complexity by counting the number of keypoints detected by Difference-of-Gaussians (DoG) detector through feature extraction explained in Section 3.1.1. There is a semantic relatedness between our proposed image complexity metric and edge-dependent image complexity (local) metrics presented in [89]. We count number of keypoints (batch) detected via DoG approach which refers to points and/or regions in an image that are either brighter or darker than the surrounding, while in [89] the number of edges per unit in an image is determined as the image complexity measure. The main advantage of using keypoints instead of edges to determine image complexity is that the former is scale-invariant. In addition, it imposes no additional computation cost since keypoints are needed to be detected earlier for the keypoint matching process. We

introduce a symmetric DoC measure for keyframe pair K_A and K_B as

$$\mathrm{DoC}(K_A, K_B) = \frac{|n_A - n_B|}{\max(n_A, n_B)}, \tag{3.13}$$

where n_A and n_B denote numbers of keypoints detected in K_A and K_B, respectively. This pairwise symmetric DoC measure returns zero for exact duplicate keyframes (i.e. the simplest case) and a relatively low value for near duplicate keyframes (i.e. more difficult cases) and can be an arbitrary value between zero and one for non-NDKs, as shown in Figure 3.10.

Note that the proposed DoC measure is not considered a discriminative feature by itself but it can improve the NDK detection result together with two other features (i.e. number of matching keypoints and the color-based feature). For instance, returning to the problem of inadequate number of matching keypoints, keyframes in Figure 3.1(e) have 176 and 217 keypoints, respectively, and we could not find a significant number of matching keypoints, using the proposed CSM method. In such a case, in addition to color-based similarity (i.e. 0.85), a low DoC measure (DoC=0.18) can incorporate and boost the probability of being a NDK. In other words, the DoC score along with the color-based similarity score plays an important role for keyframe pairs having inadequate number of keypoints.

3.2.3 SVM-based Learning Algorithm

After extracting the features, we train the separating boundary using a SVM classifier with RBF kernel for the NDK detection task. We utilize Columbia dataset for this purpose which includes 150 NDK pairs and 300 non-NDK keyframes. Accordingly, there are 150 positive samples versus approximately $600 \times 599 - 300 =$

359,100 negative samples. This means that we deal with extremely unbalanced data points for classification. To reduce the negative data size, we use subsampling and then we apply *KNN*-based filtering. Specifically, first we apply stratified random sampling [94] to reduce the negative sample size down to 1,200 by partitioning the data points into strata along two axes — number of matching keypoints and color-based similarity score — independently and then sub-sampling by a factor of 300. The 1,200 negative samples obtained thus are classified by *KNN* classifier with $K = 5$. Each data point which has at least one K-nearest neighbor that is a positive sample is retained and the rest are removed. Finally, we obtain 150 positive samples versus 255 negative samples which are more likely to be around the desired separating boundary (Figure 3.10). The optimal RBF kernel parameters (i.e. γ and C) are then determined through a coarse-to-fine grid search based on the Equal Error Rate (EER) for near duplicate detection. EER measures the accuracy at which the number of false positives and false negatives are equal. We evaluate the proposed NDK detection algorithm based on pre-specified training and testing partitions in Section 3.4.4.

3.3 Content-based Clustering of Keyframes

Effective clustering of video shots is an important step in applications involving content-based video analysis and retrieval. For instance, Rui et al. [95] investigated how a proper grouping of video shots can be useful for video content browsing and retrieval. Furthermore, a wide range of other video-related applications from content-based annotation of a video to video summarization can benefit from effective clustering of similar shots [45, 92]. Generally speaking, dif-

ferent shot clustering approaches utilize either all the frames in a video [27] or only a frame representing the shot, called the keyframe [84] as the initial unit of a video. In this study, we consider a keyframe as the representation of a video shot and tackle the keyframe clustering problem to group the keyframes within a news story.

In the proposed NDK clustering approach, the SVM probability from the previous section serves as a keyframe similarity metric. For this purpose, a set of keyframes is represented in a graph structure where each keyframe is a vertex and the edge weight between a pair of keyframes is the probability of the pair being an NDK as determined in the previous section. We call this probability as Near Duplicate Score (NDS). Hence, each story is represented as graph $G = \{V, E\}$ with $V = \{v_i\}$ the set of vertices and the edge weight between vertices i and j given by

$$E_{i,j} = \text{NDS}(v_i, v_j),\ i, j = 1, 2, .., n,\ i \neq j, \tag{3.14}$$

where n is the number of keyframes in the story and $\text{NDS}(v_i, v_j)$ refers to the probability that the i-th and the j-th keyframes are NDKs. Hence, the problem boils down to finding the representative subgraphs $G_i, i = 1, 2, .., k$ within G so that the intra-subgraph similarity is maximum and the inter-subgraph similarity is minimum, simultaneously. This problem is also called the *mincut* problem in graph theory. One of the most common objective functions for this problem is the RatioCut introduced in [49]:

$$\text{RatioCut}(G_1, G_2, .., G_k) = \frac{1}{2} \sum_{i=1}^{k} \frac{W(G_i, \bar{G}_i)}{|G_i|} \tag{3.15}$$

where $W(G_i, \bar{G}_i) = \sum_{i \in G_i, j \in \bar{G}_i} E_{i,j}$, $\bar{G}_i$ denotes the complement of G_i and k is the number of clusters. Although this optimization problem is NP-hard, its relaxed

version leads to an unnormalized spectral clustering problem whose steps are outlined in Figure 3.11 [76]. In the spectral clustering scheme, we need to provide a meaningful graph similarity measure and pick a suitable k as the number of clusters as shown as the inputs in Figure 3.11. We utilize E defined in Equation (3.14) to construct the weighted adjacency matrix, $\mathbf{W} = [w_{i,j}]_{n \times n}$ whose elements are:

$$w_{ij} = \begin{cases} E_{i,j} & if \quad E_{i,j} > \tau_S \\ 0 & \text{otherwise} \end{cases} \tag{3.16}$$

where τ_S is a threshold ($0 < \tau_S < 1$). By thresholding NDS, we prune outlier keyframes whose visual information is rare in the entire news story. Since NDS is determined as a semantically enriched similarity measure, we demonstrate its effectiveness in removing the outlier keyframes in the experiment section.

It should be mentioned that NDS meets the condition of an appropriate similarity function in the spectral clustering algorithm since local neighborhood induced by this similarity function is meaningful and robust to lighting and camera lens variations, object motion and editing effects. Furthermore, in other clustering schemes like k-means using Euclidean-based similarity metric, the triangle inequality is applicable. However, NDS as a content-based similarity function does not necessarily follow the triangle inequality property and the spectral clustering framework does not suffer due to this fact. Recall that triangle inequality states that the similarity between keyframes X and Z is at most as large as the sum of the similarity between keyframes X and Y and the similarity between keyframes Y and Z.

To address the second challenge of choosing a suitable number of clusters, we

Input: Weighted adjacency matrix $W \in \mathbb{R}^{n \times n}$, k number of clusters to construct

1. Construct the unnormalized Laplacian L as $L = D - W$, where D is the diagonal matrix whose (i,i)-element is the sum of W's i^{th} row.

2. Compute matrix $U \in \mathbb{R}^{n \times k}$, whose columns are the eigenvectors corresponding to the first k smallest eigenvalues of L.

3. Form the normalized eigenvector matrix Y by renormalizing each of U's rows to have unit length, $Y_{i,j} = \frac{U_{ij}}{(\sum_j U_{ij}^2)^{\frac{1}{2}}}$.

4. Use k-means clustering on the rows of Y to form k clusters of $C_1, C_2, ..., C_k$.

Output: Subgraphs $G_1, G_2, .., G_k$ with $G_j = \{v_i | Y_{i:} \in C_j\}$.

Figure 3.11: The unnormalized spectral clustering algorithm — We use W, explained in Equation (3.16) as the adjacency matrix.

utilize Within-Cluster Similarity (WCS) Score as

$$\text{WCS}(k) = \sum_{c=1}^{k} \frac{1}{G_c} \sum_{i,j \in G_c} E_{i,j}, k = 1, 2, .., \lfloor \frac{n}{2} \rfloor \tag{3.17}$$

where G_c denotes the cluster c, $|G_c|$ is the number of keyframes in G_c and n is the total number of keyframes in the story and also the eigengap heuristic [76] which is particularly designed for spectral clustering given by

$$\delta(k) = |\lambda_k - \lambda_{k+1}|, k = 1, 2, .. \left\lfloor \frac{n}{2} \right\rfloor, \tag{3.18}$$

where λ_k refers to the k smallest eigenvalues of graph Laplacian matrix (i.e. L in Figure 3.11). The number of clusters that maximizes Equation (3.17), K_{WCS}, and Equation (3.18), k_δ, are determined. In practice, we iterate the spectral clustering algorithm for $k = 1, 2, .., \lfloor \frac{n}{2} \rfloor$ and the appropriate number of clusters is then given by

$$K_T = \left\lfloor \frac{k_{\text{WCS}} + k_\delta}{2} \right\rfloor. \tag{3.19}$$

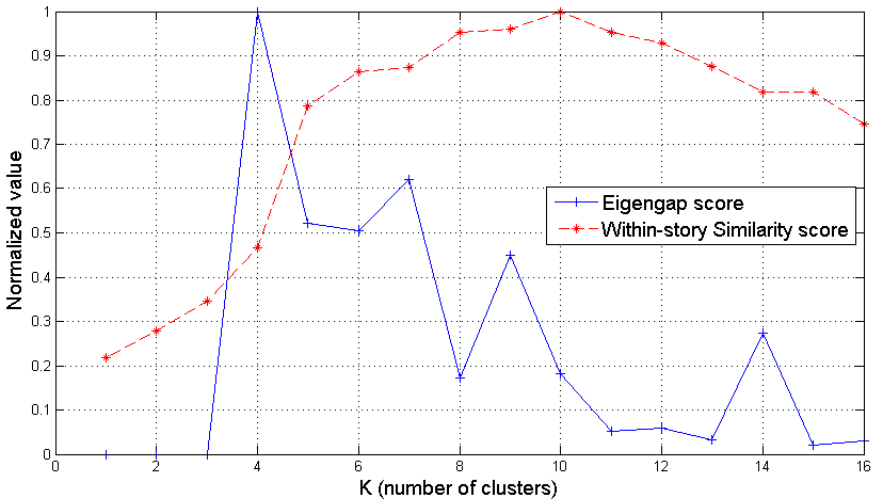

Figure 3.12: Variations of eigengap score, $\delta(k)$, and Within-Cluster Similarity score, WCS(k), with the number of clusters — They are determined using Equation (3.17) and (3.18), respectively.

Eigengap statistic is shown to work well when the clusters in data are well separated [76]. For instance in Figure 3.12, we plot the eigengap score for a different number of clusters for the news story shown in Figure 3.13. In this example, it turns out that after pruning the outliers according to Equation (3.16), we are left with three connected components, i.e., three disjoint subgraphs. This causes the associated graph Laplacian to have three zero eigenvalues. Thus, the eigengap is equal to zero for k from 1 to 3. $k_\delta = 4$ maximizes the eigengap score in this example.

However, since we deal with noisy and overlapping clusters in some news stories, we also use the WCS score which indicates normalized within-cluster similarities for a particular partitioning of keyframes for a specific k. It is expected that clusters obtained by maximizing WCS will be more balanced. More specifi-

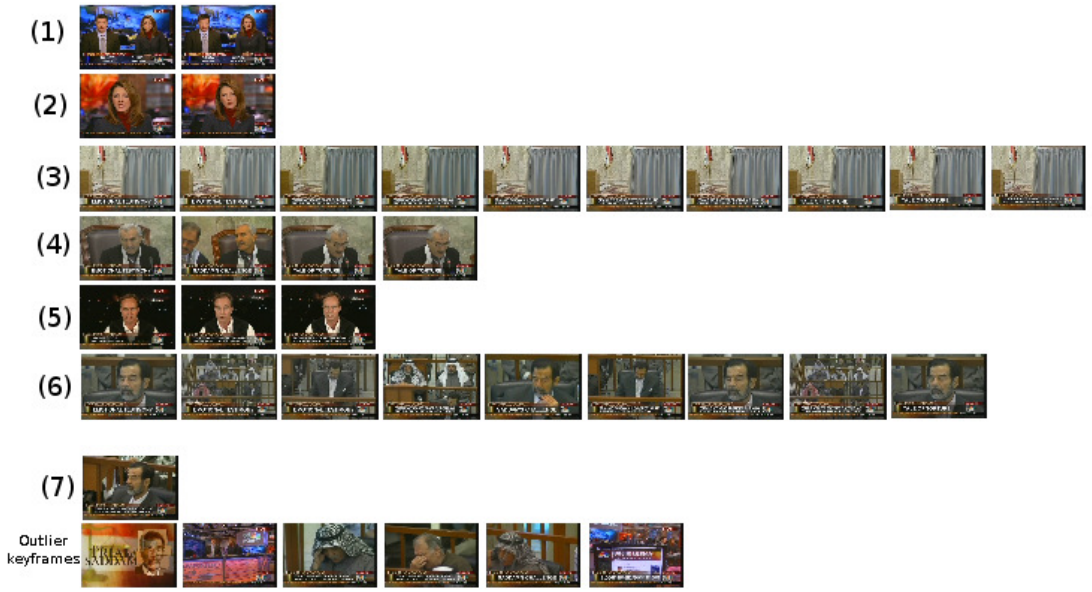

Figure 3.13: An example of our proposed keyframe clustering algorithm result — We could detect 7 NDK clusters and 6 outlier keyframes, shown in the last row.

cally, it avoids having numerous singleton clusters. In extreme case of n singleton clusters, WCS will be zero. The avoidance of singleton clusters is desirable in our case since we filter out outlier keyframes earlier through Equation (3.16) and the remaining keyframes are more likely to be part of reasonably large clusters.

As shown in Figure 3.12, k_{WCS} equals to 10 maximizes the WCS in this example. From Equation (3.19), we drive k_T equals to $\lfloor \frac{4+10}{2} \rfloor = 7$ as the number of clusters in this example. The clustering result is shown in Figure 3.13 where the last row indicates keyframes filtered out through Equation (3.16) and other rows indicate constructed clusters. The first, the second and the fifth clusters refer to anchors and reporter shots, while the other clusters depict different scenes captured in the "Trail of Saddam Hussein" event.

To demonstrate an application of the keyframe clusters, we generate a sto-

Figure 3.14: An example of a generated storyboard — (a) using clusters medoid, (b) extended storyboard including extra images for clusters with the low WCS.

ryboard for a news story. Storyboard refers to a set of images representing a summarized version of a video [27]. To do so, we find the medoid of each cluster, as the representative member of cluster. The generated storyboard for the story, mentioned in Figure 3.13, is shown in Figure 3.14(a). Note that in addition to filtered out keyframes, the singleton clusters like the 7-th cluster in Figure 3.13, is not considered for storyboard generation. For a smoother representation of the news story, we may select two or even more representative keyframes for the cluster with low within-cluster similarity like the 6-th cluster in the above example. This smoother version is shown in Figure 3.14(b).

Furthermore, we can employ the proposed keyframe clustering algorithm to detect anchorperson keyframes in a given news video footage, which can be later used to identify news story boundaries in the news video footage of interest. To do so, we use a common semantic style of the broadcast which is anchorperson(s) often appear at the beginning/end of a news story to present the prepaid materials related to the news story of interest. Applying the proposed keyframe clustering algorithm on a news video footage, we obtain a set of keyframe clusters, among which the cluster including keyframes, which are in average temporally far from

each other, can be potentially the anchorperson keyframe cluster. Detecting anchorperson keyframes, we can determine segments of news stories. In addition, we can also provide a better news story summarization by excluding anchorperson keyframes. For instance, in Figure 3.13(a) and (b) the first two keyframes can be omitted, using detected anchorperson keyframes.

3.4 Experimental Results

First, we assess the proposed configuration of the constrained keypoint matching method, explained in Section 3.1.2. Next, we evaluate the proposed NDK retrieval, detection and clustering methods, respectively.

3.4.1 Dataset

For the NDK retrieval and detection, we use Columbia dataset [125] and NTU dataset which is extracted from keyframes of TRECVID 2005 and 2006 corpus [28]. Each dataset includes 150 NDK pairs and 300 non-duplicate keyframes. For the NDK clustering, we use 20 news stories extracted from TRECVID 2006 videos. News stories are between two to five minutes length and are segmented manually as a group of keyframes. Keyframes are provided by National Institute of Standard and Technology (NIST) [2]. It includes a master keyframe as the middle I-frame of each shot and some auxiliary keyframes extracted from every two-second period of the video shots to cover more information within shots.

3.4.2 Selection of Threshold

In order to arrive at a suitable value of the threshold τ mentioned in Equation (3.7), and to evaluate the CSM algorithm, we implement the NDK retrieval task on

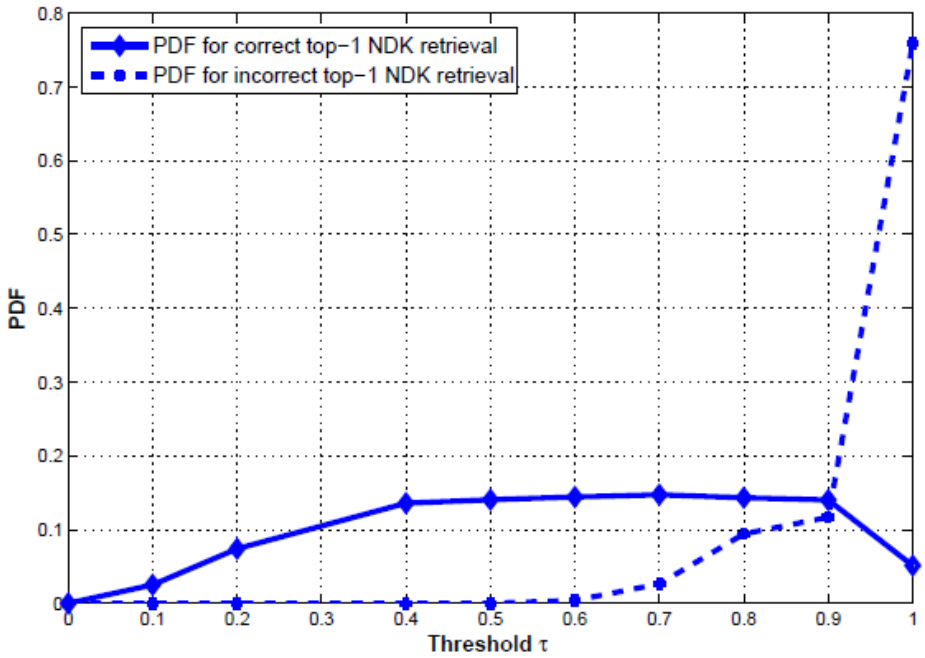

Figure 3.15: Selection of the threshold — PDF of τ for correct (solid line) and incorrect (dashed line) top-1 NDK retrievals.

NTU dataset. Figure 3.15 shows the Probability Density Function (PDF) of τ for correct and incorrect top-1 NDK retrievals. For a threshold of 0.9, about 85% of the incorrect retrievals are rejected; at the same time, about 15% of the correct retrievals are also rejected. The PC score is derived from the matching keypoints and hence, it is required that these be more reliable, i.e. the erroneous matching should be minimized. At the same time, the number of true matching keypoints should not be very small. Thus, there is a trade-off between reliability and number of true matches. Therefore, we set τ to 0.7 where about 97% of the wrong retrievals are rejected. In doing so, there might be pairs of keyframes that do not contain any matching keypoints. In such cases, the PC score cannot be calculated.

3.4.3 NDK Retrieval Evaluation

The retrieval experiment consists of using all the 300 keyframes from each dataset as queries so that each query is compared against 599 keyframes. The probabilities generated by the SVM are used to rank the other keyframes. The retrieval performance is quantified by the probability of retrieving a correct NDK in the top-k position of the ranked list given by [125]:

$$P(k) = \frac{Z_c}{Z} \tag{3.20}$$

where Z_c is the number of queries that rank their NDKs within the top-k position and Z is the total number of queries.

Table 3.1 shows the top-1 retrieval accuracy and the average retrieval accuracy of the top-5 retrievals on Columbia dataset. It can be seen that in NDK retrieval task, keypoint matching techniques like the proposed method, NIM method and OOS-SIFT method performs better than others that use block-to-block matching techniques like [67] and visual keyword method in [115]. The proposed method (CSM+PC) is comparable to NIM, which is the best result reported in this dataset. However, our method processes 7 pairs of keyframes per second, compared to their 10 frames per second. Our method outperforms [128] with a large margin, which confirms the effectiveness of the proposed CSM method compared to its matching algorithm (OOS) and also demonstrates the distinguishing power of PC score in NDK retrieval task.

In Figure 3.16(a) and (b), we demonstrate the top-30 retrieval results on Columbia and NTU datasets, respectively, using different features. The results show the effectiveness of LDA for feature weighting and justify choice of min instead of max of $(\mathrm{PC}_h, \mathrm{PC}_v)$. CSM performs better than max for the top-17

Table 3.1: Comparison of retrieval performance for top-1 and average of top-5 NDK on Columbia dataset.

Method type	Keypoint matching methods						Block matching methods	
Method	CSM	CSM+PC	OOS-SIFT [115]	LIP-IS+OOS [128]	NIM [133]	LDSS [28]	VK [115]	SPM [67]
Top-1 (%)	84.33	85.67	84.67	79.00	86.33	84.67	76.67	78.67
Average Top-5 (%)	87.93	88.62	88.06	83.06	88.53	87.59	80.73	80.86

retrievals after which the latter shows slightly improved performance. Training the SVM using the 3-D vector consisting of PC_h, PC_v, and CSM gives the worst top-1 and average of top-5 retrieval results on Columbia dataset. Hence, we can conclude that in order to retrieve NDK pairs, it is sufficient to check for alignment of keypoints in either vertical or horizontal direction, instead of mixing the quantized alignments as in [83].

Table 3.2 shows the top-1 retrieval accuracy (%) on NTU dataset using the proposed method and comparisons with the results reported in [28]. In addition, we implement the NIM method [133] on this dataset. The proposed method surpasses the LDSS method which is the best result reported on NTU dataset. There is a slight improvement when using the PC score in addition to the number of matching keypoints (CSM) compared to using only CSM because the number of matching keypoints is a meaningfully more discriminative feature than PC score as illustrated in Figure 3.8 and PC score affects the result only for a small group of samples located in the left side of plot in Figure 3.8 (i.e. samples with less number of matching keypoints). As mentioned earlier, our method is able to identify NDK pairs in the presence of extreme zooming and significant object motion. However, there are very few such instances in Colombia dataset. This explains why the performance on NTU dataset which does include large variations, is significantly higher and NIM method [133] does not perform as well in this dataset.

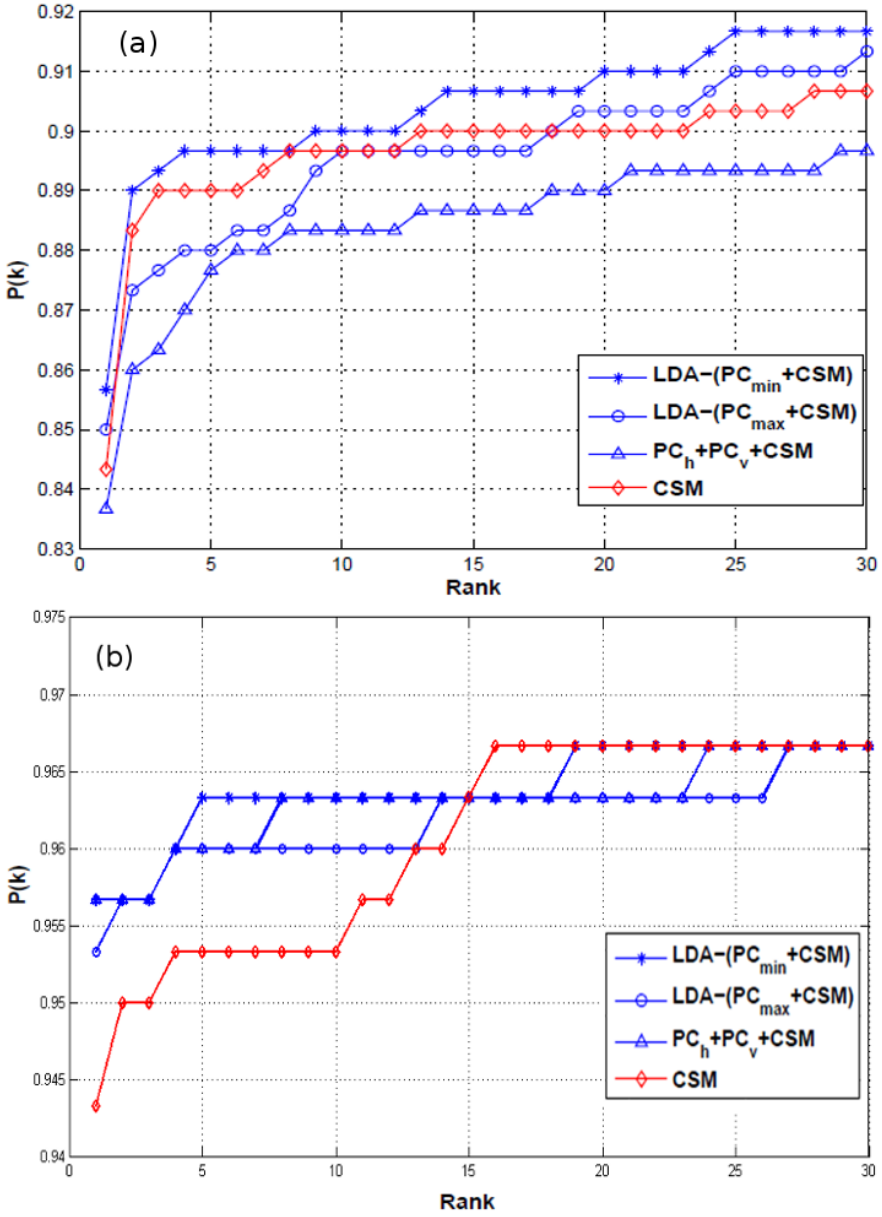

Figure 3.16: The top-k NDK retrieval results — on (a) Columbia and (b) NTU datasets for different features.

In NTU dataset, there are 18 NDK pairs which could not be correctly detected in top-1 via NIM method mostly due to zooming and object motion or combination of them as expected. However, the proposed method handles these

cases well. It should be mentioned that there are a few cases from NTU dataset that our proposed method fails to retrieve the NDK pair since no matching lines could be found between keyframes using CSM method. In these cases, the NIM method was successful.

Table 3.2: Comparison of retrieval performance for top-1 NDK on NTU dataset.

Method	CSM	CSM+PC	BOW [67]	LDSS [28]	NIM [133]
Top-1 (%)	94.33	95.67	88.00	92.00	89.67

Note that since other methods like NIM and OOS-SIFT are keypoint-based approaches, we also use only keypoint-based features, which are number of matching keypoints and PC score, for the sake of a fair comparison for the NDK retrieval evaluation.

3.4.4 NDK Detection Evaluation

In this part, we compare our proposed NDK detection method with other state-of-the-art methods addressing this problem. As baseline algorithms, we use the three distances computed at three independent levels from Spatial Pyramid Matching (SPM) method [67] , Temporal Pyramid Matching (TPM) method [118] and Spatially Aligned Pyramid Matching (SAPM) method [117] as input features, and further apply SVMs for classification. In SAPM+NCA and SAPM+GNCA, the authors used Neighborhood Component Analysis (NCA) and Generalized Neighborhood Component Analysis (GNCA) to convert a 45-dimensional feature into 3D space and then adopted an SVM classifier. Following the same training and testing framework as in [117], we randomly portioned the data into training and test sets. All experiments were repeated 10 times with different random training

and test samples, with mean and standard deviations reported in Table 3.3. In each run we used 60 positive and 240 negative samples for SVM training. The total number of positive and negative test samples are 90 and 4,840, respectively.

Table 3.3: Equal Error Rate (EER%) comparison of algorithms for NDK detection on Columbia and NTU datasets.

Method	SPM	TPM	SAPM	SAPM+NCA	SAPM+GNCA	CSM	CSM+PC	CSM+CS	CSM+CS+DoC
Columbia dataset	84.8±2.3	85.7±1.9	86.3±2.6	88.8±1.2	91.2±1.0	88.3±2.6	88.8±1.2	90.4±3.1	$\mathbf{92.2 \pm 2.1}$
NTU dataset	90.1±1.0	90.1±1.3	91.7±1.1	92.6±2.8	94.4±2.2	93.4±1.0	94±1.0	94.2±1.1	$\mathbf{95.9 \pm 1.2}$

As show in Table 3.3, our proposed NDK detection method using all three features of number of matching keypoints, color-based similarity and difference of complexity, outperforms the best result reported on Columbia dataset which is [117]. Even using only number of matching keypoints, we obtain better performance compared to SPM, TPM and SAPM. Adding color-based similarity as the second feature, we achieve 2% improvement in average. Finally considering DoC as the third feature, we observe the improvement of around 2%.

The proposed NDK algorithm outperforms the best result reported on NTU dataset as well. We also observe similar improvement by using the first feature, then the first two features and then all three features. The possible explanation for the overall better performance gained on NTU dataset compared to corresponding results on Columbia dataset is that we observe more discrimination between two classes of NDK and non-NDKs with respect to used features on NTU dataset as shown in Figure 3.10. In conclusion, these results show the effectiveness of our keypoint matching algorithm, and usefulness of DoC and color-based similarity features to handle cases in which matching keypoint feature is distorted or not conveying adequate information.

In another experiment, we incorporate the PC score, explained in Section 3.1.4,

and use LDA output of CSM and PC score (Section 3.1.5) instead of only CSM for NDK detection. It slightly improves the performance both on Columbia and NTU datasets compared to CSM results as shown in Table 3.3. However, the best detection result (i.e. CSM+CS+DoC) does not change significantly when we also consider PC score. The possible explanation is that the PC score improves the performance only for a small group of pairs of keyframes with less number of matching keypoints (located in the left side of plot in Figure 3.8), and color and DoC features contribute to a group of keyframes with low number of keypoints, as explained in Section 3.2.2, which can potentially have less number of matching keypoints as well. Therefore, PC score is not useful in the presence of color and DoC features.

3.4.5 Content-based Keyframe Clustering Evaluation

We utilize 20 stories extracted from TRECVID 2006 video. We compare our proposed keyframe clustering approach to another spectral clustering based approach [84] where Bhattacharayya coefficient of color histogram in RGB space is used to measure the visual similarity across the keyframes. They adopt the spectral clustering algorithm to group the keyframes with respect to the number of clusters determined. The proposed criteria for the number of clusters are that firstly, all generated clusters must have large enough eigengap and secondly, fraction of total weight of edges not covered by clusters must be less than a specific threshold. Although the spirit of our method and that of [84] is similar and based on spectral clustering, the proposed approach benefits from content-based visual similarity function and an effective strategy to choose a suitable number of clusters explained in Section 3.3.

To evaluate the quality of the clustering, we determine Precision Recall measures [77]. For this purpose, we use a set of classes in an evaluation benchmark produced by human observes with a good level of inter-observer agreement. Then according to human-assigned class labels, we index keyframes within all generated clusters. To evaluate how well our clustering method performs, we compute the contingency table shown in Table 3.4. To determine each of True Positive (TP), False Negative (FN), False Positive (FP), and True Negative (TN) values, we count all (number of) keyframe pairs belonging to the same/different cluster with the same/different class labels.

Table 3.4: Contingency table of NDK clustering.

	Same cluster	Different cluster
Same class	TP	FN
Different class	FP	TN

Computing contingency matrix for each story, we determine Precision (P), Recall (R), and F-measures based on the following equations.

$$P = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \tag{3.21}$$

$$R = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{3.22}$$

$$F = 2 \times \frac{P \times R}{P + R}. \tag{3.23}$$

In Figure 3.17, we illustrate box plot of these three measures for our proposed keyframe clustering approach and the color histogram based approach [84]. On

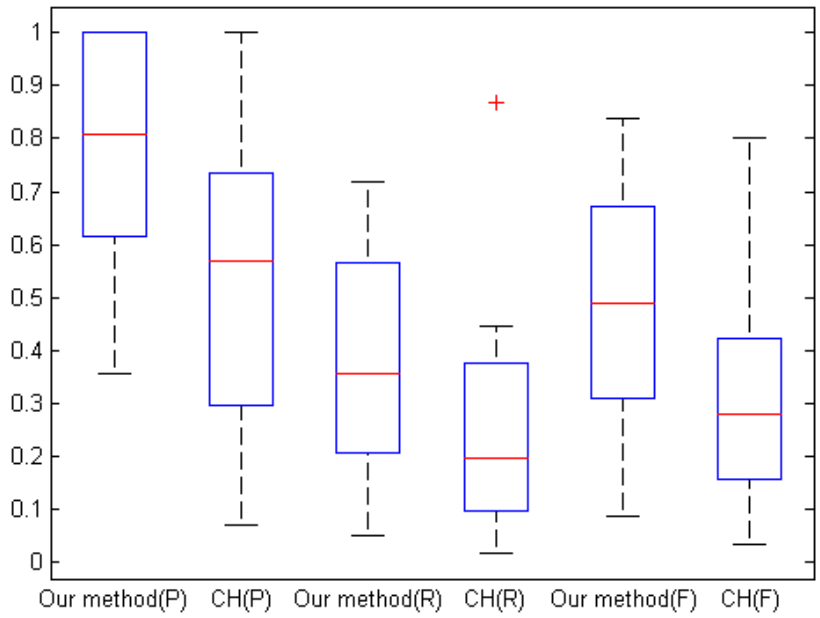

Figure 3.17: Precision, Recall and F-measure comparisons for the NDK clustering evaluation — CH method refers to the method proposed in [84].

each box, the central mark is the median, the edges of the box are the 25-th and the 75-th percentiles, the whiskers extend to the most extreme data points not considered outliers. Since the similarity function significantly affects the number of clusters, to study only the effectiveness of similarity function, we used ground truth number of clusters to implement spectral clustering in [84]. However, we use our own proposed strategy to choose a suitable number of clusters in our method. Our proposed keyframe clustering approach outperforms the other with a large-margin improvement of 24%, 16%, and 21% in Precision, Recall and F-measure, respectively, as indicated in Table 3.5. Our clustering method results in a better Recall (and consequently F-measure) compared to NDK detection method where we cluster keyframes together if they are detected as NDK using our NDK detection algorithm explained in Section 3.2.

Table 3.5: Precision, Recall and F-measure of our proposed method, NDK detection method and color histogram method for NDK clustering.

Method	Precision(%)	Recall(%)	F-measure
Proposed method	80.7	35.7	48.9
NDK detection method	80.1	27.7	41.2
Color Histogram method [84]	56.8	19.6	27.8

The possible explanation for relatively low values of Recall compared to Precision score in our approach is due to semantic gap between what our NDK identification algorithm can find as NDKs and the ground truth; there might be different clusters containing keyframes from the same class (e.g. clusters 1 and 2 in Figure 3.13). This fact ends up with a high value of FN in Table 3.4. Furthermore, due to the reliable similarity measure, we generally observe high purity within clusters (i.e. containing keyframes from the same class rather than different class) which leads to a low value of FP and accordingly a high value of Precision in our approach. In addition, the shorter rang of Precision in our approach shows that the proposed similarity is more robust and less sensitive to data rather than the color histogram approach, as shown in Figure 3.17.

In Figure 3.18, the keyframe clustering result based on [84] is shown. We use the same story represented in Figure 3.13. This method performs appropriately for keyframes which are roughly the same like the first three clusters. However, a poor discriminative power is observed when we deal with semantically distant keyframes with a similar color property like the fourth cluster which leads to a poor Precision. Moreover, another by-product of its general incapability to discriminate keyframes appears in filtering out irrelevant keyframes as shown in the last row in Figure 3.18.

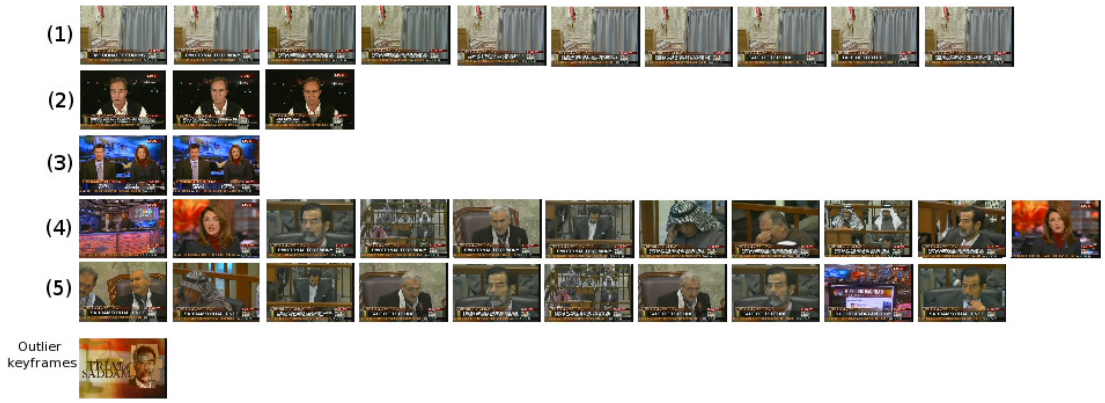

Figure 3.18: Keyframe clustering results based on [84] — Five NDK clusters and a single outlier keyframe were detected.

Finally, we examine our proposed approach to find the appropriate number of clusters (i.e. K_T) against the ground truth number of clusters (K_{GT}) determined by human. To compare these two sets of data, we perform a paired t-test of the null hypothesis that the data in the difference (i.e. $K_T - K_{\mathrm{GT}}$) are random samples from a normal distribution with mean zero and an unknown variance, against the alternative that the mean is not zero. It is generally agreed that if the sample size is greater than or equal to 20, the null distribution can be approximated by a normal distribution [94]. We compute t score for our 20 stories dataset as follows.

$$t = \frac{\bar{\mathbf{K}}_{\mathbf{D}}}{\mathrm{var}(\mathbf{K_D})/\sqrt{n}}, \tag{3.24}$$

where $\mathbf{K_D}(i) = \mathbf{K_T}(i) - \mathbf{K_{GT}}(i)$ for $i = 1, 2, ..., n$, and n is the total number of samples (i.e. $n = 20$). $\bar{\mathbf{K}}_{\mathbf{D}}$ denotes the average of $\mathbf{K_D}(i)$. We determine associated p-value equals to 0.0563 with respect to the degree of freedom of t statistic (i.e. $n - 1$). The test fails to reject the null hypothesis at the default value of the significance level $\alpha = 0.05$. Under the null hypothesis, the probability of observing a value at least as extreme of the test statistic, as indicated by

the p-value, is greater than α. In other word, the 95% confidence interval (i.e. $[-0.64, 0.14]$) on the mean contains 0. This result confirms the satisfying closeness of paired data points (i.e. $\mathbf{K_T}$ and $\mathbf{K_{GT}}(i)$ for $i = 1, 2, ..., n$).

3.5 Conclusion

In this chapter, we proposed a novel scheme for NDK identification and clustering. The number of matching keypoints and the Pattern Coherency (PC) score were learnt based on NDKs and non-NDKs provided by the Columbia and the NTU datasets. A trained SVM classifier was used to retrieve NDKs. The NDK retrieval evaluation results confirm the effectiveness of our keypoint matching method and usefulness of the PC score. In addition, we investigated two more features, namely color and DoC features, in the context of NDK detection problem and incorporated them to train an SVM classifier. These two features are helpful especially when we deal with NDKs with low number of keypoints.

Next, we used the SVM score as a measure of visual similarity of keyframe clustering using spectral clustering scheme. We determined the number of clusters based on within-cluster similarity and the eigengap heuristic. Quantitative assessment shows the proposed keyframe clustering algorithm performs well in terms of precision-recall scores and it also estimates the number of clusters, accurately. As a by-product, it also provides a content-based partitioning of a news story, which can be used to generate a storyboard presentation of the news story of interest.

Chapter 4

Scene Signatures for Unconstrained News Video Stories

The problem of an effective representation of a video sequence is important since it has a direct bearing on the performance of several tasks like content-based video retrieval, near duplicate video detection, topic detection and threading etc. Such a representation is called a video signature. In this chapter, we propose a robust and compact video signature for unconstrained news video analysis. Unconstrained news videos refer to a wide range of broadcast news videos from different types as explained in Section 1.2. The proposed scene signature is generated at the scene level as opposed to earlier approaches that were generated at the frame-level (i.e., using individual frames or keyframes)[110, 113, 114] or at the shot-level[132]. The scene-level video signature would enable much of the semantics to be encoded in the signature.

We address the challenging scenario like the one depicted in Figure 1.1(b) for which we need to generate a discriminative video signature. The figure shows

keyframes from similar news stories broadcast by ABC, NDTV, and CNN channels. Despite addressing the same topic, i.e. Bush press conference, the visual cues are significantly different. Furthermore, their lengths in terms of the number of keyframes and even their temporal order are different. They also include dissimilar visual contents irrelevant to the main topic, like anchorwoman and reporter. Such unconstrained news videos also contain significant variations in lighting conditions, object placement, viewpoints etc., causing tasks like near-duplicate video detection to fail.

The proposed scene signature (SS) aims to represent a news story in a compact as well as in a semantically meaningful way. It is essentially a group of SIFT descriptors. Although this seems to be a simplistic representation, through experiments in video retrieval, we show that it encodes much of the informative visual cues in a news story so that the semantic information is also captured.

The rest of this chapter is organized as follows. Section 4.1 explains the proposed approach to generate a scene signature. Section 4.2 describes keyframe sampling, feature extraction and NDK clustering methods. In Section 4.3, we explain the proposed keypoint processing applied on all keypoints within an NDK cluster. In Section 4.4 and Section 4.5, we explain scene signature generation and refinement steps, respectively. In Section 4.6, we conduct extensive experiments to show the effectiveness, robustness and compactness of the proposed scene signature compared to other video signatures.

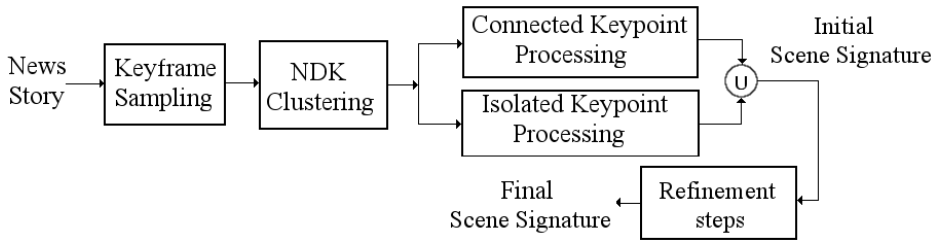

Figure 4.1: Our proposed framework for generation of scene signature.

4.1 Framework to generate scene signature

The proposed framework to generate a scene signature for a given news story is illustrated in Figure 4.1. The first step is to extract keyframes by sampling the video frames at a constant interval. Given the keyframes of a news story, we group the keyframes using NDK clustering. The SIFT keypoints in each cluster are categorized into connected keypoints and isolated keypoints i.e., they do not have any matching keypoints in other frames within the cluster. Note that isolated keypoints can be notably informative since in some shots/scenes, especially short ones which are so common in news story domain, connected keypoints are few and may not convey any visual information. These two categories are analyzed to generate an initial scene signature for each cluster. Through three refinement steps we merge some of the clusters which, in turn, are represented by more compact and discriminative scene signatures. As mentioned earlier, the scene signature is simply a collection of SIFT descriptors. We explain each of the blocks of the framework in the following sections.

4.2 Keyframe Sampling and NDK Clustering

4.2.1 Keyframe Sampling and SIFT Feature Extraction

Similar to other approaches relying on local features, the first step requires sampling of a news video. There are three types of sampling methods reported in the video retrieval literature, namely, uniform sampling, stable sampling (content-based sampling) and asymmetric sampling. In uniform sampling, video frames are sampled using a constant sampling rate while in the stable sampling method, first the original video is segmented into shots and then each shot is represented by one or a set of keyframes [40]. Since each shot contains homogeneous content, this method is referred to as content-based sampling. In asymmetric sampling, sampling results are integration of both uniform and stable sampling outputs [40].

We use the uniform sampling method with one frame-per-second sampling rate. While using uniform sampling leads to more sampled frames and subsequently more computational cost compared to stable sampling, it will not be affected by possible errors in shot boundary detection that might occur for stable sampling. For news videos that we encounter in this research like those with picture-in-picture style or with complex scene transitions (fading in/out or dissolving), detecting shot boundaries can be a challenging task.

Next, we extract SIFT keypoints [75] and corresponding descriptors from each extracted keyframe. Keyframes from different news videos with different quality (resolution) can have few to even more than 3000 keypoints under the default SIFT extractor configuration. However, keypoints located on logos or banners, which every news channel watermarks on its own published news stories, make NDK detection within a news story difficult since they can get matched across

all keyframes. Similarly, keypoints extracted from textual regions i.e., closed captions, subtitles or inserted static texts in the keyframe, can get matched across some keyframes including the same alphabets but not necessarily the same visual content. In order to remove such faulty matched keypoints, we determine a global mask for the entire news video and a local mask for each extracted keyframes. The global mask is determined to mark parts of the video frame containing channel logo or banners while the local mask marks parts of the keyframe wherein textual content exists.

To determine the global mask, we find segments of video frame that do not change significantly with time. For this purpose, first we select 15 keyframes using uniform sampling and mark pixels, gray-scale values of which are invariant and have very low standard-deviation along time to form the gray-scale mask. Since some banners and logos are transparent, they can not be marked accurately using their gray-scale values which might vary along time. Accordingly, we compute another mask to determine pixels with a constant edge value along time using Sobel edge detector [64], and take the OR it with the obtained gray-scale mask. The determined mask is dilated to fill holes. Note that even keypoints close to a logo or a banner are highly affected by their pixel values since by default the SIFT descriptor of a keypoint is determined in the 3-by-3 neighborhood basis. Applying the dilation, we also filter out these noisy keypoints to obtain a more informative set of keypoints. Conveniently, the method also removes keypoints close to black layout borders that are not very useful for matching. In Figure 4.2(c) and (d), black regions refer to determined global masks for two sample keyframes shown in Figure 4.2(a) and (b), respectively. Note that we deal with the solid CNN logo in Figure 4.2(a) and the transparent Al-Jazeera logo in Figure 4.2(b).

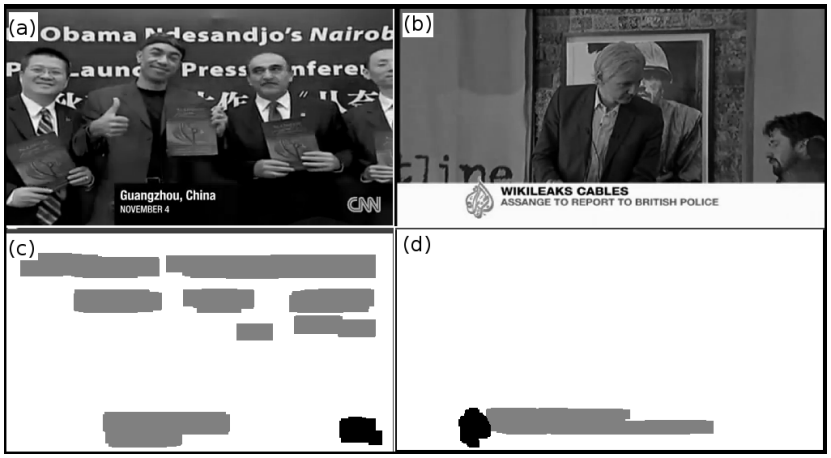

Figure 4.2: Global and local masks generation — (a) and (b) show sample keyframes with textual regions. In (c) and (d), black regions refer to the global masks and gray regions refer to the corresponding local masks.

Since typically closed captions change many times in a news video, they can not be detected using the global mask. Moreover, some news videos contain moving texts or transparent texts. Therefore, we employ a simple yet effective textual region detector that relies on the analysis of the spatial density of vertical edges within every single keyframe. In this method, first, vertical edges which indicates low-to-high or high-to-low transitions of gray-scale values of pixels within every image row, are detected using the Sobel operator [64] and binarized with respect to a predefined threshold. A pixel is classified as a part of a textual region if the density of the edges within the window, centered at the pixel of interest, is higher than a predefined threshold. Once all pixels are classified as text/non-text, the detected textual region is expanded by applying the dilation operator within every row to ensure a margin around textual regions and also to fill out holes between or within letters. Since the above method relies only on the presence of vertical edges, it works well for solid and transparent letters. It is also quite

effective in detecting both, subtitles inserted at the bottom of the frame, as well as other inserted text. In Figure 4.2(c) and (d), we show the local mask in gray for sample keyframes shown in Figure 4.2(a) and (b), respectively. We have used two different sets of parameters for the dynamic text detector. For the bottom (one-fourth) of the frame, where it is more likely to have text, we set the three parameters (thresholds controlling the edge binarization, edge density, and the length of the sliding window) so as to achieve high recall of text detection, while on the remaining part where it is less likely to have text, the parameters are set to achieve high precision. Since the closed caption text is expected to have smaller font than text inserted in the upper part of the frame, the window used for the bottom of the frame is smaller than the one used for the upper part.

For each keyframe, we take the AND of the global and local masks so that unwanted keypoints are filtered out. The remaining keypoints are ranked with respect to their scale values. The top-800 keypoints are chosen to represent the keyframe. This is because the larger scale indicates more informative and robust keypoints according to the SIFT descriptor extraction process [121]. Furthermore, shrinking the number of representing keypoints to 800 can decrease computational cost in the keypoint matching process.

4.2.2 NDK Clustering

The NDK clusters are formed by grouping together those keyframes which are detected as NDKs. A pair of keyframe are detected as NDKs if the number of matching keypoints, obtained using Constraint Symmetric Matching method explained in Section 3.1.3, exceeds a specific threshold. We do not incorporate other features such as color or DoC developed in Section 3.2, since the proposed

scene signature is based on the matching keypoints between keyframes in an NDK cluster and we want to make sure that detected NDKs, which form an NDK cluster, do have enough matching keypoints. Note that color and DoC features are independent to the number of matching keypoints. The former is based on the Bhattacharyya coefficient (Equation (3.12)) and the latter is based on the whole number of keypoints in keyframes (Equation (3.13)). If we incorporate them for NDK detection, it is possible to obtain NDK clusters where pairs of keyframes have high color or DoC similarity but not a significant number of matching keypoints which is not in favor of scene signature generation as mentioned earlier.

We use the *kd*-tree algorithm [16] for nearest neighbor search needed in our proposed method to speed up the matching process. Then we modify the candidate matching keypoints by applying $RANSAC$ algorithm [78] on matching patterns. Finally we use the number of matching keypoint as the only discriminative feature for the NDK/non-NDK classification problem.

Instead of brute-force pairwise comparisons across all extracted keyframes, we look for NDK pairs between two successive keyframes since they are more likely to be NDKs. Therefore, computational expense is reduced from $O(n^2)$ to $O(n)$ where n is the number of keyframes within a news story. Later in Section 4.5, through the proposed refinement steps applied on initial scene signatures, we union similar NDK clusters which are temporally far from each other.

4.3 Processing of SIFT keypoints

The scene signature is an attribute of an NDK cluster which is semantically meaningful and integrates all visual information inherent in the cluster. As mentioned

earlier, it is a group of SIFT descriptors. The first step in generating the scene signature is to categorize all keypoints within the keyframes in a cluster into two categories — connected and isolated. The former refers to matching keypoints that contribute to the NDK detection and therefore, they address the mutual visual cues in the NDK cluster. The latter refers to the rest of the keypoints that did not find any matches, yet they convey novel visual cues for the keyframes in the cluster. In the following, we analyze these two groups of keypoints, individually, to generate a compact and representative scene signature for each cluster.

4.3.1 Connected Keypoint Analysis

We track connected keypoints between keyframes within an NDK cluster to create a trajectory of connected keypoints. In an NDK cluster with n keyframes, a trajectory of connected keypoints can include at least 2 and maximally n keypoints. We represent each trajectory of connected keypoints, including $2 \leq m \leq n$ keypoints, by one of its keypoints with the largest scale and determine its degree as the length of the trajectory i.e., $m-1$. Note that the scale of each keypoint refers to the radius of the region associated to the keypoint of interest. It has been determined earlier along with the keypoint coordinates and orientation through the keypoint detection and the SIFT description computation. Next, we study co-occurrence of the connected keypoints in an NDK cluster. An NDK cluster will typically have a main object as shown in Figure 4.3(i). Line segments a, b and c in Figure 4.3(i) denote sets of matching keypoints between the frames where the line starts and ends, e.g., line segment a represents matching keypoints between each pair of keyframes. However, in an NDK cluster that contains keyframes with a split screen or a picture-in-picture format (which often exists in news videos)

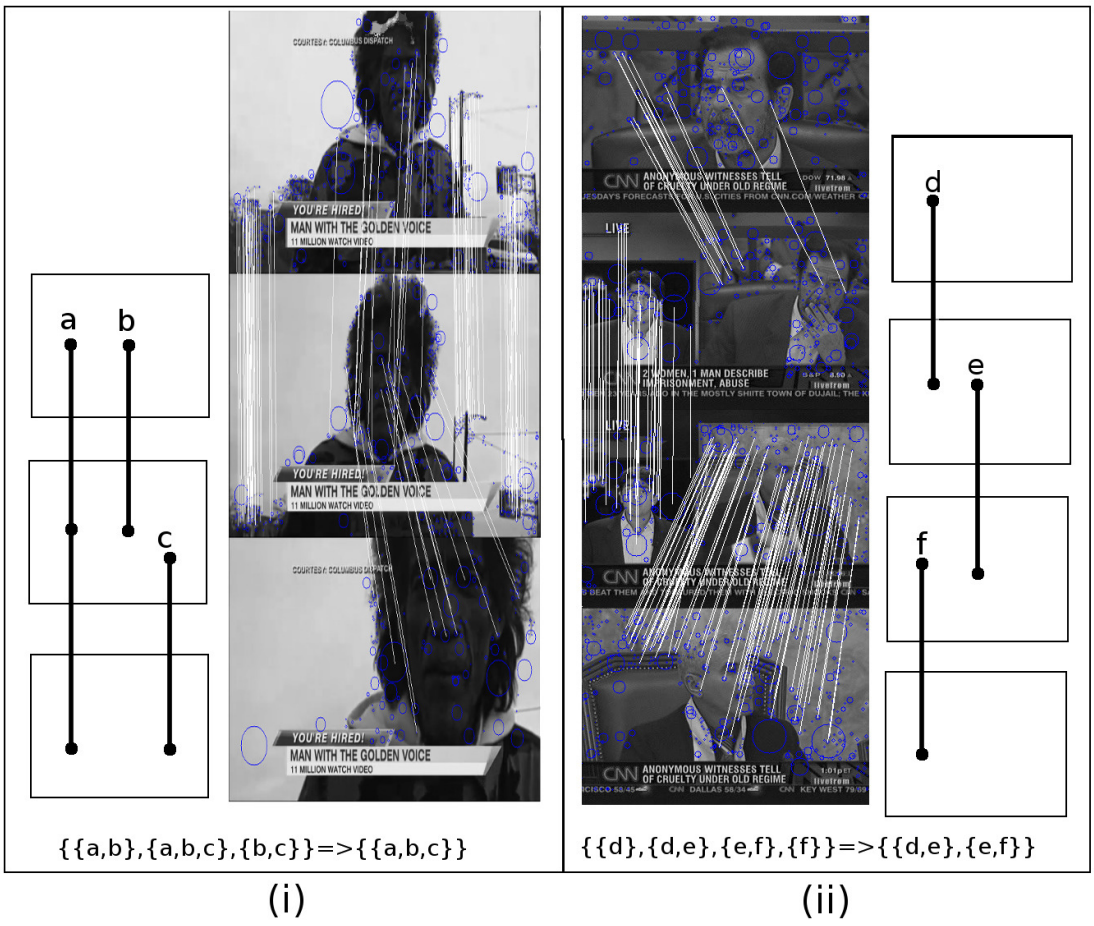

Figure 4.3: Connected keypoint analysis — (i) shows a typical example of inter-keyframe connections within an NDK cluster. (ii) shows connected keypoints arrangement within an NDK cluster with the picture-in-picture format. $a, b, c, d, e,$ and f are the matching keypoint sets between keyframe pairs.

or with large camera or object motion, there may not be matching keypoints between every pair of keyframes. For instance, in Figure 4.3(ii) the visual cues carried by line segments d and f are irrelevant to each other. Thus, we can divide a scene containing a large number of keyframes into sub-scenes with coherent visual content and take into account the co-occurrence of keypoints to generate a more precise scene signature.

C_k: candidate keypoint-set of size k
L_k: frequent keypoint-set of size k
L_1 = { connected keypoints };
for ($k = 1; L_k \neq \emptyset; k++$) do begin
    C_{k+1} = candidates generated from L_k;
    for each transaction t in database do increment
        the count of all candidates in C_{k+1} that are contained in t
    L_{k+1} = candidates in C_{k+1} with min-support (i.e. 1)
    end
return $\cup_k L_k$;

Figure 4.4: Finding maximal patterns in the keypoint sets of the NDK cluster based on the *Apriori* algorithm [11].

To study co-occurrence of the connected keypoints in an NDK cluster, we employ the concept of frequent itemset pattern identification. For each cluster, we determine a transaction database where its items are all connected keypoints and its itemsets are keyframes within the cluster. We find the maximal pattern for each cluster transaction database considering the minimum support of one using *Apriori* algorithm [11] and use them to consider the co-occurrence of connected keypoints as shown in Figure 4.4. Returning to Figure 4.3, the maximal pattern set for Figure 4.3(i) and (ii) are $\{a, b, c\}$ and $\{\{d, e\}, \{e, f\}\}$, respectively. Thus, we obtain a set of keypoints that represents a particular visual content in the corresponding scene better. Note that if we simply consider all connected keypoints together to generate a scene signature, we ignore the fact that some of them did not appear together. In Section 4.4, we utilize this co-occurrence information inherent with max-patterns for each NDK cluster to determine similarity between scene signatures and between a scene signature and a keyframe.

4.3.2 Isolated Keypoint Analysis

As stated earlier, isolated keypoints refer to keypoints that have not been matched through NDK clustering process and we assign zero degree to them. As shown in [132], only about 50% matches are found between the keyframe and its next frame on an average, and no more than 20% of the keypoints are matched with each other on an average from 8 successive frames. These ratios can get lower in shots/scenes with a fast motion or a significant object displacement. To cope with this insufficiency, we aim to enrich the scene signature by incorporating informative isolated keypoints. Isolated keypoints arise either because there is indeed no other matching keypoints in other keyframes, or the NDK detection algorithm fails to identify other matching keypoints due to lighting changes, partial occlusion etc. In either case, isolated keypoints possess valuable visual information which could even be the most informative visual content. For instance, in Figure 4.5(b), the isolated keypoints marked in blue circles, most of which describe the news subject (i.e. Julian Assange), are semantically more meaningful compared to the connected keypoints on the background, which are from a text as shown in Figure 4.5(a), which is not highly related to the news subject. In such cases, ignoring the isolated keypoints and solely considering the connected keypoints can lead to an ineffective scene signature since a desired scene signature must cover all visual contents in a scene.

Since roughly more than 80% of the keypoints are isolated keypoints, it is important to select the most discriminative one among them. They should be spatially as far away as possible from the keypoints that have been matched. In other words, isolated keypoints near a bunch of matching keypoints do not carry

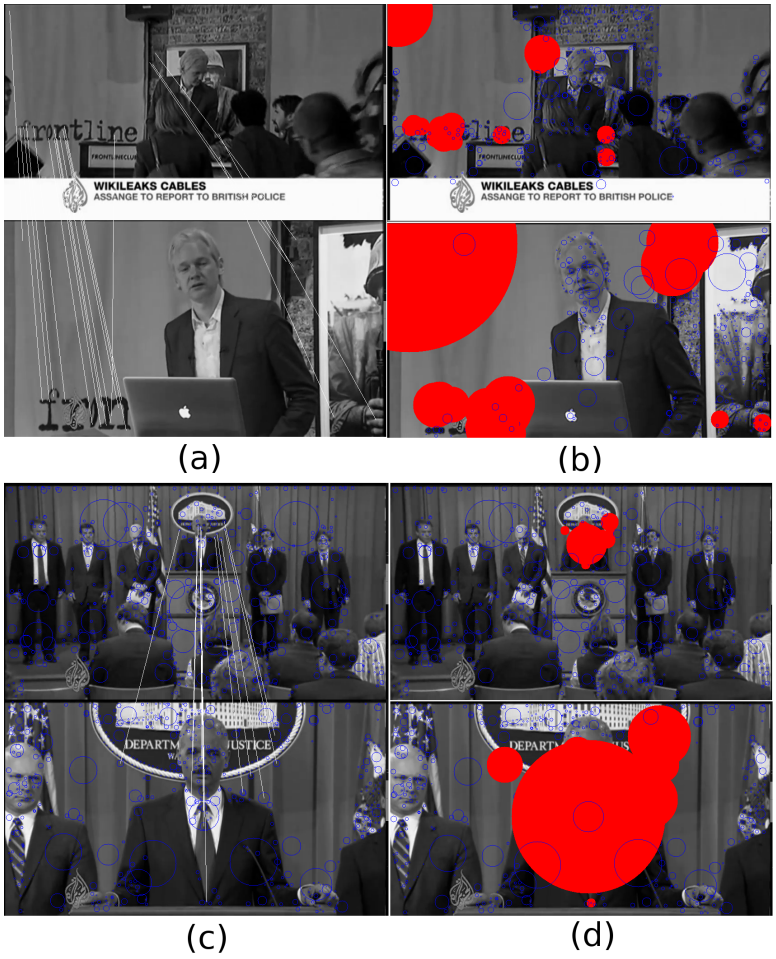

Figure 4.5: Isolated keypoints selection — (a) and (c) show the matching lines between connected keypoints in a pair of keyframe. (b) and (d) show connected keypoints neighborhoods by the red solid circles and isolated keypoints by the blue circles.

sufficient discriminative information. Hence, we need to identify a neighborhood of connected keypoints from which the isolated keypoints must be removed. Figure 4.5(a) shows a pair of keyframes that belongs to an NDK cluster of a particular news story and their matching keypoints. Each matching keypoint is depicted as a red solid circle in Figure 4.5(b) whose radius is proportional to the scale. We remove isolated keypoints from within the circle, which serves as the

neighborhood. The union of the remaining isolated keypoints in each keyframe in the NDK cluster gives the final set of isolated keypoints. Figure 4.5 (c) shows another pair of NDK where the neighborhood of matching keypoints are shown in the red circles in Figure 4.5(d).

4.4 Scene Signature Generation and Similarity

An initial scene signature is generated by first defining a fixed budget for the connected keypoints (i.e. $N_c = 800$) and determining a dynamic budget for isolated keypoints as

$$N_i = \min(\max(0, 400 - 0.4 \times n_c), n_i), \tag{4.1}$$

where n_c and n_i are the number of connected keypoint representatives and isolated keypoints, respectively. N_i can be varied from zero (when $n_c \geq 1,000$) up to 400 (when $n_c = 0$). Note that we choose the weight and constant value in Equation (4.1) to assure that the generated signature will have at least 400 keypoints which is a safe number to represent a visual unit as discussed in [121], and it will not include many isolated keypoints when there are large enough number of connected keypoints. The importance of the i-th keypoint is determined by

$$S_i = d_i + \text{scale}_i / \max(\text{scale}_j), j = 1, 2, .., n_c \text{ or } n_i, \tag{4.2}$$

where d_i and scale_i refer to the degree and the scale of the keypoint, respectively. d_i is equal to zero for isolated keypoints as mentioned earlier. We simply rank connected keypoints representatives and isolated keypoints based on their scores and pick the best N_c and N_i keypoints, respectively, to form a scene signature.

To compute the similarity between scene signatures SS_1 and SS_2, we take co-occurrence information (explained in Section 4.3.1) into account and determine

the co-occurrence matrix $\mathbf{C} = [c_{ij}]_{n_{\text{all}} \times n_{\text{mp}}}$ for each generated scene signature where n_{all} refers to all keypoints within the NDK cluster and n_{mp} indicates the number of detected max-patterns and

$$c_{ij} = \begin{cases} 1 & \text{if } (\text{KP}_i \in \text{MP}_j) \text{ or } (\exists \text{KP}_m \in \text{MP}_j | \text{KP}_m \in \text{KF}_i), \\ 0 & \text{otherwise} \end{cases} \tag{4.3}$$

where KP_i indicates the i-th keypoint in the NDK cluster and MP_j refers to the j-th max pattern and KF_i indicates the keyframe containing KP_i. For instance, the co-occurrence matrix for Figure 4.3(i) has a single column of ones while the co-occurrence matrix for Figure 4.3(ii) has two columns marking the first and the second max-pattern sets of keypoints (i.e. *d,e* and *e,f*) and the corresponding isolated keypoints.

Next, we determine affinity matrix $\mathbf{S} = [s_{ij}]_{n_1 \times n_2}$ where n_1 and n_2 are number of keypoints in SS_1 and SS_2, respectively. s_{ij} is equal to 1 if the i-th keypoint from the first scene signature is matched with the j-th keypoint from the second scene signature. The final similarity score between two scene signatures is determined as:

$$\text{Sim}(\text{SS}_1, \text{SS}_2) = 1 - e^{-\frac{\max(\mathbf{C_1^T} \times \mathbf{S} \times \mathbf{C_2})}{\tau_s}}, \tag{4.4}$$

where $\mathbf{C_1}$ and $\mathbf{C_2}$ are $n_1 \times \text{np}_1$ and $n_2 \times \text{np}_2$ co-occurrence matrices of SS_1 and SS_2, np_1 and np_2 are number of max-patterns of SS_1 and SS_2, respectively. We experimentally set τ_s to 17. We consider the two scene signatures near-duplicate if their similarity is greater than 0.5. To determine similarity between a scene signature and a keyframe, we use Equation (4.4) by considering a unit vector $(n_2 \times 1)$ as $\mathbf{C_2}$ where n_2 is the number of keypoints in the keyframe of interest.

In some cases where we could not find any NDK within the news story, there is no NDK cluster for scene signature generation either. This often happens in

short news stories in which shots change quickly. In such cases we consider all the keypoints in each extracted keyframe as a scene signature and follow the procedure, accordingly.

4.5 Refinement of Initial Scene Signature

In some news stories, there are NDKs that are temporally far from each other and which would not be clustered together through the NDK clustering scheme, proposed in Section 4.2.2, since we limited the exploring area to two successive keyframes. Considering the fact that a scene signature is basically a bag-of-SIFT, we can use the same NDK clustering approach stated in Section 4.2.2 for scene signature clustering as well. Therefore, we utilize the same keypoint matching method to match keypoints across scene signatures. After clustering near-duplicate scene signatures (i.e. scene signatures sharing adequate matching keypoints), we can generate the second generation of scene signatures similar to the first generation but with different configuration.

A further refinement of the scene signatures is done by re-clustering all keyframes again using the second generation of scene signature as the cluster centroids and soft-assigning each keyframe to the clusters with the similar centroids. We consider a keyframe and a scene signature similar if the determined Sim score in Equation (4.4) is greater than 0.5. This step results in a more enriched and semantic cluster since some similar keyframes could not be clustered together in the earlier stages. This semantic improvement is obtained by the integration of relevant visual cues obtained in the second generation of scene signatures. In Figure 4.6(a), the original NDK clustering result is shown. In Figure 4.6(b), a

Figure 4.6: An example of scene signature refinements — (a) NDK clusters, (b) clustering using initial scene signatures, and (c) re-clustering after scene signatures merging, (d) news story summarization using a set of thumbnails.

copy of a keyframe from cluster 2 is added to cluster 1 since that keyframe was found semantically similar to the scene signature of cluster 1. Now that there exists a link between cluster 1 and cluster 2, they are merged together as shown in Figure 4.6(c) and a new scene signature is generated for the merged cluster.

The final refinement step involves the transitivity property of keyframes within a cluster. A binary relation R over a set X is transitive if a related to b, and b related to c, implies a is related to c. Near-duplicity of keyframes is a binary relation between two keyframes which inherently has transitivity property. As mentioned in [83], given two pairs of NDKs (k_1, k_2) and (k_2, k_3), one may infer the NDK identity of (k_1, k_3). Figures 4.7(a) and 4.7(b) show examples of transitivity applicable to the keyframes illustrated, and similar to [83] we treat them as true positive of an NDK pair.

Here we expand the idea of transitivity property over NDKs to NDK clus-

Figure 4.7: Transitivity property of NDKs — All keyframes in row (a) or row (b) are NDKs [83].

ters where we assume that clusters sharing the same keyframes are associated to each other and their scene signatures can be merged, accordingly, to obtain a more semantic and compact scene signature. We re-use the first step of the refinement procedure, explained in the first paragraph of this section to merge corresponding scene signatures. In Figure 4.6(c), we show the final NDK clustering result after applying the transitivity property and merging the corresponding scene signatures.

Another example of the proposed scene signature refinement is shown in Figure 4.8. Figure 4.8(a) shows the initial NDK clustering result. Figure 4.8(b) and (c) show the NDK clustering results using scene signatures before and after the refinement steps. However, there are still some semantic gaps that could not be bridged using scene signatures. For example, the second and the sixth cluster in Figure 4.8(c) have not got merged while they are captured from the same scene. However, the final clusters clearly are more meaningful compared to the original clusters of near-duplicate keyframes.

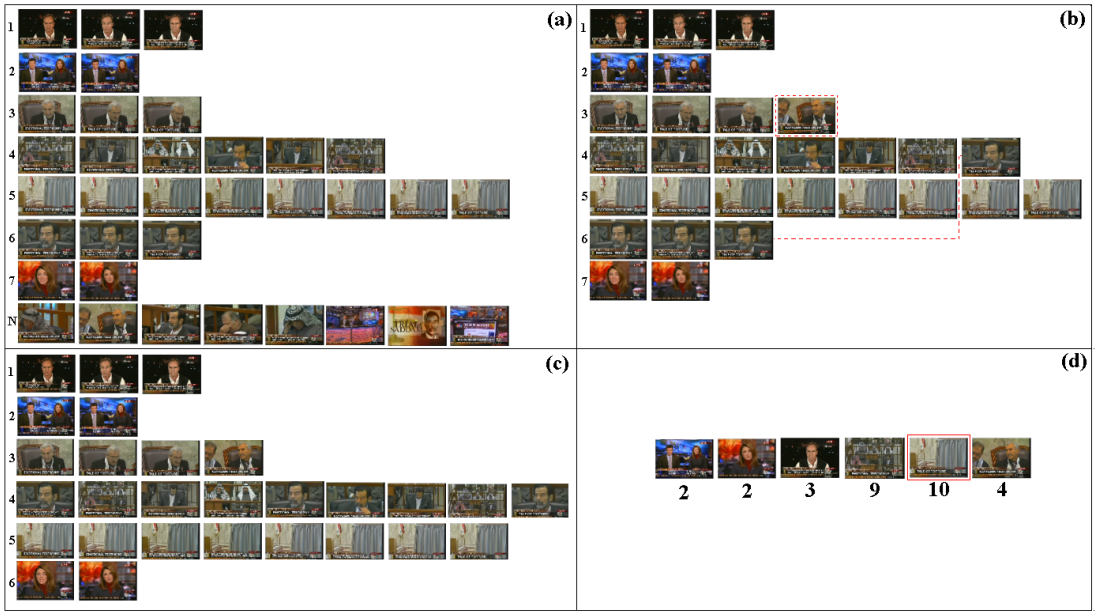

Figure 4.8: Another example of scene signature refinements — (a) NDK clusters, (b) clustering using initial scene signatures, and (c) re-clustering after scene signatures merging, (d) news story summarization using a set of thumbnails.

4.6 Experimental Results

In this section, first we investigate the limitations of keypoint aggregation in representing a shot/scene of a video and explain how we suppress these restrictions through our keypoints analysis process. Second, we evaluate the distinguishing power of our proposed scene signature against other global and local signatures. We also illustrate the efficiency of the proposed scene signature and explain the role of keypoint budgeting to obtain a unique and compact representation of a news story. Next, we assess our proposed scene signature performance through the retrieval task. Finally, as another application of scene signatures, we use them to generate a storyboard representation of a news story.

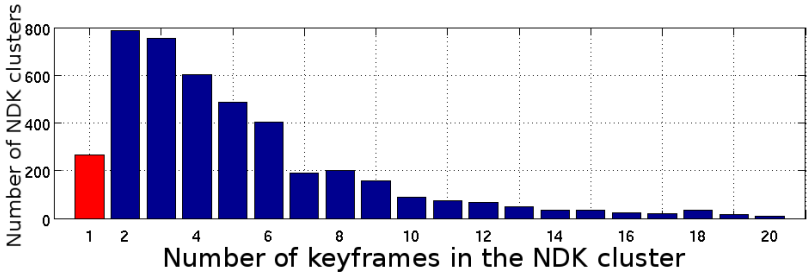

Figure 4.9: Distribution of the number of detected NDK clusters including different number of keyframes — Most of the NDK clusters contain 2 or 3 keyframes.

4.6.1 Dataset

For scene signature evaluations and the keyframe cluster retrieval experiment, we gather a dataset containing of 100 news stories from different channels downloaded from YouTube [5] in February 2011. We sampled each video with a sampling rate of 1 fps. They range from 2 to 5 minutes covering world events. Figure 4.9 illustrates the distribution of the number of detected NDK clusters including different number of keyframes in our dataset. Note that the NDK clusters with one keyframe refer to the outlier keyframes for which we could not find a near duplicate within a news story. The number of NDK clusters decreases exponentially as the number of keyframes increases.

For the associated news story detection task, we use news videos from one month in the TRECVID 2006 dataset including 830 news stories out of which there are 132 similar news videos belonging to the first and the second categories of associated news stories as in Figure 1.1(a) and (b). We manually segment the news stories as a group of keyframes and label them based on their main topic.

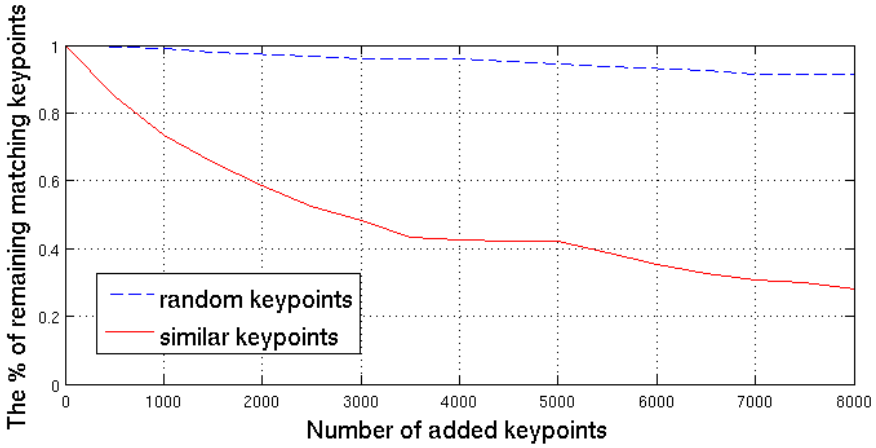

Figure 4.10: The effect of keypoint aggregation on keypoint matching performance — Aggregation of similar keypoints degrades the keypoint matching performance, dramatically.

4.6.2 The Effect of Keypoint Aggregation on Keypoint Matching Performance

In this section, we show how aggregation of keypoints into a single group of SIFT descriptors can affect the keypoint matching process and, consequently, detected NDKs and matched scene signatures. To do so, for every pair of NDK detected in our dataset (around 17,000 NDKs) we add a number of random keypoints and then count the number of matching keypoints while the number of added keypoints increases. As show in Figure 4.10, the number of matching keypoints decreases when the number of added keypoints increases since the probability of being matched can decrease for a particular keypoint.

We conduct a similar experiment but instead of random keypoints we add similar keypoints to NDK pairs. The similar keypoints come from the other keyframe which are in the same NDK cluster as NDKs of interest. As shown in Figure 4.10, similar to the previous trend, number of matching keypoints decreases but with

a higher ratio as expected since through keypoint matching process, initially detected matches may not be detected when initial keypoints gathered together with similar keypoints into the same group of SIFT descriptors.

These observations show the essence of connected keypoint processing step to represent the connected keypoint trajectory by a single descriptor, and also isolated keypoint processing step to retain more informative keypoints out of all isolated keypoints. Note that by taking these two steps, not only we can match keypoints across scene signatures, more effectively and precisely, but it also improves the compactness of scene representation.

4.6.3 Discriminative and Compactness Analyses of Scene Signature

In this section, we cluster NDKs in each news story using different video signatures. We compare the clustering performance of our proposed scene signature with other global and local video signatures listed in Table 4.1. Global signatures considered are gray-scale histogram (GH), color histogram (CH) and edge orientation histogram (EOH). The local signatures are (i) bag of SIFT description extracted from each keyframe (BOSK), (ii) bag of SIFT extracted from each NDK cluster (BOSC), (iii) bag of averages of connected keypoints descriptions in each NDK cluster (ACKP), (iv) bag of connected keypoints representative in each NDK cluster (CKPR) as determined in Section 4.3.1, and (v) the final scene signature (MSS) obtained in Section 4.5. The dissimilarity between global signatures is measured by Bhattacharya coefficients and cosine similarity. The dissimilarity between SIFT-based signature is measured using an exponential function as stated in Table 4.1. Using keyframe-level signature such as GH, CH, EOH and

Table 4.1: Global and local video signatures and their descriptions and dissimilarity measures

Symbol	Global / Local	Signature	Description	Dissimilarity measure
GH	Global	Gray-scale histogram	normalized 32-dimensional intensity histogram	$d(H_1, H_2) = 1 - \sqrt{1 - \sum_I \frac{\sqrt{H_1(I).H_2(I)}}{\sqrt{\sum_I H_1(I).\sum_I H_2(I)}}}$
CH	Global	Color (RGB) histogram	normalized 96-dimensional color histogram	$d(H_1, H_2) = 1 - \frac{\sum_I H_1(I).H_2(I)}{\sqrt{\sum_I H_1^2(I).\sum_I H_2^2(I)}}$
EOH	Global	Edge Orientation histogram	(16 × 8)-dimensional edge orientation histogram extracted from 4 × 4 blocks	
BOSK	Local	Bag-of-SIFT per Keyframe	n keypoints described by 128-dimensional SIFT descriptor	$d(B_1, B_2) = e^{-\frac{n}{12}}$ where n is the number of matching keypoints
BOSC	Local	Bag-of-SIFT per NDK cluster		
ACKP	Local	Average of Connected keypoints per NDK cluster		
CKPR	Local	Connected keypoints Representative per NDK cluster		
MSS	Local	merged scene signatures per NDK cluster		$d(SS_1, SS_2) = 1 - Sim(SS_1, SS_2)$

BOSK, we cluster two keyframes together if their keyframe-level dissimilarity is lower than 0.5. Regarding the scene-level signatures such as BOSC, ACKP, CKPR and MSS, we use generated signatures as the cluster centroids and cluster all keyframes within a news story based on their distance to centroids computed as $e^{-\frac{n}{12}}$ where n is the number of matching keypoints.

To assess the NDK clustering performance, we measure cohesion and separation of clustering results. We compute the mean and variance of within-cluster sum of squares (WSS) and between-cluster sum of squares (BSS) as

$$\text{WSS} = \frac{1}{n_C} \sum_{i=1}^{n_C} \frac{1}{|C_i|} \sum_{x \in C_i} d^2(S_i, x), \tag{4.5}$$

$$\text{BSS} = \frac{1}{n_C} \sum_{i=1}^{n_C} \frac{1}{n_C - 1} \sum_{j \neq i}^{n_C} d^2(S_i, S_j), \tag{4.6}$$

where x refers to a keyframe. $|C_i|$ is the size of the cluster i. S_i is the centroid of the cluster i. $d(,)$ is the dissimilarity measure determined for each

Table 4.2: WSS, BSS, and WSS-to-BSS ratio for different video signatures

Signature	GH	CH	EOH	BOSK [124]	BOSC	ACKP [132]	CKPR	MSS
WSS	0.17 ± 0.16	0.17 ± 0.18	0.16 ± 0.15	0.22 ± 0.33	**0.12 ± 0.29**	0.21 ± 0.33	0.17 ± 0.34	0.16 ± 0.33
BSS	0.56 ± 0.17	0.50 ± 0.57	0.64 ± 0.36	**0.92 ± 0.08**	0.63 ± 0.25	0.85 ± 0.13	0.86 ± 0.14	0.85 ± 0.13
WSS/BSS	0.30	0.34	0.28	0.24	0.21	0.24	0.20	**0.18**

signature from Table 4.1. n_C refers to the number of clusters. Note that for keyframe-level signature, we replace $\sum_{x \in C_i} d^2(S_i, x)$ with $\frac{1}{|C_i|} \sum_{x,y \in C_i} d^2(x, y)$ in Equation (4.5). Similarly, in Equation (4.6), we replace $\frac{1}{n_C - 1} \sum_{j \neq i} d^2(S_i, S_j)$ with $\frac{1}{|C_i| \times (n - |C_i|)} \sum_{x \notin C_i, y \in C_i} d^2(x, y)$ where n is the number of all keyframes within the news story. We also compute the ratio WSS to BSS.

Table 4.1 lists these values for different signatures used on the dataset described in Section 4.6. The lowest WSS-to-BSS ratio belongs to MSS while the highest BSS belongs to BOSK. Although the lowest WSS belongs to BOSC, it is not discriminative and its corresponding BSS is relatively high. Among the global signatures, EOH performs reasonably well and GH signature outperforms CH both in terms of BSS and WSS-to-BSS ratio.

To study discriminative characteristic of local signatures further, we determine their performance in terms of BSS and WSS for NDK clusters with different number of keyframes. As shown in Figure 4.11(a), WSS decreases with increasing the number of keyframes for all local signatures except for BOSK since they are based on NDK clusters and hence have more matching keypoints. WSS for MSS and CKPR get closer for NDK cluster with the higher number of keyframes. Because as the number of keyframes in a cluster increases, the number of connected keypoints increases which leads to a lower budget for isolated keypoints based on Equation (4.1). It results in a high similarity between CKPR and MSS performance in NDK clusters with a higher number of keyframes. However, due to

inclusion of isolated keypoints, MSS shows less WSS for NDK clusters with a low number of keyframes, while CKPR and ACKP perform worse than even BOSK. By increasing the number of keyframes within NDK cluster, WSS for ACKP gets higher than CKPR as shown in Figure 4.11(a), since connected keypoints trajectory will be larger and, consequently, taking average of the connected keypoints can result in an artificial descriptor which does not describe the connected keypoints batch, properly. However, in CKPR, the representative of the connected keypoints is the one with the largest scale and is always valid.

Although the lowest WSS belongs to the BOSC, it also has significantly low BSS which makes it a less discriminative signature. This may be caused by the large number of keypoints that BOSC has which can lead to generally higher number of matching keypoints. On the other hand, BSS for other local signatures have the slightly decreasing trend from 0.91 to 0.84 out of which BOSK has the most stable and the highest values for NDK clusters with more than 4 keyframes. For NDK clusters with less than 4 keyframes, ACKP and CKPR obtain the highest BSS, respectively, which can be explained by more related and relatively low number of keypoints they have, which leads to a low number of matching keypoints between two irrelevant NDK clusters.

To study the compactness of the proposed scene signature, we show the distribution of keypoints with different degree within scene signatures extracted from NDK clusters with different number of keyframes in Figure 4.12(a). For all NDK clusters, most of the information is contained in the first-degree keypoints. Figure 4.12(b) shows the same plot after imposing the budget determined for the connected and isolated keypoint as explained in Section 4.4. Depending on the number of keyframes within an NDK cluster, we could compress the visual infor-

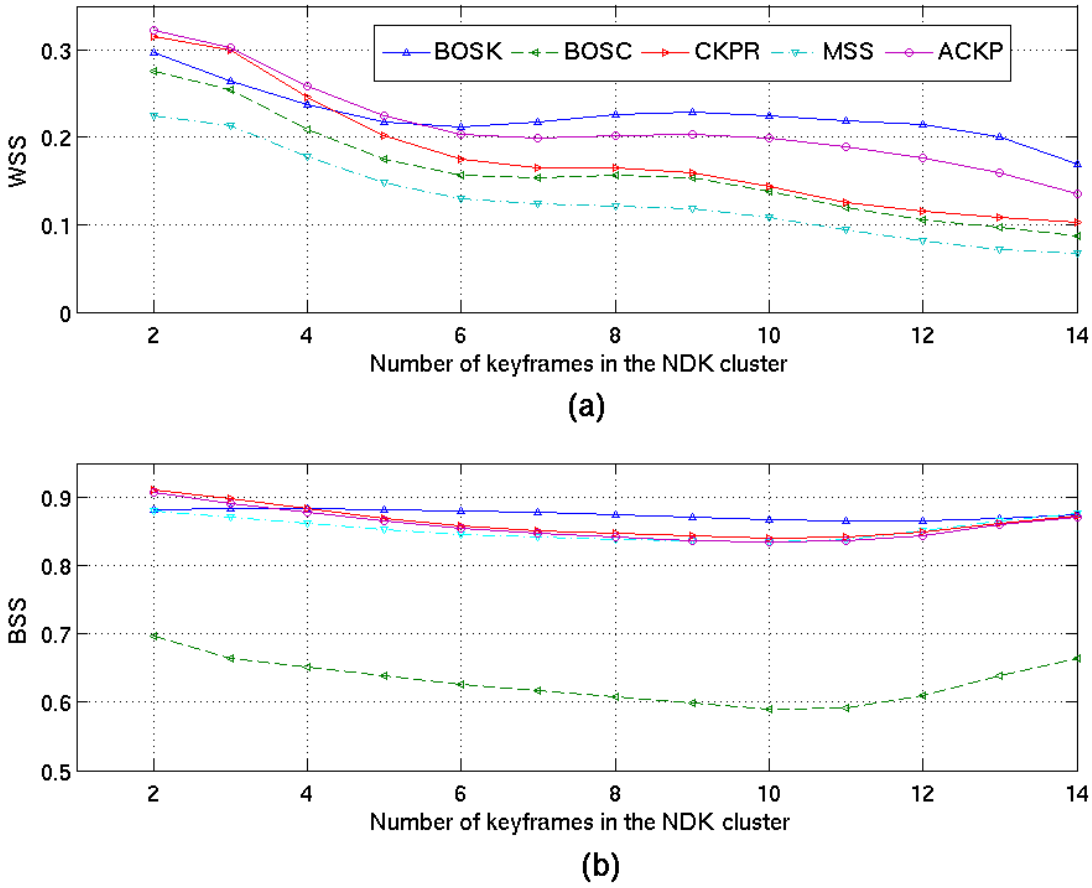

Figure 4.11: WSS and BSS values of different local signatures for NDK clusters with different number of keyframes — The best WSS belongs to the proposed MSS approach while the best BSS belongs to the BOSK approach.

mation by about 17%-40% in terms of number of keypoints. This compactness in indexing of visual cues accelerates news story retrieval process dramatically, roughly by 50 times, since we deal with a quadratic comparison of two stories' signatures.

4.6.4 Keyframe Cluster Retrieval Using Scene Signature

We now evaluate the effectiveness of the proposed scene signature for cluster retrieval. This task can be addressed as a prior task for duplicate, near-duplicate,

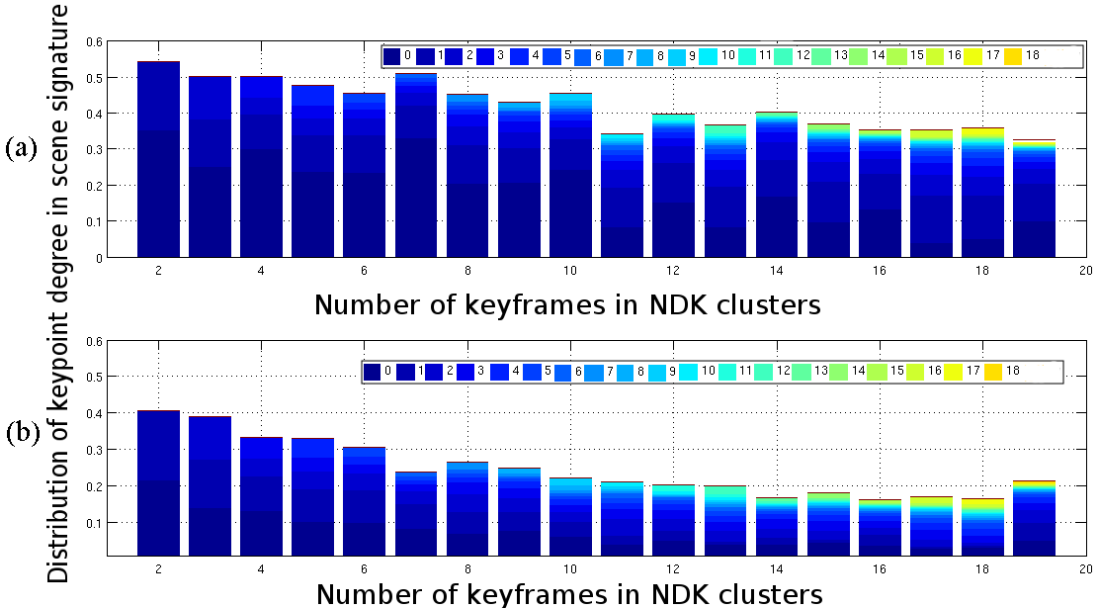

Figure 4.12: Distribution of keypoint degrees in scene signatures for NDK clusters with different number of keyframes — We normalize the distribution with the number of keypoints.

partially near-duplicate video detection especially for similar video pairs with different length, temporal order, novel visual contents and significant variations in the mutual visual content e.g., different setting, lighting condition, camera viewpoint.

As mentioned earlier, we extract keyframes from each story with the sampling rate of 1 fps and determine a time stamp for each keyframe denoting the second at which it is recorded with respect to the beginning of the story as the origin. Next, we group the extracted keyframes into two sub-stories of odd and even keyframes based on their determined time stamps as shown in Figure 4.13(a) and (b). We group NDKs within each sub-story to form NDK clusters using the method explained in Section 4.2.2. For instance, there are three and two NDK clusters in Figure 4.13(a) and (b), respectively. Since keyframes are extracted

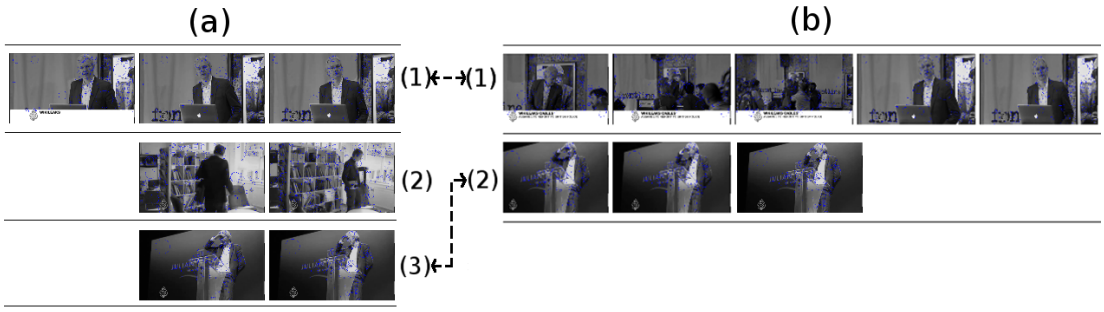

Figure 4.13: An example of two sub-stories — We group keyframes with odd time stamp and keyframes with even time stamp into two sub-stories shown in (a) and (b), respectively. NDKs are clustered into three and two groups in the sub-story (a) and (b), respectively. Dashed lines connect equivalent NDK clusters between sub-stories.

at 1 fps, for every shot longer than four seconds we have more than 4 extracted keyframes. Accordingly, each sub-story is expected to have at least two of those keyframes forming equivalent NDK clusters from an identical shot like cluster 1 in Figure 4.13(a) and cluster 1 in Figure 4.13(b). Consequently, the extracted video signature representing the equivalent NDK clusters should be similar and discriminative. Note that they will not be the same due to possible slight to intense variations across successive keyframes.

We manually label 2,070 NDK clusters as equivalent. We consider each of them as a query and measure the similarity between the query and all other clusters using different scene-level local signatures indicated in Table 4.1 and then rank them, accordingly. In Figure 4.14, we show the top-k NDK cluster retrieval results. The retrieval performance is quantified by the probability of retrieving the corresponding NDK cluster in the top-k position of the ranked list given as $P(k) = Z_c/Z$, where Z_c is the number of queries that rank their corresponding NDK cluster within the top-k position and Z is the total number of queries, which

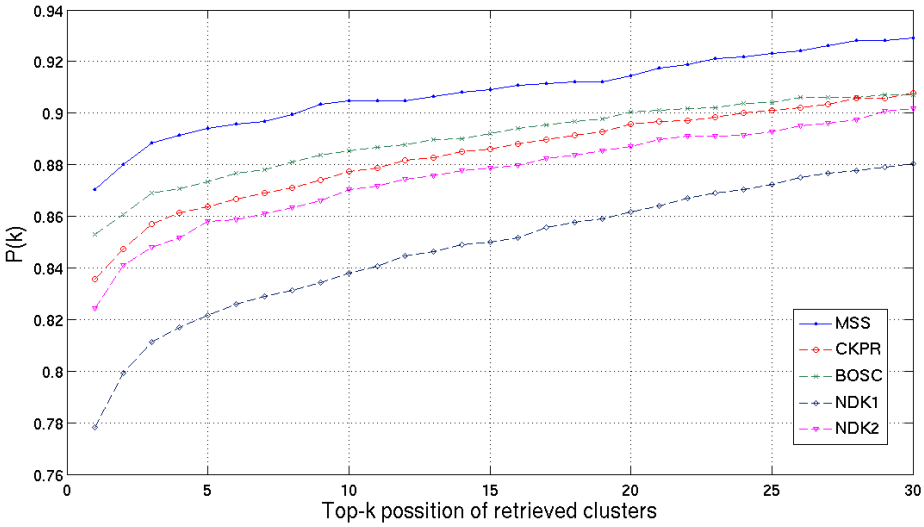

Figure 4.14: NDK cluster retrieval using different local signatures.

is 2,070. The best result belongs to our proposed MSS approach followed by BOSC and CKPR methods. The superiority of MSS confirms the effectiveness of our connected and isolated keypoints analyses and usefulness of refinement steps. Using BOSC, we obtain a slightly better performance than CKPR which shows the usefulness of isolated keypoints to form more discriminative video signature.

Figure 4.14 also illustrates the NDK cluster retrieval performance based on the keyframe-level similarity between their keyframes. NDK1 and NDK2 indicate the retrieval results where the least and the most similar keyframes between two NDK clusters are considered, respectively. We use BOSK method, as described in Table 4.1, to determine the similarity between every keyframe in the query cluster and every keyframe in all other clusters. As mentioned in [132], keyframe-level signatures and similarity methods are highly affected by the keyframe extraction/selection, since even keyframes extracted from an identical shot might be

represented by fairly distant features due to significant variations. The retrieval results show that even if we do not consider the effect of keyframe selection, in the best case (i.e. NDK2 where we take the most similar keyframes between clusters to measure their similarity) we will not obtain the retrieval performance to be as good as the scene signature approach. Obviously, we obtain the worst result when we use the least similar keyframes between clusters to measure their similarity through NDK1. We can infer that the performance of the keyframe-level approach varies between NDK1 and NDK2 if we use an arbitrary keyframe pair of NDK clusters instead of using the least and the most similar keyframe pair. It should be mentioned that in addition to the better retrieval performance obtained using MSS approach compared to NDK1 and NDK2, the former is much faster (up to 50 times) due to the compact structure of the generated scene signature as shown in Figure 4.12.

4.6.5 Associated News Story Detection Evaluation

In this section, we evaluate the proposed scene signature representation for associated news story detection. We use news videos from one month in the TRECVID 2006 dataset including 830 news stories out of which there are 132 similar news videos belonging to the first and the second categories of associated news stories as in Figure 1.1(a) and (b). We consider each associated news story as the query and measure the similarity between the query and all reference stories using different local representations and similarity measures and then rank the reference videos, accordingly. The retrieval performance is quantified by the probability of retrieving the associated news stories in the top-k position of the ranked list given as $P(k) = Z_c/Z$, where Z_c is the number of queries that rank their associated

news stories within the top-k position and Z is the total number of queries, which is 132.

We compare our method with the classic bag-of-words (BOW) approach, which is a story-level local-feature-based approach, and with the Must-link approach [112], which is a keyframe-level local feature based approach. In the former, we build a visual word dictionary with a vocabulary size of 20,000 and present a news story as a tfidf vector of visual words and determine between-story similarity using cosine similarity. For the Must-link and the proposed scene signature approach, after finding similar scene/keyframes, we determine between-story similarity as

$$\text{Sim}(S_i, S_j) = |S_i \cap S_j| \times (1/|S_i| + 1/|S_j|), \tag{4.7}$$

where S_i refers to the set of scene/keyframe signatures contained in the i-th story. $|S_i \cap S_j|$ indicates the number of near-duplicate keyframes/scenes between S_i and S_j.

Table 4.3: Equal Error Rate (EER) comparison of methods for associated news story detection.

Method	BOW	Must-link [112]	scene signature
EER (%)	61.0	85.3	**87.4**

Table 4.3 shows the EER for these three approaches. The BOW approach uses the bag-of-SIFT structure to represent the entire news story. This approach works reasonably well when we deal with almost duplicate news stories i.e., the first category of associated news stories like Figure 1.1(a). These are stories sharing the same original footage and they might also include additional contents. However, for near-duplicate news stories which are not from the same footage (i.e. some

news stories from the second category, explained in Section 1.3), the performance is degraded since the final BOW representation might include the significantly noisy information from the event of interest. On the other hand, the Must-link approach performs well on both the first and the second categories of the news stories. However applying the brute-force m-by-n comparisons across two news stories with m and n keyframes makes the Must-link approach computationally expensive.

In the proposed scene signature approach, the detection result is improved by around 2% compared to the Must-link approach. This improvement can be explained by post-processing steps to filter out less informative keypoints and integrating the connected keypoints within keyframe clusters through the scene signature generation. Moreover, since the number of scene signatures for a news story in the dataset is around one seventh of the number of keyframes and we conduct pairwise comparisons across news stories, our approach can process roughly 49 times (i.e. 7×7) faster than the Must-link approach.

4.6.6 Storyboard Generation using Scene Signature

A storyboard is a sequence of still frames depicting a moving sequence. It can be used for video summarization by listing its distinct visual contents [82]. As shown and explained in Figures 4.6 and 4.8, we could obtain more compact and meaningful clustering of keyframes through scene signature generation process which can be used for storyboard generation. Figure 4.6(d) shows a compact representation of the story as the storyboard in which each cluster is represented by a keyframe that is the most similar keyframe to the corresponding scene signature. We calculate the similarity between a scene signature and keyframes as explained

in Section 4.4. For example, the representative of the first cluster in Figure 4.6(c) is a medium shot, shown by the last keyframe in the first cluster, that contains most of the information within the cluster including the long shot as well as the close-up. Another example of a compact and semantic representation of a news story is shown in Figure 4.8(d). The numbers along the storyboard keyframes indicate the number of keyframes in the corresponding cluster.

Note that if we select the most connected keyframe in terms of near duplicate connection in each NDK cluster as its representative in the storyboard, then for example, we should choose the first keyframe for the fourth cluster in the Figure 4.8(d). The selected keyframe using the corresponding scene signature is the second keyframe which includes more details about the scene even though it is not the most connected keyframe in the NDK cluster. This result shows the effectiveness of scene signature to capture and integrate the semantics of a scene compared to the keyframe-level representations.

A further compact representation of a news story can be in the form of a thumbnail which is the corresponding keyframe of the scene signature with the largest weight. The weight refers to the number of keyframes in the corresponding cluster. Such thumbnails are marked in Figure 4.6(d) and Figure 4.8(d) by the red rectangle.

4.7 Conclusion

We proposed a novel video signature, called scene signature, which can be applied for various tasks in the unconstrained news video domain. First, we detected NDK clusters within a news story. For each cluster, we generated an

initial scene signature including most informative common and distinct visual descriptions. Next, through three steps of refinement of initial scene signatures, we shortened the semantic gap to obtain the final scene signatures which are more compact and semantic and are not sensitive to potential shot boundary detection errors or keyframe selection strategies. Using the SIFT local feature as the descriptor, scene signatures become robust against routine variations in lighting, object/camera displacement, or display layout. Moreover, since a scene signature is based on the visual clues presented in the context of a video scene and it does not consider temporal information, it is also robust against changes in temporal order, the length of story/shots, and additional/redundant visual information which often occur in associated news stories. The experiment results show the efficiency as well as robustness and uniqueness of the proposed scene signature compared to other global and local video signatures.

Chapter 5

Enhanced Textual Content Similarity

Broadcast news videos contain enriched auditory, textual and visual cues that can be used for news story retrieval. For example, in a *reader*, which is a type of news article read without accompanying video, the words, spoken by an anchorperson, contain the inherent semantic information. In order to extract the semantics, we need to resort to ASR techniques to retrieve spoken words. In this chapter, we aim to exploit textual information in news videos. The two sources of textual information are the OCR transcript and the output of the ASR engine.

Unlike documents, overlaid texts in news videos possess a wide range of sizes, fonts, colors, and mostly complex, dynamic, and/or transparent backgrounds. These complicating factors cause the OCR output to be highly erroneous. For instance, the word accuracy for detected text was estimated to be only 27% for VOCR in [54]. Hence, our first objective is to correct OCR errors. Next, we use textual semantic similarity to measure the relatedness between all the extracted text, i.e. from OCR transcripts as well as from the ASR engine. This similarity will be used together with visual similarity for the final associated news story

retrieval. However, in this chapter, we illustrate its effectiveness by an experiment on text-based video retrieval.

5.1 OCR Refinement using Local Dictionary

We carry out a simple but fast pre-processing procedure followed by a novel OCR post-processing method using the concept of a local dictionary to effectively correct erroneous OCR output and also to tackle the large number of OCR outputs. Then we use the modified OCR data together with the ASR transcript to generate keywords that are specific to a particular story. To evaluate our system and to understand how much textual information alone is effective, we utilize this enriched textual feature for news story retrieval purpose.

5.1.1 Optical Character Recognition

In news videos, overlaid text is mostly located at the bottom of the screen. We create a profile for each of seven channels in the dataset that specifies the spatial information of overlaid components in the screen. This area is highlighted as a box in the original keyframe shown in Figure 5.1(a). We spatially extract the gray-scale text box using the prior knowledge of the position of overlaid text box in the broadcast channel as in Figure 5.1(a)(i). The gray-scale text box is binarized using Otsu's method [87] as shown in Figure 5.1(a)(ii), and is input to the OCR engine [8].

It should be mentioned that for the OCR binarization, scholars have studied more complex and accurate methods such as Multiresolution Otsu [109] and Markov model for OCR binarization [48]. Multiresolution Otsu is a local version of the Otsu's method which takes blocks of pixels at different resolution levels

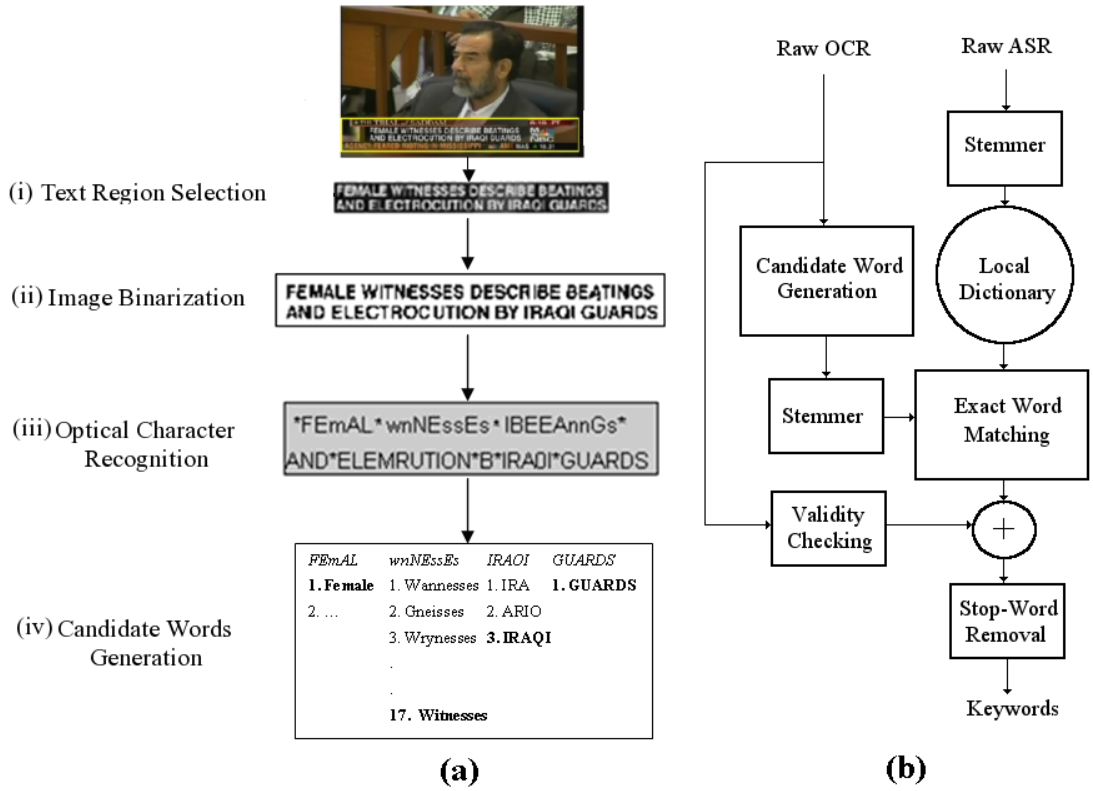

Figure 5.1: Overview of the OCR refinement — (a) pre- and (b) post-processing.

when determining the threshold. Markov model considers structure of each letter to be independent of its surroundings and calculates the probability that each pixel is text, given its neighbor pixels. In this research, most of the news stories do not have a complex text region background and the relatively low accuracy of OCR is mainly due to poor quality of keyframes. In addition, we mainly focus on the post-processing OCR refinement. Therefore, we employ the Otsu's method for image binarization.

The output of the OCR is a group of highly erroneous terms for each keyframe. In Figure 5.1(a)(iii), terms are separated by asterisks (*).

5.1.2 OCR Output Refinement

We use the spell-checker engine, called ASPELL [7], which generates a group of candidate words for each incorrectly spelled OCR output. The candidate words are ranked in ascending order of their similarity to the raw OCR output. This similarity score is determined by considering typo analysis. In Figure 5.1(a)(iv), the three words "FEmAL", "wnNEssES", and "IRAOI" are input to ASPELL and it generates 28, 28, and 8 candidates, respectively. In addition, the correctness of some terms (i.e. GUARDS) can be verified by the spell-checker. Eventually, there is a group of incorrectly spelled terms in OCR output which can not be refined by the spell checking procedure. For instance, there is no "ELECTROCUTION" among candidate words generated for the term "ELEMRUTION" in the above example.

In order to pick the right word among the generated candidates, we utilize the fact that both spoken words and overlaid text address an identical story content and most likely share some common important words. Hence, we build a local dictionary for each story using ASR transcript as shown in Figure 5.1(b). The local dictionary is basically the raw ASR transcript with the words converted to their roots by a stemmer. The local dictionary is looked up for the words that are output by the spell checker and converted to the root form by a stemmer, and a set of words common to the local dictionary and the stemmer output is created. Note that if we use the global dictionary, each OCR term should be converted to the closest word in the global dictionary. In Figure 5.1(a), "Wannesses" and "IRA" are closest words to "wnNEssEs" and "IRAOI", respectively, according to the global dictionary. This is not correct. Those words that have been correctly

recognized by OCR, e.g., GUARD in the above example, and may not necessarily exist in the ASR transcript, are also detected by validity checking process in Figure 5.1(b). The compilation of these valid words and common words are passed through stop word removal filter and the result is a set of keywords.

5.1.3 Term Weighting Scheme Using Keywords and ASR Transcript

We introduce an effective term weighting scheme for the extracted keywords and ASR transcript. A common method for representing text feature is the $tfidf$ (term frequency-inverse document frequency) feature,

$$\text{tfidf}(i,j) = \text{tf}(i,j)/\text{df}(i), \tag{5.1}$$

where $\text{tf}(i,j)$, called the term frequency, is the number of times term_i appears in the j-th document and $\text{df}(i)$, called the document frequency, is obtained by dividing the number of documents containing the term_i by the number of all documents. To prevent the high dimensionality of the tfidf feature in text document analysis, the infrequent words, for instance those occurring only once, are removed.

A high tfidf score is achieved by a high term frequency (in the given document) and/or a low document frequency of the term in the entire collection of documents; hence, the score tends to filter out common terms. In other words, terms occurring frequently in the entire collection of documents receive less score. This is not always desirable. For example, although the term "war" is a common term and is widely used in news reports, it is significant for the event "Iraq war" if the retrieval task needs to take the event "war" into account. Therefore,

instead of df, we utilize a Local Document Frequency (LDF) within a temporal window in each story. Specifically, the LDF score is determined for each term based on temporal proximity of news stories, i.e., the denominator of LDF will be the number of documents including term_i within, say, a day. In order to construct the enhanced tfidf measure, we combine the extracted keywords from OCR and from the ASR transcripts as

$$\text{wtfidf}(S_j, \text{term}_i) = \frac{\text{tf}_{A_j}(\text{term}_i) + \text{tf}_{O_j}(\text{term}_i)}{\text{LDF}(\text{term}_i)}, \tag{5.2}$$

where tf_{A_j} is the term frequency in the ASR transcript of story S_j and tf_{O_j} is the term frequency of keywords of story S_j. Since the extracted keywords only contain those OCR outputs whose rectified outputs are contained in the ASR transcript (other than the already correct OCR outputs), the post-processing method enables the suppression of a large number of irrelevant OCR outputs while also reducing the number of wrongly rectified stop-words. These keywords act as a natural tag for a news story which could be used for indexing.

5.2 Semantic Relatedness in Textual Domain

Here, we propose a measure for semantic similarity between news stories based on textual modality. The textual information from OCR transcript and ASR has been represented using weighted tfidf according to Equation (5.2). This representation enables the computation of similarity between two news stories by simply calculating the cosine of the angle between their corresponding wtfidf vectors. However, there is a significant number of incorrectly recognized words in the extracted ASR/OCR transcripts. Also it often happens that different news agencies use different but related words to report an identical event. Particularly, for non-

Table 5.1: Extracted ASR transcripts for "Fassir war tour" news published by NBC and MSNBC channels.

NBC channel on Fassir War Tour	MSNBC channel on Fassir War Tour
NBC nightly news monday He is nicotine use travels have left people shaking their **heads** in disbelief this State Department says **ferris** is signed is on his way home after getting into **Baghdad** buy and sell to carry out a **high school** journalism project while he may be honest way back to **Florida** and now when everyone wants to know is how he ended up in the rocky in the first place here's NBC scary sanders A lot of fun get an **american** teenager on **vacation** in **Baghdad** **sixteen** year old **Ferris** is sons improbable **journey** began in one of **Fort Lauderdale** wealthiest neighborhoods and **odyssey** as **mother** says she did not sanction Some progress and by Diane Terrifying to hasn't said that she undertook There is a son's date off began in **december** when he secretly **flew** from **Florida** to **Kuwait** City hired a **cab** to **drive** them to Baghdad but was turned back at the **border** flew to Beirut where a family friend got him a **visa** the day after **christmas** he landed in the **iraqi** capital...	**Sixteen** year old **Paris** the sun is alive safe and in deep trouble with his **mom** He has no idea what's he had with the suffer too The **Fort Lauderdale** **teenager** had threatened for months that he was going to **Baghdad** his mom says she had no reason to believe that he actually try to have a leader I have a nineteen appetite and see deadline said that's it yes they were sunday at **tickets** and new and he would be able to **travel** along it is about twelve it's to me and I hate to ask about the leak **American** born **farris** who is of **iraqi** heritage left South **Florida** before **christmas** his **odyssey** to come first to **Kuwait** we're trying to get a close the **border** a **taxi** he then **floated** the route where it's believed he was able to get a **flight** into **Baghdad** This is the thing that I astonish me how could see a sixteen year old gets a **visa** to go to **backup** ...

English news, since we use machine translation to generate the English version of the extracted ASR transcript, using semantic similarity can play a critical role to measure the textual similarity between stories more effectively. The extracted ASR transcripts for the "Fassir war tour", reported by NBC and MSNBC channels, are shown in Table 5.1. We highlight some incorrectly recognized words in green such as "heads","Ferris","backup" which were supposed to be recognized as "hands","Fassir" and "Baghdad". We also show some identical words in bold. In addition, we also highlight some words in yellow that are semantically related in the two stories, such as "taxi" and "cab", "mom" and "mother", "Baghdad" and "Iraqi", "ticket" and "flew" etc. Incorporating such semantic similarities, we can improve the measure for textual similarity between these two stories.

There are two main approaches to measure semantic similarity between terms.

The first method is based on the terms relative positions in the WordNet hierarchy [80]. The second approach is the distributional method which is based on the co-occurrence of terms in a large-scale dataset like Wikipedia [4]. Here we briefly explain them.

5.2.1 WordNet Similarity

WordNet is a large lexical database for the English language [80]. It groups different types of words such as nouns, verbs, adjectives and adverbs into sets of cognitive synonyms called synsets, each of which expresses a distinct concept. WordNet includes 117,000 synsets. Relations between synsets are built based on lexical relations and semantics between different concepts. Due to this meaningful structure, WordNet has found many applications in computational linguistics and natural language processing [93].

Synonymy is the main relation between words in the WordNet through which related words, indicating the same concept, are grouped together into a synset (e.g. "shut" and "close" or "car" and "automobile"). There are some words with several distinct senses e.g., "bass" can be a type of fish or tones of low frequency, which are represented in distinct synsets in WordNet. In Table 5.2, we show other noun relations in the WordNet. In the verb hierarchy, verbs towards the bottom of the tree express more specific manners describing an event, as in {parent:"communicate", child:"talk-whisper"}. The explored specific manner relies on the semantic field; volume (as in the example above) is just one aspect along which verbs can be expressed. Another example of specific fields is speed {move-jog-run} or intensity of emotion {like-love-idolize}. In Table 5.3, we show different verb relations in WordNet.

Table 5.2: WordNet noun relations [93].

Relation	Also called	Definition	Example
Hypernym	Superordinate	From concepts to superordinates	$breakfast^1 \rightarrow meal^1$
Hyponym	Subordinate	From concepts to subtypes	$meal^1 \rightarrow lunch^1$
Member Meronym	Has-Member	From groups to their members	$faculty^2 \rightarrow professor^1$
Has-Instance		From concepts to instances of the concept	$composer^1 \rightarrow Bach^1$
Instance		From instances to their concepts	$Austen^1 \rightarrow author^1$
Member Holonym	Member-Of	From members to their groups	$copilot^1 \rightarrow crew^1$
Part Meronym	Has-Part	From wholes to parts	$table^2 \rightarrow leg^3$
Part Holonym	Part-Of	From parts to wholes	$course^7 \rightarrow meal^1$
Antonym		Opposites	$leader^1 \rightarrow follower^1$

Table 5.3: WordNet verb relations [93].

Relation	Definition	Example
Hypernym	From events to superordinate events	$fly^9 \rightarrow travel^5$
Troponym	From a verb (event) to a specific manner elaboration of that verb	$walk^1 \rightarrow stroll^1$
Entails	From verbs (events) to the verbs (events) they entail	$snore^1 \rightarrow sleep^1$
Antonym	Opposites	$increase^1 \Longleftrightarrow decrease^1$

The most detected relation among synsets is the super-subordinate relation which links more general synsets like “furniture”, “piece-of-furniture” to increasingly specific ones like “bed” and “bunkbed”. Therefore, WordNet indicates that the category “furniture” includes “bed”, which in turn includes “bunkbed”. Conversely, concepts like “bed” and “bunkbed” form the “furniture” category. All noun hierarchies ultimately end up with the root node called “entity”. WordNet also discriminates Types (common nouns) and Instances (specific persons, countries and geographic entities). For example, “wheelchair” is a type of “chair”, “Singapore” is an instance of a “country”. Instances are always leaf (terminal/end) nodes in the hierarchies.

WordNet-based Similarity Two words are similar if they are nearby in the WordNet hierarchy. Figure 5.2 shows a small fragment of WordNet hierarchy where related words are connected based on relationships described earlier.

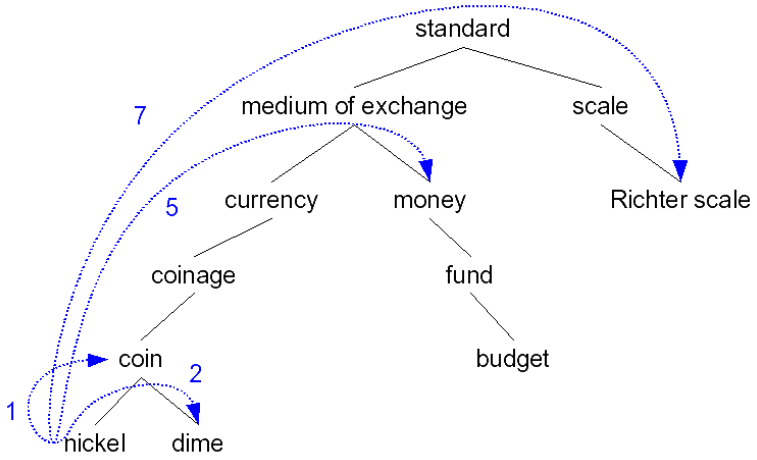

Figure 5.2: Path-based Similarity — short paths between some words in the WordNet Hierarchy [93].

"Dime" and "nickel" are directly connected to the "coin" since they are the instances of "coin". "Fund" and "budget" are directly connected since they are synonyms.

There are three main categories of the WordNet-based similarity methods: (1) edge counting measured based on the path length between terms [68]. For instance, the path length between "money" and "nickel" is five in Figure 5.2. (2) information content measured as a function of the probability of occurrence of the terms in the corpus [93]. The more general a word is, the lower information content it has. (3) combination of the above approaches where the path-based similarity between two words is weighted with respect to their information content [99]. The problem with the basic path-based similarity is that it assumes each link represents a constant distance. For instance, "nickel" to "money" seems closer than "nickel" to "standard" as shown in Figure 5.2. In this research, we utilize WordNet similarity in [37] where authors introduce a metric called WNSim, to measure similarity between a word and a multi-word expression based on the

WordNet hierarchy. The problem with the general WordNet-based approaches is that many words e.g., "Obama", "IBM" etc., and also technical terms are missing from the hierarchy. Moreover, it relies on hyponym information which is good for nouns but not so for adjectives and even verbs. In addition, the notion of similarity differs across domains. These facts motivate us to incorporate another semantic similarity called distributional similarity using Wikipedia collection.

5.2.2 Distributional Similarity

The main idea behind distributional similarity is that similar terms appear in similar contexts [37, 69]. The first step is to represent each word as a vector, components of which are labeled with other words (context words). Value of each component is determined based on the co-occurrence of the target words with component label. Note that context can be defined based on syntactic roles, location or distribution within collection of documents. For instance, if w='tesguino', v_1='bottle', v_2='drunk', v_3='matrix' then $\mathbf{w} = (1, 1, 0)$. We can calculate semantic similarity between two words simply as the cosine similarity between their representative vectors.

We utilize semantic similarity proposed in [65] which is based on the the statistical analysis of Wikipedia collection. They experimentally investigate how different factors like context, corpus preprocessing and size, and dimension reduction techniques can affect the semantic properties of the resulting word spaces. In their proposed method called DISCO (extracting DIStributionally similar words using COoccurrences), they use a simple context window of size three words for counting co-occurrences. They show that it is beneficial to take the exact position within the window into account. Consequently, the feature that describes a

word distribution is not just a bag of words, but ordered pairs of word and the window position. For example, considering the sentence "the nuts provide palm oil", palm is presented using triples: <palm, −2, nuts>, <palm, −1, provide> and <palm, +1, oil> where the numbers indicate their position distance.

5.2.3 Semantic Similarity Refinement

We aim to use the above-mentioned semantic similarity metrics to determine the textual similarity between the wtfidf representations of the news stories calculated in Equation (5.2). The final semantic similarity score is given as

$$\text{Semantic\_Similairty}(t_1, t_2) = \begin{cases} \text{WNSim}(t_1, t_2) + \text{WikiSim}(t_1, t_2) & \text{if } (\text{WNSim}(t_1, t_2) > \tau_w), \\ 0 & \text{otherwise} \end{cases} \tag{5.3}$$

where $\text{WNSim}(t_1, t_2)$ and $\text{WikiSim}(t_1, t_2)$ denote the WordNet-based and Wikipedia-based semantic similarities between t_1 and t_2, respectively. According to preliminary experiments, we found that WordNet-based similarity is more robust and reliable than Wikipedia-based similarity in our case. Hence, we filter out unreliable relatedness if WNSim between two words is lower than a particular value ($\tau_w = 0.3$) as shown in Equation (5.3).

Considering all words that are semantically related according to Equation (5.3), we observe many noisy connections between terms, which degrades the retrieval performance. In Figure 5.3, we show the number of words in the dataset which are connected to each extracted ASR/OCR word using WordNet similarity. There are 8,827 different words in the ASR/OCR transcripts among which general terms such as "go", "make", "take" can get connected to over 100 words. This is clearly very noisy and is therefore undesirable. On the other hand, specific words such

as "rain","damper", "bomber" have much fewer numbers of related words in the dataset.

To suppress the problem with terms having noisy connections, we adopt a soft-mapping scheme to only consider the top-k semantic similarities for each word in the dataset. We calculate k through a 10-fold cross-validation process. We split the development data into 10 folds and use 9 out of 10 folds for the training where we employ a grid search ($k = 1, 2, .., 15$) to find the best k which results in the best retrieval performance in the remaining fold of development data. Next, we calculate the proximity matrix, $\mathbf{SS} = [s_{ij}]_{n \times n}$, where s_{ij} refers to the semantic similarity between the i-th and the j-th words in the dataset as determined in Equation (5.3). Considering the determined k for soft-mapping, the i-th column of $\mathbf{SS}$ has non-zero values for only the top-k most semantically similar words in the dataset to the i-th word. Note that n denotes the number of words in the dictionary which is the compilation of all extracted words in the dataset passed through the stop word removal and the stemming process. We can calculate the semantic similarity between stories as

$$\text{Score}_t(S_1, S_2) = \text{wtfidf}^{\top}(S_1) \times \mathbf{SS} \times \text{wtfidf}(S_2), \tag{5.4}$$

where wtfidf(S_1) and wtfidf(S_2) refer to the weighted tfidf features for S_1 and S_2 as calculated in Equation (5.2). Note that if we set $\mathbf{SS}$ to $\mathbf{I_{n \times n}}$, Equation (5.4) reduces to the conventional cosine similarity between wtfidf features. In the experiment section, we evaluate the effectiveness of our textual semantic similarity metric using wtfidf features against Vector Space Model (VSM) method as the baseline.

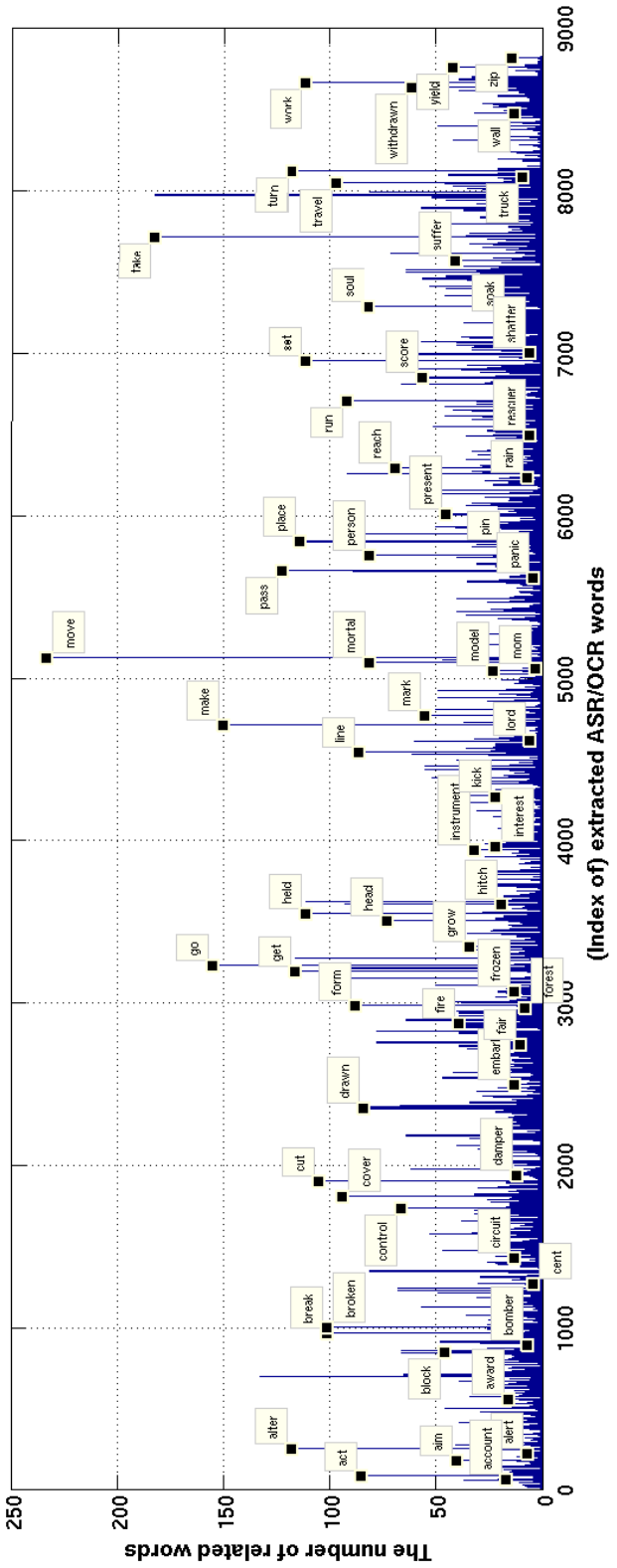

Figure 5.3: The original WordNet Similarity — the more general a word is, the more number of related words it has.

5.3 Experimental Results

5.3.1 Dataset

We evaluate different text-only approaches for associated news story retrieval. We use news videos from the TRECVID 2006 corpus from 7 different channels addressing world news stories in December 2005. We use ASR transcripts provided by [2]. We manually segment the news stories and label them based on their main topic.

The dataset contains 830 news stories out of which 296 pairs of associated news stories, from all three categories, are labeled. The ASR transcript of a news story contains between 0 to 1320 unique words (128 in average). The OCR transcript of a news story has 34 words in average. Note that due to erroneous OCR output, multiple versions/duplicates of an identical word can be often seen in the OCR transcript of a news story. Considering only unique words, there are 21 words in average in an OCR transcript. Considering all ASR and OCR transcripts, there are 8,827 unique words in the dataset.

5.3.2 Associated News Story Retrieval Evaluation

We consider each associated news story as the query and measure the similarity between the query and all reference stories using different text-only representations and similarity measures and then rank the reference videos, accordingly. The retrieval performance is quantified by the probability of retrieving the associated news stories in the top-k positions of the ranked list given as $P(k) = Z_c/Z$, where Z_c is the number of queries that rank their associated news stories within the top-k position and Z is the total number of queries, which is 296.

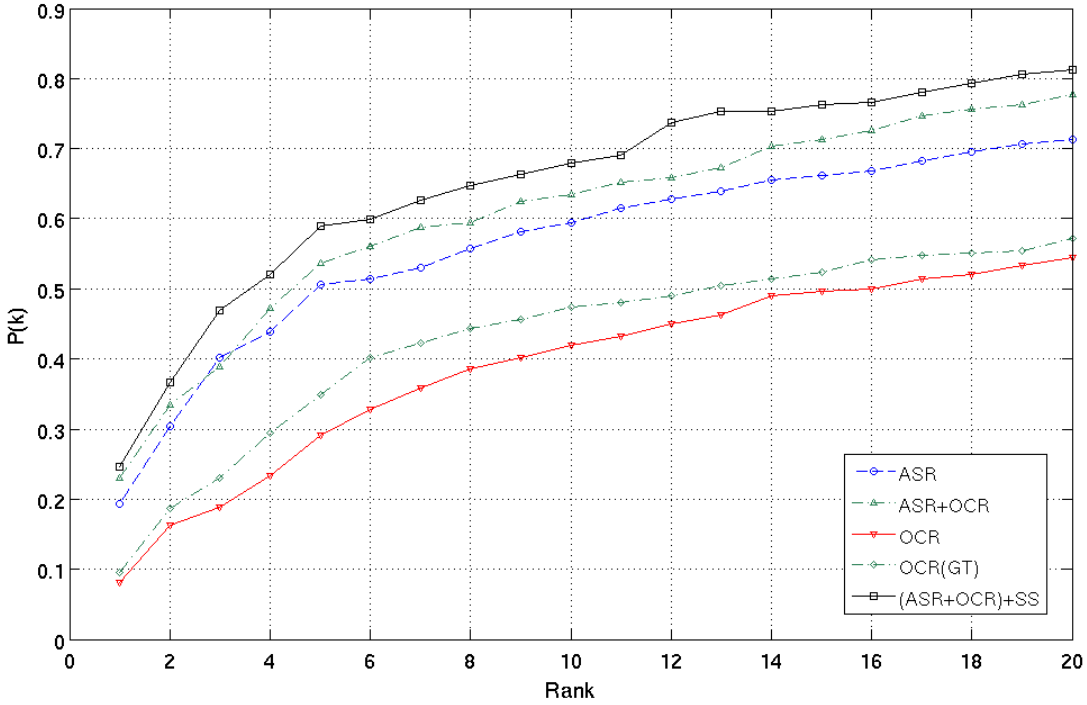

Figure 5.4: The top-k retrieval results using textual modalities — The best result is obtained by using the ASR, refined OCR output, and the proposed textual semantic similarity.

Figure 5.4 shows the top-k news story retrieval results using ASR (ASR), refined OCR (OCR), OCR ground truth (OCR(GT)), and enhanced textual representation (ASR+OCR). We also show the retrieval result of using the textual semantic similarity (ASR+OCR+SS) as explained in Section 5.2.3. The best retrieval result is obtained when we use both ASR and refined OCR transcript to generate the wtfidf feature and use our proposed semantic similarity metric of ASR+OCR+SS. Note that using only ASR transcript performs better than using only OCR transcript since only some news stories have informative captions while most of them have representative ASR transcripts extracted.

Table 5.4: Precision, Recall, and F-measure for different OCR post-processing methods

Methods	OCR	OCR + proposed method (refined OCR)	OCR + n-gram [18]	OCR+global [54] dictionary
Precision (%)	18.40	**42.25**	20.35	21.78
Recall (%)	20.67	**48.14**	24.44	22.54
F-measure	19.46	**44.99**	22.21	21.15

5.3.3 OCR Refinement Evaluation

Table 5.4 compares our OCR post-processing method, explained in Section 5.1.2, with others. We determine the local dictionary based on the assumption that story boundaries are given. Using the standard precision and recall scores, we evaluate the OCR transcripts quality before and after refinement using the ground truth of OCR which is manually provided. Precision is the fraction of the retrieved OCR words that are correctly recognized, while recall is the fraction of correctly recognized words that are retrieved. F-measure is the harmonic mean of the precision and recall scores. We could improve the OCR precision from 18% to 42% and the OCR recall from 21% to 48% using our proposed refinement steps, which are significantly better than the n-gram and the global dictionary methods. Moreover, as shown in Figure 5.4, the performance using the refined OCR outputs is close to the ground truth OCR, determined manually. This observation also confirms the effectiveness of our method to refine noisy OCR outputs and to determine more informative keywords among them.

5.4 Conclusion

In this chapter, we proposed a method to determine enhanced textual similarity where first we refined noisy OCR outputs through an early fusion process using ASR transcripts. Then we integrated ASR transcript and refined OCR outputs to generate a weighted tfidf textual representation of a news story. We also proposed a textual semantic similarity measure to determine the relatedness between textual features using the WordNet hierarchy and Wikipedia collection. Experimental results confirm the usefulness of our proposed OCR refinement method and the effectiveness of enhanced textual representation and similarity.

Chapter 6

Multi-modal Solutions for Associated News Story Retrieval

In this chapter, we investigate multi-modal approaches to retrieve associated news stories sharing the same main topic. In the visual domain, we employ near duplicate keyframe/scene detection method using local signatures to identify stories with common visual cues. Furthermore, to improve the discriminativness of visual representation, we develop a semantic signature that contains pre-defined semantic visual concepts in a news story. We propose a visual concept weighting scheme to combine local and semantic signature similarities to obtain enhanced visual content similarity.

To fuse textual and visual modalities, we investigate different early and late fusion approaches. In the proposed early fusion approach, we employ two methods to retrieve the visual semantics using textual information. Next, using a late fusion approach, we integrate uni-modal similarity scores and determine early fusion similarity score to boost the final retrieval performance. Experimental results show the usefulness of the enhanced visual content similarity and the early fusion approach, and the superiority of our late fusion approach.

The rest of this chapter is organized as follows. In Section 6.1, we propose an effective enhanced visual content similarity measure between stories using local and semantic signatures. In Section 6.2, we explore early fusion methods to retrieve visual semantic signatures using textual information. In Section 6.3, we combine the enhanced visual, textual and early fusion similarities through different fusion approaches to improve retrieval performance. In Section 6.4, we assess the enhanced visual content similarity measure and also different multi-modal approaches for associated news story retrieval.

6.1 Enhanced Visual Content Similarity

We determine enhanced visual content similarity between stories using local and semantic signature. Local signature refers to the keyframe- or scene-level local feature representation explained in Chapter 4. Here, we use Bag-of-SIFT to represent each keyframe within a news story and determine inter-story similarity as keyframe set similarity as

$$\text{KF\_Sim}(S_i, S_j) = |S_i \cap S_j| \times (1/|S_i| + 1/|S_j|), \tag{6.1}$$

where S_i refers to the set of keyframes contained in the i-th story. $|S_i \cap S_j|$ indicates the number of near-duplicate keyframes between S_i and S_j based on the number of matching keypoints explained in Section 3.1.3. Although local signature is robust to low to mid-level object displacement, edit and camera setting variations, it does not perform well when we deal with associated news stories with significant object/camera movements, or stories with conceptual connections like Figure 1.1(c) where the visual similarity between stories can be implicitly retrieved based on common visual concepts they share such as “fire”, “firefighter”,

"jungle", etc. This observation motivates us to incorporate a visual semantic representation for a news story, called *semantic signature*.

6.1.1 Semantic Signature

Semantic signature is essentially a 374-dimensional vector representing the probabilities of the presence of 374 predefined visual concepts [60] in a news story. These concepts are related to events (e.g. "election campaign", "parade"), objects (e.g. "U.S. flag","vehicle"), locations (e.g. "urban park", "outdoor"), people (e.g. "male person", "corporate-leader"), and activities (e.g. "swimming", "dancing"). They have been selected through a multi-step process involving input solicitation, expert critiquing, comparison with related ontologies, and performance assessment [6]. We determine semantic signature as

$$\mathbf{SemSig}(S_i) = \sum_{j=1}^{n} \mathbf{SIN}(\mathrm{KF}_j^i), \tag{6.2}$$

where n denotes the number of keyframes within story S_i and $\mathbf{SIN}(\mathrm{KF}_j^i)$ refers to the Semantic INdexing of the j-th keyframe in S_i which is a 374-dimensional vector indicating probabilities of the presence of the predefined visual concepts in the j-th keyframe. In [60], the authors have computed these probabilities by training an SVM classifier. The similarity between semantic signatures can be simply determined using cosine similarity measure. In practice, semantic signature is not discriminative enough due to limited number of visual concepts (i.e., 374) and relatively low precisions of visual concept detectors. However, we aim to incorporate it with local signature similarity to explore their possible complementary roles.

Figure 6.1: NDK detectability for various visual concepts — The figure shows three sample NDKs with (a) dynamic and (b) static visual concepts. The more static visual concepts a keyframe contains, the more likely the NDK can be detected.

6.1.2 Visual Concept Weighting

To combine local and semantic signature similarities, first we investigate when local signature works well and when it fails so that the semantic signature similarity can be weighted, accordingly. Local signature does not work well when keypoint matching scheme does not perform well. Generally speaking, keypoint matching methods suffer from significant object/camera movements occurring in scenes with concepts that are dynamic. For instance, Figure 6.1(a) shows three NDK pairs depicting a basketball game, dancing at a party, and an explosion, respectively. In all of these NDKs, keypoint matching scheme does not work properly due to the existence of dynamic visual concepts. On the other hand, keypoint matching scheme performs well in scenes with mostly static concepts. For instance, Figure 6.1(b) shows three NDKs depicting a speaker, a building and mountains, in all of which keypoint matching scheme works well.

This observation motivates us to study the relation between visual concepts represented in a scene and the ability of local signature to capture the visual similarity across scenes. We use TRECVID 2006 dataset [2] including around 160 hours of news video from 7 different channels. There are around 21,000

shots each of which contains several keyframes which are NDKs. We determine matching keypoints between NDKs within each shot using the method explained in Section 3.1.3. We categorize shots into two groups of detectable and non-detectable if the number of matching keypoints between the NDKs in a shot exceeds a specific threshold. For each visual concept in each group, we calculate mean (μ_i) and variance (σ_i) of its probability over all keyframes. Then for the i-th visual concept, we employ the *t-test* [94] and determine t-score(i) as

$$t\text{-score}(i) = \frac{\mu_i^{(1)} - \mu_i^{(2)}}{\sqrt{\frac{\sigma_i^{(1)}}{n_1} + \frac{\sigma_i^{(2)}}{n_2}}}, \tag{6.3}$$

where $\mu_i^{(1)}$ and $\sigma_i^{(1)}$, and $\mu_i^{(2)}$ and $\sigma_i^{(2)}$ denote mean and variance of the i-th visual concept probability in the detectable and non-detectable groups, respectively. A significant t-score(i) means that the presence of the i-th visual concept leads to a failure of NDK detection within the shots. In Figure 6.2, the t-score for all visual concepts are shown. Concepts like "sitting", "U.S. flag", "furniture", "windows" and "address or speech" have high t-score which implies that NDK detection algorithm is generally capable of finding scenes having these static concepts. On the other hand, concepts such as "shooting", "dancing", "ruins", "natural disaster" have low t-score which means that there is a difficulty to detect NDK containing these concepts. Accordingly, we compute Detectability Score (DS) for each concept as

$$\text{DS}(i) = 1 + \frac{1}{(1 + t\text{-score}(i))}, i = 1, 2, .., 374. \tag{6.4}$$

Hence, we assign higher DS on a concept with a lower t-score. We modify semantic signature similarity between stories by weighting visual concepts as

$$\text{Sim\_Sem}(S_i, S_j) = (\mathbf{DS} \circ \mathbf{SemSig}(S_i))^{\intercal}(\mathbf{DS} \circ \mathbf{SemSig}(S_j)), \tag{6.5}$$

where $\circ$ denotes the element-wise product between two vectors. Note that high t-scores means that we deal with relatively static visual concepts in the story and local signature similarity can probably capture visual similarity. Accordingly, we assign a low weight to semantic signature concepts that decreases Sim_Sem. On the other hand, a low t-score means that local signature similarity is not reliable to measure the visual similarity for the given news story. Thus, we assign more weights to semantic signature concepts in Equation (6.5) that increases Sim_Sem.

6.1.3 Fusion of Local and Semantic Signature Similarities

Finally, through a leave-one-out training process, we train an SVM classifier using local signature similarity and weighted semantic similarity as two inputs and use the trained model to determine enhanced visual content similarity score between a pair of story. In the proposed visual concept weighting, when local signature is more effective, semantic signature similarity score is decreased using DS in Equation (6.5) which in turn decreases its contribution in the trained SVM model. In contrast, when local signature similarity is likely to fail, we increase semantic signature similarity score which also increases its contribution in the trained SVM model.

Note that we can alternatively use scene signatures instead of bag-of-SIFT of keyframes as local signature and analogously determine their set similarity as explained in Equation (6.1) where $|S_i \cap S_j|$ denotes the number of similar scene signatures between the i-th and the j-th story as determined in Equation (4.4). In the experiment section, we also assess the final retrieval performance using this local signature.

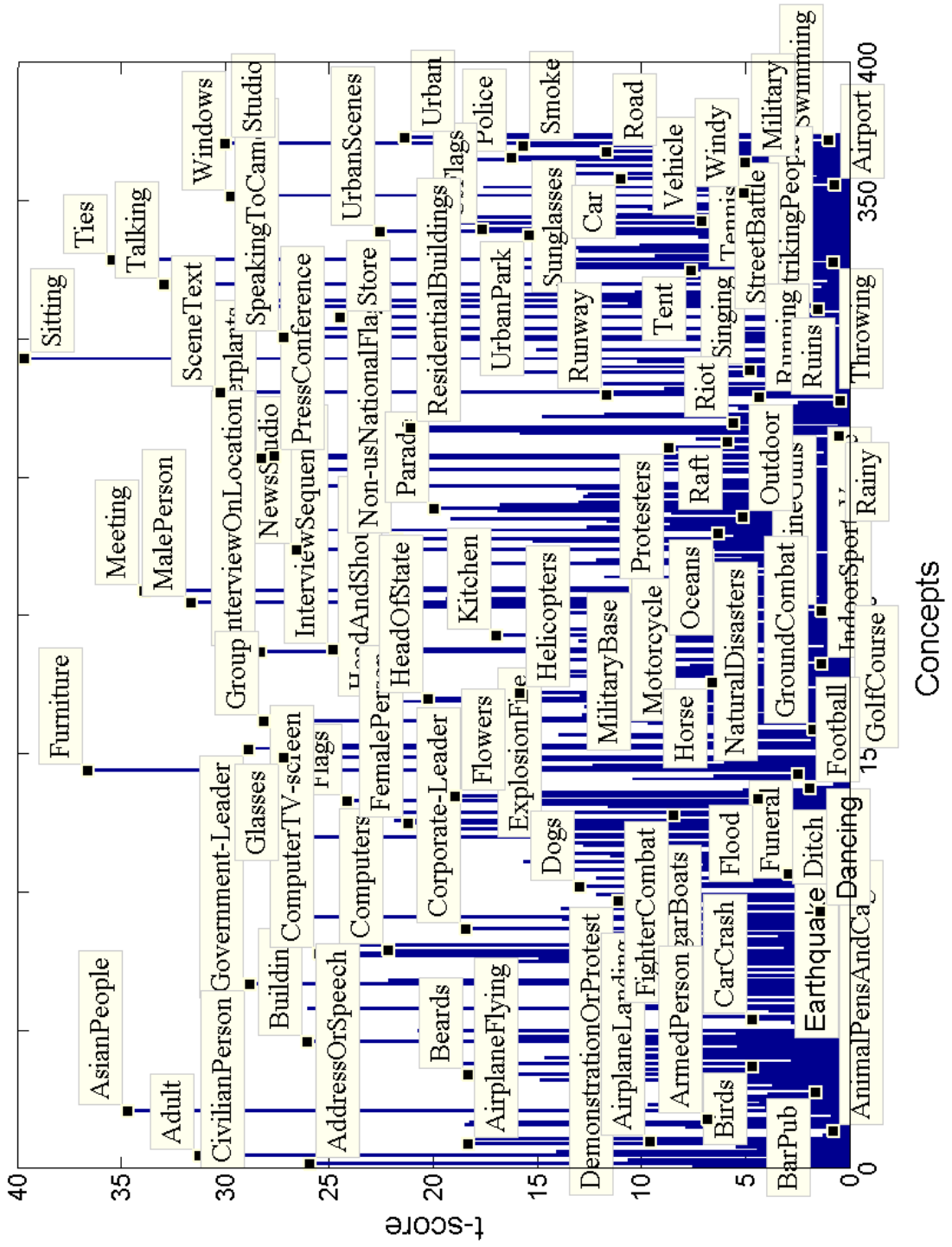

Figure 6.2: *t*-score for 374 visual concepts for TRECVID 2006 dataset [2].

6.2 Early Fusion of Visual and Textual Information

Although most of the news stories contains significant amount of textual information extracted via ASR and OCR, it often happens that some stories do not have representative visual cues which can be extracted through local or semantic signatures. For instance in *reader*, which is a type of news story read without accompanying video or sound, there is no relevant visual cue to the topic of interest. This fact leads to poor retrieval results using visual representation. To improve the retrieval performance for these stories, we can integrate both visual and textual modalities through early or late fusion approaches. In the early fusion, we combine these two modalities in the feature-level while in the late fusion we integrate similarity scores at the decision-level. In this section, we aim to propose an early fusion scheme where we retrieve the visual content using the textual content. In the visual domain, we represent each video using semantic signatures proposed in Equation (6.2), and in the textual domain, we describe a video using wtfidf representation explained in Equation (5.2). As explained in Section 5.1.3, we determine wtfidf representation for a story using ASR and refined OCR transcripts, weighted by local document frequency.

Since textual representation, wtfidf, and semantic signature, SemSig, come from different sources of knowledge and consequently different feature spaces, they are considered heterogeneous features and are not directly comparable. Hence, the proposed framework uses two scores to retrieve semantic signatures of reference stories for a given query story as shown in Figure 6.3. Reference stories refer to all stories in dataset other than the query story. First, we directly map

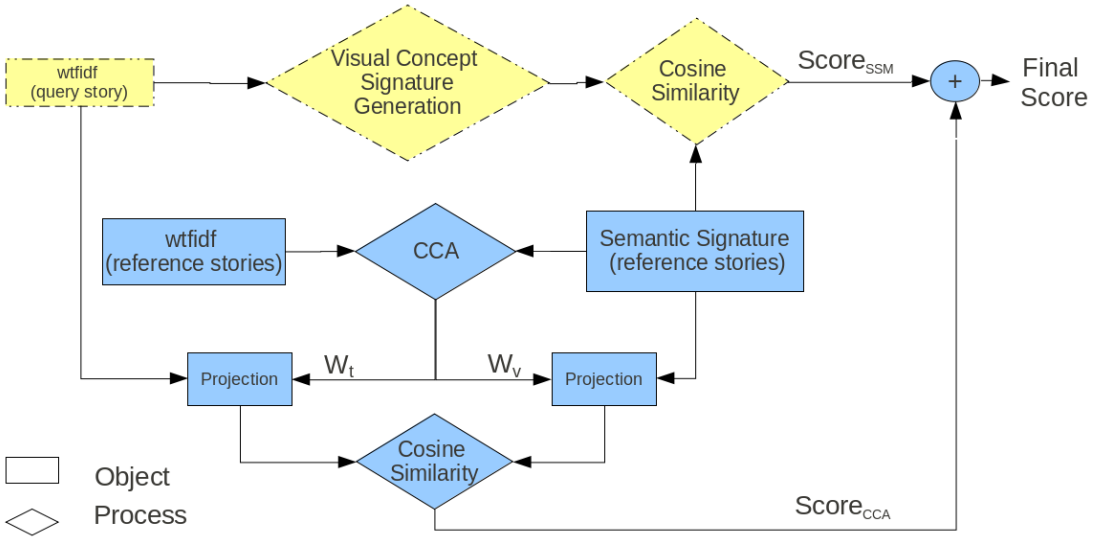

Figure 6.3: The proposed early fusion approach — Using CCA and the direct Semantic Similarity Mapping (SSM) of textual information on the visual concept list.

the wtfidf of the query story to the visual semantic feature space using Semantic Similarity Mapping (SSM) and generate visual concept signature as explained in the next section. Then we determine its cosine similarity to semantic signature of reference stories to obtain $\text{Score}_{\text{SSM}}$. The second score is obtained by mapping these heterogeneous features to a third feature space where they are comparable. We use Canonical Correlation Analysis (CCA) to learn the required projection functions as explained in Section 6.2.2. Using the projection functions, we map wtfidf of the query story and semantic signature of reference stories onto a third space where their cosine similarity is computed as $\text{Score}_{\text{CCA}}$. Finally, similarity between two stories is obtained as the sum of $\text{Score}_{\text{SSM}}$ and $\text{Score}_{\text{CCA}}$ as shown in Figure 6.3.

6.2.1 Semantic Similarity Mapping of Textual Information onto Visual Space

We intend to generate a visual concept signature using ASR and OCR transcripts for a query story and use it to retrieve visual semantic signature of reference stories, calculated in Section 6.1. It is especially useful when the query story has a noisy and an unreliable semantic signature but contains significant amount of textual information. For this purpose, first we need to map wtfidf of the query story to the list of 374 visual concepts. We use the semantic similarity, developed in Section 5.2.3, to determine textual semantic similarity between every term in the wtfidf dictionary and visual concepts. We build $\mathbf{SS} = [ss_{ij}]_{m \times n}$ where m and n refer to the number of visual concepts and the number of terms in wtfidf dictionary, respectively. $m = 374$ and $n = 8,827$. The i-th row of **SS** has non-zero values, determined by textual semantic similarity measure of Equation (5.3), for the top-k most semantically similar words in the dictionary to the i-th visual concept. Accordingly, we determine the visual concept signature for story S_i as

$$\mathbf{G\_SemSig}(S_i) = \mathbf{SS} \times \mathbf{wtfidf}(S_i), \tag{6.6}$$

where $\mathbf{wtfidf}(S_i)$ is the n-by-1 vector indicating the textual representation of S_i in Equation (5.2), n denotes the number of words in the dictionary, and $\mathbf{G\_SemSig}(S_i)$ is the m-dimensional vector representing the visual concept signature generated from textual information. Figure 6.4(a) shows an example story about "Fire in Oklahoma" broadcast by NBC channel. Figure 6.4(b) shows the ASR transcript of the story of interest. We bold some words in the extracted ASR transcripts that contribute to the semantic similarity mapping in the **SS** matrix calculation, e.g. "fire", "burn", "danger", "firefighter". Figure 6.4(c) shows the

visual concept signature (**G_SemSig**). We obtain relatively high scores for associated concepts like "Natural Disaster", "Firefighter", "Outdoor", "Smoke", etc. Moreover, there are some completely irrelevant words also with significant scores such as "Glass", "Lawyer", "Snow" etc.

To retrieve associated news stories for a given query story, we use cosine similarity to measure the similarity between the generated visual concept signature (i.e. **G_SemSig**) of the query story, as determined in Equation (6.6), and the visual semantic signatures of reference stories, calculated in Equation (6.2). We assess the retrieval performance in the experiment section.

6.2.2 Canonical Correlation Analysis

In this section, instead of directly mapping textual information onto visual feature space, we map both textual and visual semantic features to a third space where they are comparable. To determine the projection functions, we assume that the mapped textual and semantic signature of a news story are close together in the projection feature space. We use Canonical Correlation Analysis (CCA) to learn the co-occurrence of the textual information and visual concepts. In statistics, CCA is an approach to study cross-covariance matrices. If two sets of variables, $a_1, a_2, ..., a_n$ and $b_1, b_2, ..., b_m$, are correlated, then we can find a linear combination of the a_i and the b_i that has maximum correlation. In our case, we use textual feature ($\mathbf{T}$) and visual semantic indexing ($\mathbf{V}$) as a and b and consider their co-occurrence in a news story to determine the correlation between them. $\mathbf{T}$ is essentially an n-dimensional **wtfidf** feature vector determined for each story as explained in Section 5.1.3 where n is the number of words in the dictionary. $\mathbf{V}$ is a m-dimensional semantic signature determined for each video as in Equation (6.2)

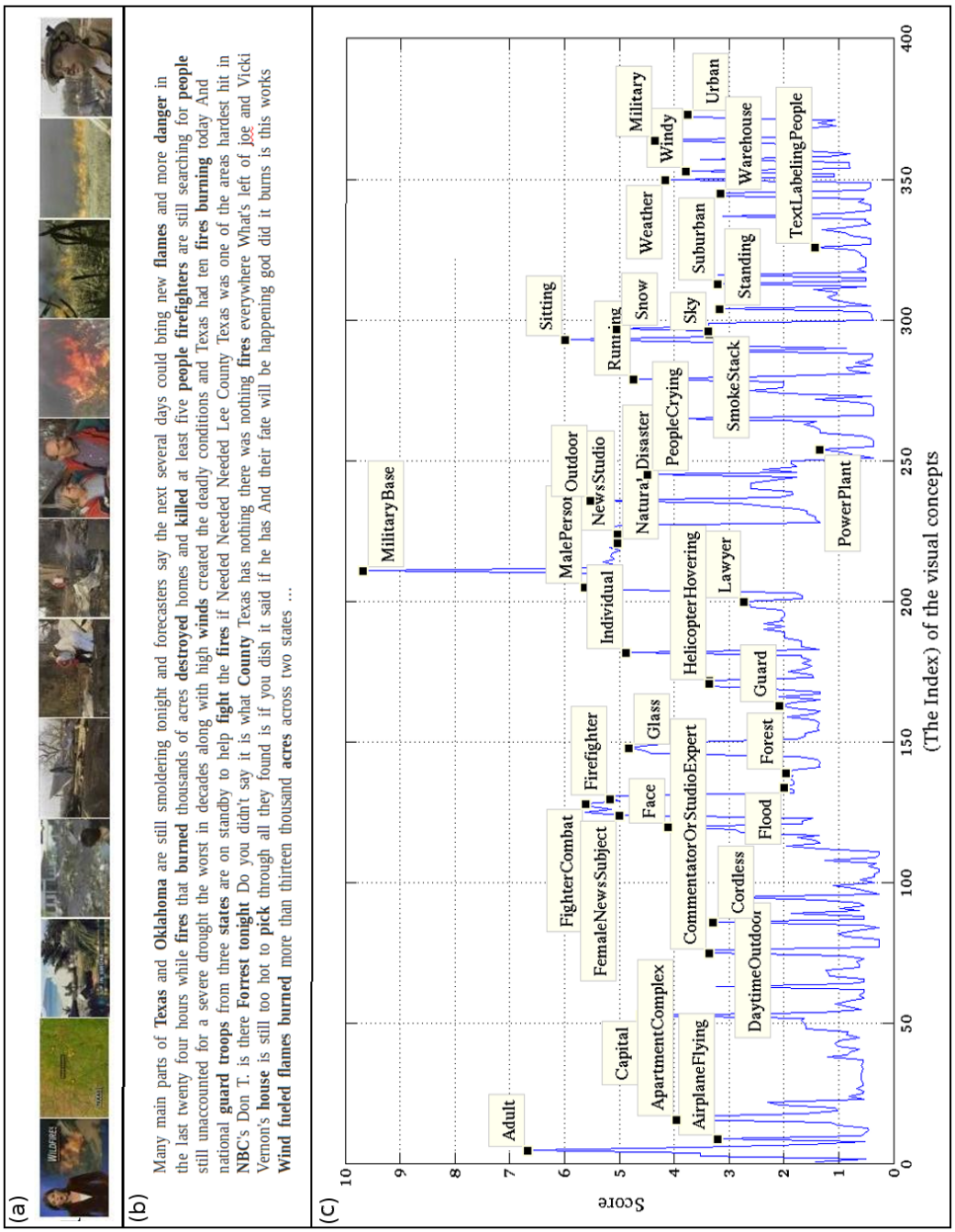

Figure 6.4: An example of a generated visual concept signature using extracted ASR transcript — (a) keyframes, (b) ASR transcript, and (c) generated visual concept signature.

where m is the number of visual concepts. We learn projection functions, $\mathbf{w_t}$ and $\mathbf{w_v}$, from $\mathbf{T}$ and $\mathbf{V}$ extracted from training videos. Specifically, $\mathbf{w_t}$ and $\mathbf{w_v}$ are sets of basis vectors for textual and visual features, respectively, to maximize the following correlation function [51]:

$$\rho = \frac{E[\mathbf{w_t^T T V^\intercal w_v}]}{\sqrt{E[\mathbf{w_t^\intercal T T^\intercal w_t}]E[\mathbf{w_v^\intercal V V^\intercal w_v}]}}. \tag{6.7}$$

The maximum of ρ with respect to $\mathbf{w_t}$ and $\mathbf{w_v}$ is the maximum canonical correlation. We can re-write the above equation as

$$\rho = \frac{E[\mathbf{w_t^\intercal C_{tv} w_v}]}{\sqrt{E[\mathbf{w_t^\intercal C_{tt} w_t}]E[\mathbf{w_v^\intercal C_{vv} w_v}]}}, \tag{6.8}$$

where $\mathbf{C_{tt}}$ and $\mathbf{C_{vv}}$ are the within-set covariance matrices of $\mathbf{T}$ and $\mathbf{V}$, respectively, and $\mathbf{C_{tv}}$ is the between-sets covariance matrix. The canonical correlations between $\mathbf{T}$ and $\mathbf{V}$ can be found by solving the eigenvalue equations [51]

$$\begin{aligned} \mathbf{C_{tt}^{-1} C_{tv} C_{vv}^{-1} C_{vt} w_t} &= \rho^2 \mathbf{w_t} \\ \mathbf{C_{vv}^{-1} C_{vt} C_{tt}^{-1} C_{tv} w_v} &= \rho^2 \mathbf{w_v}, \end{aligned} \tag{6.9}$$

where the eigenvalues ρ^2 are the squared canonical correlations and the eigenvectors $\mathbf{w_t}$ and $\mathbf{w_v}$ are the normalized canonical correlation basis vectors. More details about CCA and possible solutions for Equation (6.7) can be found in [51].

Next, the projection of the textual feature of the query story, $\mathbf{wtfidf_q}$, onto $\mathbf{w_t}$ is given by $\mathbf{wtfidf_q^\intercal} \times \mathbf{w_t}$. Similarly, the semantic signature of a reference story ($\mathbf{SemSig_r}$) is projected to $\mathbf{w_v}$ to obtain $\mathbf{SemSig_r^\intercal} \times \mathbf{w_v}$. Note that we normalize both $\mathbf{wtfidf_q}$ and $\mathbf{SemSig_r}$ to have zero mean in each feature dimension before we calculate their projections. Finally, we retrieve similar reference news stories based on the cosine similarity between their projections and the query story projection, as shown in Figure 6.3. We evaluate the effectiveness of the CCA

approach compared to the semantic similarity mapping approach presented in Section 6.2.1 in the experiment section.

As shown in Figure 6.3, the final similarity score in the proposed early fusion framework is determined by adding the similarity scores in SSM and CCA methods.

6.3 Late Fusion of Textual and Visual Modalities

In this section, we explore the effect of late fusion of textual and visual modalities by combining enhanced textual similarity (Section 5.2.3), enhanced visual similarity (Section 6.1), and early fusion similarities (Section 6.2). Similar to [130], we also use early fusion approach as an individual classifier in addition to the other two classifiers in the proposed late fusion scheme. This is called *double-fusion* approach since it contains both conventional early and late fusion approaches. We use the SVM-based and the Ranked List late fusion methods to fuse the above-mentioned similarity scores. In the SVM-based fusion approach, we employ a leave-one-out training framework and treat similarity scores between pairs of stories in the training dataset as inputs to the SVM classifier. In our case, we have three inputs of the enhanced textual and the visual similarities and the early fusion similarity score. We utilize the RBF kernel and find its optimal parameters (C and γ) through a coarse-to-fine grid search using training data [24]. We use the trained SVM model to calculate the similarity between a given query video and all reference videos and rank them, accordingly. In the Ranked List method [38], first we retrieve and rank reference stories for a given query

story using every single similarity score. Next, we determine the final rank of a reference story as the minimum of the ranks determined using different similarity scores.

6.4 Experimental Results

6.4.1 Dataset and Evaluation Metric

In this section, we evaluate different multi-modal approaches for associated news story retrieval. We use news videos from the TRECVID 2006 collection from seven different channels addressing world news stories in December 2005. It is the same dataset as in Chapter 4 which contains 830 news stories out of which 296 pairs of associated news stories are labeled. We use ASR transcripts provided by [2]. We manually segment the news stories as a group of keyframes and label them based on their main topic.

We consider each associated news story as a query and measure the similarity between the query and all the other stories. The retrieval performance is quantified by the probability of retrieving the associated news stories in the top-k position of the ranked list given as $P(k) = Z_c/Z$, where Z_c is the number of queries that rank their associated news stories within the top-k position and Z is the total number of queries which is 296.

In practice, one can assume that related stories often come on the same days in multiple channels and accordingly reduce the exploring area when looking for associated news stories in the dataset for a given query story. Although it seems an expectable pattern between associated news stories, we did not incorporate it here because first, our main dataset includes only one month of news stories

which is not long enough to investigate the role of incorporating above-mentioned pattern. Particularly, shrinking the exploring area, only few stories would be left for a given query story which can highly affect (in most of the cases boost) the retrieval performance. This may artificially shorten differences between performances of various retrieval algorithms that we aim to compare. Second, there are associated news story pairs which are temporally far (e.g. couple of days) from each other since the latter news story is about the updates on the topic of interest (e.g. "Saddam Trail"), initially presented in the former news story. Considering the mentioned pattern, we might lose these associations.

6.4.2 Enhanced Visual Similarity Evaluation

Figure 6.5 shows the retrieval performance using various visual similarities — keyframe set similarity in Equation (6.1), semantic signature similarity explained in Section 6.1.1, their ranked list and SVM-based fusion (Section 6.1.3), and the SVM-based fusion of keyframe set similarity and the weighted semantic signature similarity (Equation 6.5). The semantic signature similarity performs worse than keyframe set similarity, however, it is better than random retrieval. Because semantic signature is not discriminative enough due to the limited number of visual concepts (i.e., 374) and relatively low precisions of visual concept detectors. The poor retrieval performance of keyframe set similarity can be explained as since majority of the associated news stories belong to the third category discussed in Section 1.3, where this visual representation is not discriminative enough. The SVM-based fusion of keyframe set similarity and semantic signature similarity outperforms the Ranked List fusion method for the top-1 through the top-17 retrieval results. This superiority comes from the fact that through the SVM-based

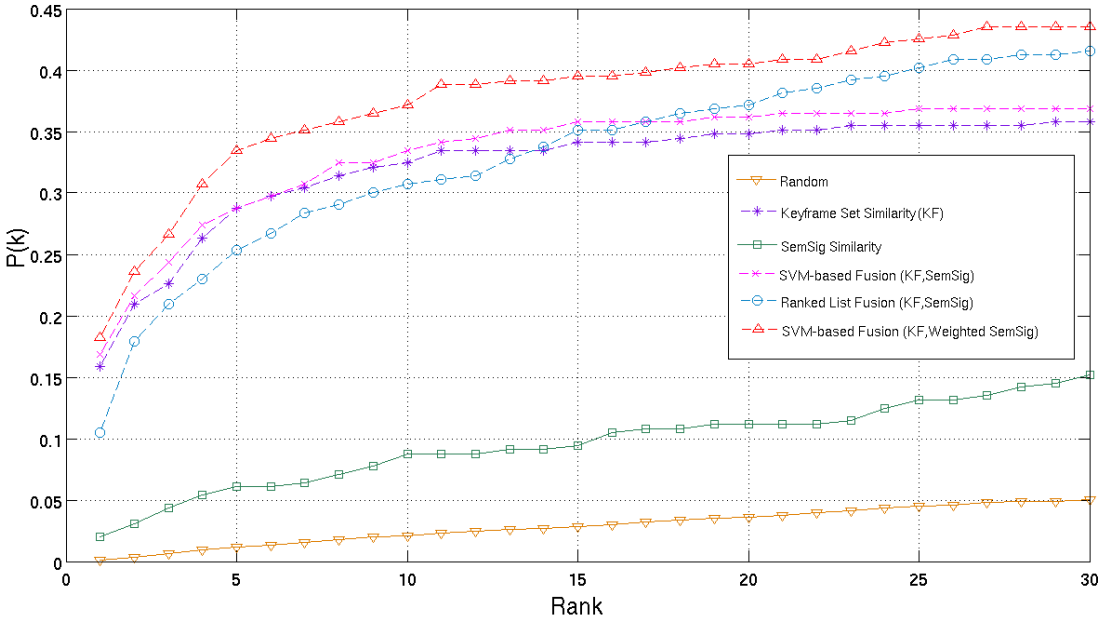

Figure 6.5: The top-k retrieval result using visual modalities — The best result belongs to the SVM-based fusion of keyframe set similarity and weighted semantic signature similarity.

fusion we learn how to fuse similarity scores using training data as explained in Section 6.1.3, while the Ranked List fusion is an unsupervised method where no training is used. The best result is obtained by the SVM-based fusion of keyframe set similarity and weighted semantic signature similarity. This superiority explains the key role of our proposed Detectability Score, (Equation (6.4)), to weight the visual concepts and facilitates the effective combination of keyframe set and semantic signature similarities.

6.4.3 Early Fusion Evaluation

In Figure 6.6, we compare the retrieval results using early fusion strategies for textual and visual information as explained in Section 6.2. It is clear that non of SSM, CCA or their linear fusion performs well, although all of them performs

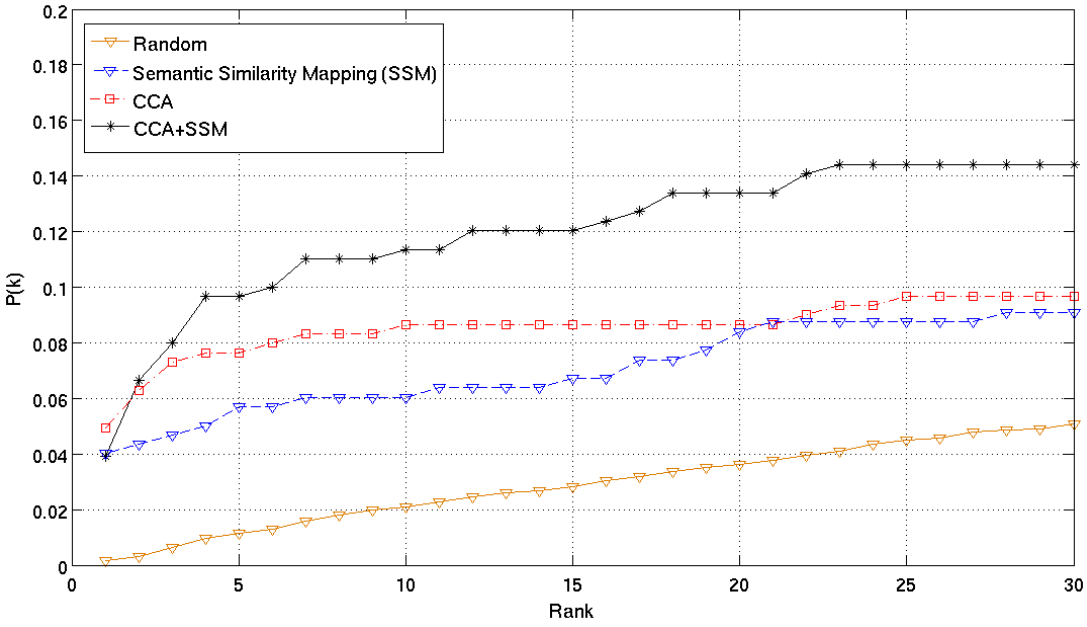

Figure 6.6: The top-k retrieval result using the proposed early fusion methods — The best result is obtained using the linear fusion of CCA and SSM outputs.

better than random retrieval. The linear fusion and CCA perform better than the SSM method. A possible explanation for these relatively poor retrieval results is that in the SSM and the CCA methods, we retrieve semantic signatures that are not discriminative enough due to limited number of visual concepts and relatively low precision of visual concept detectors. Moreover, noisy ASR and OCR transcripts can also decrease the quality of the generated visual concept signatures, which highly affects the SSM retrieval performance.

Although the SSM method does not perform well for associated news story retrieval, there are other contexts where it can play a more effective role on video retrieval. Particularly, in video retrieval tasks using only text-based query, we can retrieve visual concepts in a video using visual concept signature generated

based on a given text query. For instance in [123], visual concept signature idea, explained in Section 6.2.1, was used to tackle the ad-hoc Multimedia Event Detection task. Multimedia Event Detection (MED) is a multimedia retrieval task with the goal of finding videos of a particular event (e.g. "getting a vehicle unstuck", "wedding", "making a sandwich", etc) in a large-scale internet video archive, given text descriptions of events. In [123], the test videos were retrieved based on their visual semantics using a Visual Concept Signature (VCS) generated for each event only derived from the event description provided as the query. Visual semantics are described using the Semantic Indexing (SIN) feature which represents the likelihood of predefined visual concepts in a video, similar to SemSig in Equation (6.2) but using less number of visual concepts (i.e. 346). To generate a VCS for an event, the given event description was projected on to a visual concept list using the proposed textual semantic similarity, similar to **G_SemSig** explained in Section 6.2.1. Exploring SIN feature properties, the generated visual concept signature and the SIN feature were harmonized to improve retrieval performance. The test videos were retrieved using the cosine similarity between their SIN feature and the generated VCS for each event. The quality of the proposed VCS was assessed with respect to human expectations for each event. Through a set of experiments on the MED task [130], the effectiveness of the VCS and the harmonizing step to retrieve the test videos were shown. More details about the proposed approach and the experiment results can be found in [123].

6.4.4 Late Fusion Evaluation

In Figure 6.7, we compare retrieval results using different multi-modal late fusion strategies. First, comparing different single modalities, we found that the enhanced textual similarity, (Equation 5.4), outperforms the enhanced visual content similarity, explained in Section 6.1.3. The possible explanation for this superiority is that almost all news stories have ASR/OCR transcripts while only some of them contain salient and informative visual content. Furthermore, the accuracy of ASR transcript is relatively higher than that of visual features such as visual concepts, since a news story is mainly recorded in a studio, which provides an audio channel with high quality from which an accurate ASR transcript can be extracted.

The second best result is obtained by the SVM-based fusion of enhanced textual and visual similarities and early fusion, which outperforms the Ranked List fusion performance. Since in the SVM-based fusion, we combine different similarities through a learning process explained in Section 6.3, we obtain a better performance compared to the Ranked List fusion which is an unsupervised method.

The best result is obtained by the SVM-based fusion of enhanced textual similarity, early fusion similarity, and enhanced visual similarity using scene signature set similarity instead of keyframe set similarity as explained in Section 6.1.3. As shown in Figure 6.7, we could capture around 43.5% of associated news stories in the top-1 retrieval result. This result confirms the effectiveness of our enhanced textual and visual representations and similarities and their complementary role to capture semantics inherent in a news story.

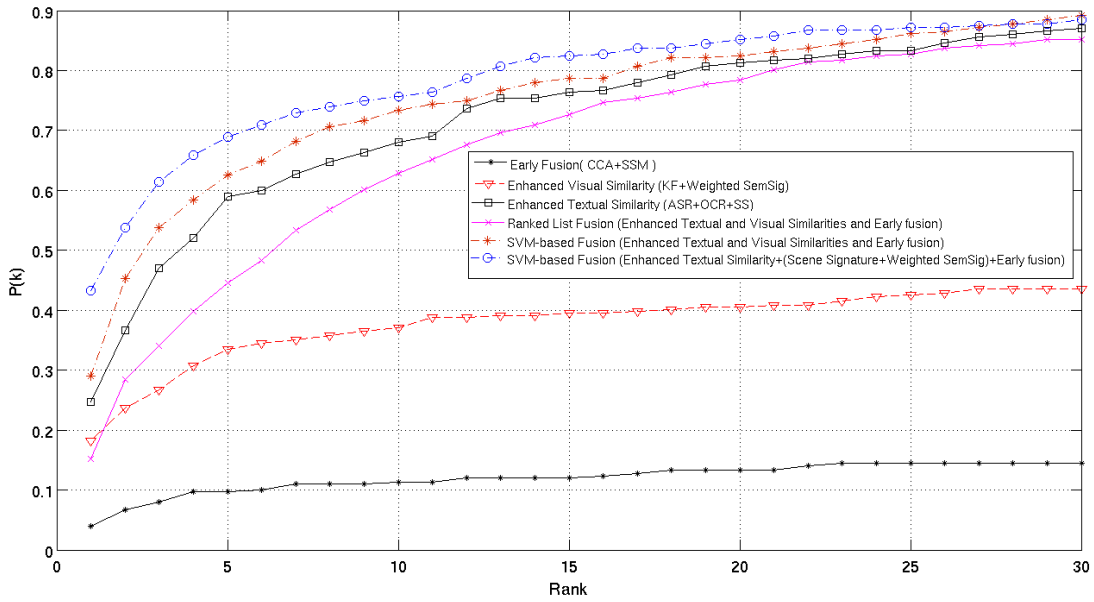

Figure 6.7: The top-k retrieval result using different modalities with different fusion strategies. — The best result is obtained by integrating our enhanced textual similarity score, scene signature and weighted semantic signature similarity scores, and early fusion similarity scores through the SVM-based late fusion.

6.4.5 Discussion

To elaborate the contribution of each modality and the proposed refinement steps for the case of the best performance, we demonstrate the trend of performance improvement in Figure 6.8. The vertical axis denotes the average of the top-5 retrieval results and the horizontal axis shows the accumulative set of modalities and refinement steps. **wtfidf**, explained in Equation (5.2), is the best single feature using which we could retrieve 39% of associated news stories in the top-1 through the top-5 in average. Incorporating textual semantic similarity (Section 5.2), the performance is improved by around 5%. Incorporating scene set similarity, the performance is improved by around 9%. Integrating weighted semantic signature

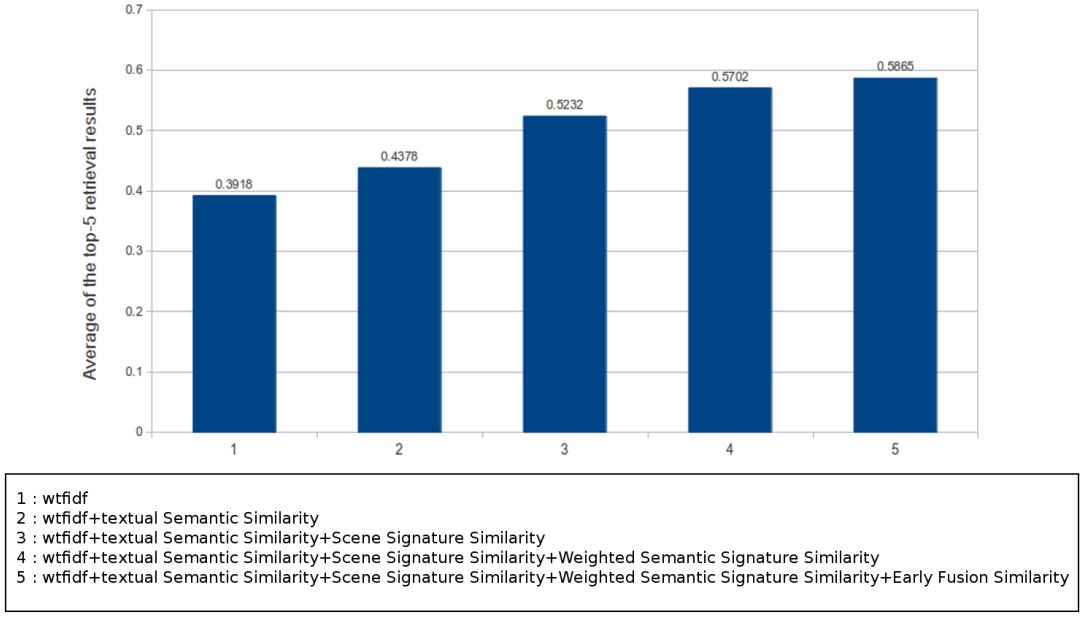

Figure 6.8: The contribution of the different modalities and refinement steps in the best performance.

similarity, we boost the performance by around 5%.

To investigate the origin of this improvement, we track down the retrieval results and observe that by incorporating the semantic signature similarity, we could improve the retrieval performance for a few clusters of news stories with the specific topics such as "New York subway strike" and "Fire in Texas and Oklahoma". This observation shows the effectiveness of the semantic signature to capture conceptual relationships between news stories within these clusters. On the other hand, there are clusters of stories with the specific topics such as "Saddam Hussein court" or "Fassir war tour" where the calculated scene set similarity is significantly greater than the semantic similarity since these clusters of stories belong to the first or the second category of associated news stories and they share the same or very similar picture, object and/or venue.

Note that in this study we focus on a narrow spectrum of early fusion schemes to improve the retrieval performance where we do not have sufficient textual information to be used for the retrieval purpose. The proposed early fusion used textual information to retrieve visual concepts in news stories. However, fused by other modalities, it does not contribute significantly to the final performance and only improves the performance by less than 2%, which is mainly due to limited number of visual concepts and relatively low precisions of visual concept detectors.

There are also other early fusion approaches that we have not explored here. For instance, early fusion methods which combine different features into a long feature through which it can implicitly model the correlations between them. These early fusion do not perform well, if the feature of different modalities is too heterogeneous with skewed length distribution and significantly different scales, which is the case here according to the set of features used. In contrast, in the late fusion, this is not a concern since the features from each modality will not compare with each other before the final fusion step. In addition, we can employ various detection techniques and classifiers according to specific feature types in the late fusion framework. Moreover, the late fusion methods are usually less computationally expensive compared with the early fusion approaches. Therefore, the late fusion techniques have become more popular and more extensively studied than early fusion techniques in the literature [107]. Here we mainly focused on the late fusion strategies where the textual and visual similarity modules provide the uni-modal decision scores which later are combined through a decision fusion module to obtain the final decision score. We investigated different fusion modules such as the Ranked List method as a rule-based approach and the SVM-

based method as a classification-based approach. As shown in Figure 6.7, latter outperforms former. However, both performs significantly better than the early fusion approach.

6.5 Conclusion

We introduced a semantic signature that represents a news story using predefined visual concepts. We determined enhanced visual content similarity by integrating local signature similarity (such as scene signature) and semantic signature similarity. We learned from failure cases of local signature similarity to tune the contribution of semantic signature similarity by assigning a proper weight to each visual concept. Next, we fused heterogeneous sources of knowledge, i.e. enhanced textual and visual similarity, through different early and late fusion strategies. In the proposed early fusion scheme, textual information is used to retrieve the visual semantic signatures. We studied different late fusion schemes to combine decisions of textual, visual and early fusion modules. The best result is obtained by fusing the proposed **wtfidf**, scene and weighted semantic signature similarities, and the early fusion similarity through the SVM-based late fusion.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

In this thesis, we investigated the associated news story retrieval problem using different modalities. We studied different approaches to determine textual and visual representation and similarity of news stories. In the textual domain, associated news stories usually share some spoken words and/or displayed words (appear as a closed caption) which are extracted through Automatic Speech Recognition (ASR) and Optical character Recognition (OCR), respectively. Since OCR transcripts have high error rate, we proposed a novel post-processing approach based on the local dictionary idea to recover the erroneous OCR output and identify more informative words. We generated an enhanced textual content representation using ASR transcript and OCR keywords through an early fusion scheme. We also employed a textual semantic similarity measure to exploit the relatedness of the textual features using WordNet and Wikipedia.

In the visual domain, associated news stories can be seen as duplicate, Near-duplicate, partially near-duplicate videos or in more challenging cases as videos sharing specific visual concepts. We investigated Near-Duplicate Keyframe iden-

tification task as the main core of the visual analysis based on which we proposed a novel video signature, called scene signature. Moreover, we generated a visual semantic signature for a news video using predefined visual concepts. Next, we combined these two sources of the visual knowledge (i.e. scene signature and semantic signature) to determine the enhanced visual content similarity.

Finally, we incorporated all enhanced textual and visual representations/similarities through an early/late fusion scheme, respectively, to study their complementary roles in the associated news story retrieval task. We evaluated our proposed NDK retrieval, detection and clustering approaches in extensive experiments on standard datasets. We also assessed the effectiveness and compactness of our proposed scene signature to represent a video compared to other local and global video signatures using a web video dataset. Finally, we showed the usefulness of multi-modal approaches using different textual and visual modalities. The experimental results showed that the proposed retrieval system benefits from the textual modality more than the visual modality. However, the best result was obtained through their fusion.

7.2 Future Work

In the research reported here, investigation were sometimes restricted to basic approaches in order to be able to focus on the important aspects of the associated news story retrieval. Moreover, some findings in this work are potentially applicable to other areas in the multimedia analysis and retrieval. Therefore, brief descriptions of these investigations are proposed for future research and listed in the following.

7.2.1 Interactive Associated News Story Retrieval

Having human in the loop, we can provide users with a better associated news story browsing and retrieval. More specifically, we can use human to bring more semantic information into the system by tagging each news story with relevant keywords. Later, we can use these keywords in addition to all textual/visual features, extracted and learned automatically, to explore association between stories, more effectively. There are many researches addressing the interactive indexing and annotation of multimedia content. For instance, in [23] authors proposed a graphical framework for video content that aims to facilitate a quick understanding and an interpretation of the semantic content of a video sequence. The proposed tool also offers functionalities for automatically and semi-automatically retrieving and annotating the shots. Such a tool emphasizes on the users fundamental role when annotating and accessing multimedia corpora.

In addition to interactive solution for multimedia indexing, we can also incorporate human to boost the quality of the retrieval results. To do so, we need to re-train the proposed retrieval system using the relevance feedbacks provided by human for an initial retrieval results for a given query. In the literature, there are many researches using relevance feedback techniques for different purposes (e.g feature selection, parameter configuration, etc) to design a more effective multimedia retrieval system. For instance, authors in [12] proposed a relevance feedback technique that captures the significance of different features at different spatial locations in an image. Spatial content is described by partitioning images into non-overlapping grid cells. Contributions of different features at different locations are modeled using weights defined for each feature in each grid cell. These

weights are iteratively updated according to user feedback in terms of positive and negative labeling of retrieval results. Experimental results on TRECVID data show that the learned weights could capture relative contributions of different features and spatial locations.

Another example is K-Space [59], Knowledge space of semantic inference for automatic annotation and retrieval of multimedia content, which is a network of leading European research teams from academia and industry conducting collective research in semantic inference for automatic and semi-automatic annotation and retrieval of multimedia content. The goal is to bridge the semantic gap between content descriptors computed automatically by algorithms, and richness of high-level semantics provided as human annotations of audio/video media. The K-Space search interface enables users to upload and index a video content and to search for a specific content. Then, user can refine the search by scoring the initial search results based on their relevance.

7.2.2 News Story Summarization and Recounting

Autonomous news story summarization is a practical news-related application which aims to generate high quality textual and visual summaries of a news story. Descriptive representation of a news story and localizing its more informative sections are key elements of an effective news story summarization system. Similar to associated news story retrieval task, different textual and visual modalities can be studied to obtain a unique and semantic representation of a news story. Metadata such as the order of frames and their time stamps can be also beneficial. Finding more informative and important parts of determined textual and visual representations can be investigated based on within or between stories comparison and

analysis. Finally, a news story can be summarized as few sentences, highlighting the main topic, and a storyboard or a skimmed video [105], indicating distinctive visual clues.

Against conventional news story summarization, in news story recounting the goal is to convert the visual summarization results onto textual representation and describe it in few sentences using natural language processing. The recounting concept has been recently studied in the multimedia event domains [1, 36] where the goal is to recount the important evidence that led the Multimedia Event Detection system to conclude that a particular multimedia clip contains an instance of particular event.

Thus, semantic representation of visual cues, e.g. predefined visual concepts used in this research, can be critical for news story visual annotation and recounting. Using recounting to create an abstract textual description of a news story, we can employ extensive researches, done in text-based information retrieval, to address different tasks such as news story understanding, threading and clustering. We can also provide a fast navigation and browsing of news stories where each news story is abstracted in one or two sentences.

7.2.3 Multimedia Event Detection

Multimedia Event Detection (MED) [130] is a multimedia retrieval task with the goal of finding videos of a particular event in a large-scale Internet video archive, given example videos and text descriptions. There are two scenarios where we either do or do not use the provided example videos. In the former, the problem becomes similar to associated new story retrieval, however, it contains a wide range of Internet videos with different topics, and audio/visual quality which can

make a difficulty in detection process. In the 'ad-hoc' scenario, where we do not have any training data, we can only use given text description to retrieve similar videos using their ASR and OCR transcript. Another interesting direction is to visualize the given text description, similar to visual concept signature generation explained in Section 6.2.1, and use it to retrieve semantic signatures of related videos, as briefly discussed in Section 6.4.3.

7.2.4 News Recommendation

Online news reading and watching have become very popular as the Internet provides access to news articles from thousands of sources around the globe. A main challenge of news service website is to help users to find news articles that are interesting to read or watch. To do so, a recommendation system should be built with a profile for each user to estimate his/her news interest based on the watched news stories. Recommending news article is traditionally done by term-based algorithms like TF-IDF [58]. In addition, we can also use other modalities such as scene or semantic signature to describe a news story. Using the created profiles and different signatures extracted from news stories, we can explore different recommendation techniques to retrieve news stories which are close to a user taste.

A good example of news recommendation system is Mesh "Multimedia Semantic Syndication for Enhanced News Services" [43]. Mesh Project proposed a complete framework to extract, compare and integrate content from multiple multimedia news sources. It automatically generates personalized news summaries, and links summaries and content based on the extracted semantic information, and provides end users with a "multimedia mesh" news navigation system. Mesh

project also uses the personalization outline by investigating the user preferences while searching the web and automatically suggests useful information.

7.2.5 Acoustic Concept Detector

Similar to visual concepts, there are acoustic concept detectors which can be used to semantically index a video based on predefined acoustic classes such as 'water splashing', 'engine noise', 'crowd', 'music, 'cheering', etc. They are mainly determined using MFCC features [17], extracted from audio channel, for each acoustic segment. Authors in [122] used 42 acoustic concepts, in addition to other textual and visual features, to describe different event for the multimedia event detection task. The acoustic concepts improve the performance especially for events in which there are too few spoken words.

Similar to the proposed semantic signature explained in Section 6.1, we can generate an acoustic signature for a news story and determine acoustic similarity between news stories. This acoustic signature and similarity can be especially useful when we deal with the *SOT* news story. *Sound on tape*, or *SOT*, is sound and/or video, usually recorded in the field. It is usually an interview or sound bite which is a short piece of actual sound from the event reported on.

7.3 Future of Video Retrieval

Here, we outline two perspectives of future of the video retrieval systems.

7.3.1 Effective and Efficient Fusion of Various Features

As shown in Section 6.4.4, combination of more features such as text, audio and visual feature would improve the performance of video retrieval systems. How-

ever, very few works have been done in effective and efficient fusion of different features for video retrieval.

In this research, in addition to the rule-based fusion approaches such as the ranked list approach, we have also studied the SVM-based fusion approach where we learn the weights to combine different modality similarities using a group of associated news stories as the training data (Section 6.3). Alternatively, we can also learn these weights for different clusters of news stories sharing a semantic property. This concept has been studied in the literature in the context of learning query-class dependent weights in video retrieval [120]. For instance, we can cluster news stories into different groups based on the amount of textual information they contain, and learn different set of weights to combine different similarity scores, calculated in different modalities, for each cluster of news stories.

Actually, we used a similar approach to combine the semantic signature similarity and the scene signature similarity, explained in Section 6.1.3. We put more weights on the semantic signature similarity for news stories which contain specific visual concepts. We can extend and employ the same idea when we fuse textual and visual similarities. In this case, the main challenge would be finding a proper visual/textual semantic property based on which we cluster news stories and then determine the fusion configuration for them.

7.3.2 Concept-based Video Retrieval under Improved Detector Performance

The recent content-based video retrieval systems mainly focus on the improvement of concept detectors [106]. On the other hand there is research on developing retrieval models to combine the output of concept detectors to address informa-

tion needs of users. Although concept-based retrieval is generally a promising retrieval scheme, the currently available concept detector engines are not accurate and fast enough for large-scale application in real-life. Clearly, week detector performance highly degrades the retrieval/search performance.

Authors in [53] used a simulation-based approach to predict the achievable performance of concept-based video retrieval engines. Using the TRECVID [2] video collection and the LSCOM [6] truth annotations of 300 concepts, they simulated performance of video retrieval under different assumptions of concept detection accuracy. They showed that "concept-based" video retrieval with fewer than 5,000 concepts, detected with minimal accuracy of 10% mean average precision could be sufficient to support a high accuracy video retrieval system, comparable to text retrieval on the web, in a typical broadcast news collection. This observation is consistent with our findings in this research. As reported in Section 6.4.2, using only 374 visual concepts to generate SemSig, we could capture only 6% of associated news stories in the top-5, while using textual information we could detect around 60% of associated news stories in the top-5. According to [53], we might need more than ten times more visual concepts to achieve a comparable performance. However, the main question is still remained unanswered: which specific concepts should be used?

There is a hope that the divide-and-conquer approach using large numbers of semantic concepts as an intermediate step enables us to develop thousands of concepts that can be robustly detected in different contexts. We hope that with sufficient numbers of these concepts available, indexing a wide range of visible things, users will ultimately be able to contract the semantic gap. These observations may serve as a starting point for the future studies considering a

large set of concepts for retrieval.

References

[1] TRECVid multimedia event recounting evaluation track. http://www.nist.gov/itl/iad/mig/mer.cfm. 161

[2] TRECVID(2006). http://www-nlpir.nist.gov/projects/tv2006/tv2006.html. ix, 17, 68, 129, 136, 139, 147, 165

[3] Twitter. http:/www.twitter.com/. 5

[4] Wikipedia, the free encyclopedia. http://www.wikipedia.org/. 16, 34, 122

[5] Youtube video search engine. http://www.youtube.com/. 1, 5, 18, 100

[6] LSCOM lexicon definitions and annotations version 1.0, DTO challenge workshop on large scale concept ontology for multimedia. Technical Report 217–220, 2006. 135, 165

[7] Aspell: free and open source spell checker. http://www.aspell.net/, 2010. 118

[8] JOCR: free optical character recognition engine. http://www.jocr.sourceforge.net/, 2010. 17, 116

[9] N. Adami, S. Benini, and R. Leonardi. An overview of video shot clustering and summarization techniques for mobile applications. In *MobiMedia '06: Proceedings of the 2nd International Conference on Mobile Multimedia Communications*, pages 27:1–27:6, New York, NY, USA, 2006. ACM. 24

[10] W. H. Adams, G. Iyengar, M. R. Naphade, C. Neti, H. J. Nock, and J. R. Smith. Semantic indexing of multimedia content using visual, audio and text cues. *EURASIP Journal on Applied Signal Processing*, 2:170–185, 2003. 38

[11] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *VLDB '94: Proceedings of the 20th International Conference on Very Large Data Bases*, pages 487–499. Morgan Kaufmann, 1994. ix, 91

[12] S. Aksoy and O. Cavus. A relevance feedback technique for multimodal retrieval of news videos. In *EUROCON '05: Processing of The International Conference on Computer as a Tool*, volume 1, pages 139–142, 2005. 159

[13] P. K. Atrey, M. A. Hossain, A. El Saddik, and M. S. Kankanhalli. Multimodal fusion for multimedia analysis: A survey. *Multimedia Systems*, 16(6):345–379, 2010. 35, 36, 38

[14] Y. Aytar, M. Shah, and J. Luo. Utilizing semantic word similarity measures for video retrieval. In *CVPR '08: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 2, 36

[15] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded up robust features. In *ECCV '06: Proceedings of the 9th European Conference of Computer*

Vision, volume 3951 of *Lecture Notes in Computer Science*, pages 404–417. Springer Berlin Heidelberg, 2006. 27

[16] J. S. Beis and D. G. Lowe. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *CVPR '97: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1000–1007, Los Alamitos, CA, USA, 1997. IEEE Computer Society. 88

[17] A. Berenzweig, B. Logan, D. P. W. Ellis, and B. P. W. Whitman. A large-scale evaluation of acoustic and subjective music-similarity measures. *Computer Music Journal*, 28(2):63–76, 2004. 163

[18] E. Borovikov, I. Zavorin, and M. Turner. A filter based post-OCR accuracy boost system. In *HDP '04: Proceedings of the 1st ACM Workshop on Hardcopy Document Processing*, pages 23–28, New York, NY, USA, 2004. ACM. 32, 131

[19] P. S. Boyd and R. Alexander. *Broadcast Journalism: Techniques of Radio and Television News*. Focal Press, 2008. 5, 6, 7

[20] D. Brezeale. Using closed captions and visual features to classify movies by genre. In *MDM/KDD '06: Poster session of the 7th International Workshop on Multimedia Data Mining*, 2006. 32

[21] D. Brezeale and D. Cook. Automatic video classification: A survey of the literature. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(3):416–430, 2008. 26

[22] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998. 38

[23] M. Campanella, R. Leonardi, and P. Migliorati. Interactive visualization of video content and associated description for semantic annotation. *Signal, Image and Video Processing*, 3(2):183–196, 2009. 159

[24] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`. 146

[25] E. Y. Chang, J. Z. Wang, C. Li, and G. Wiederhold. Rime: A replicated image detector for the world-wide web. In *Proceedings of SPIE Symposium of Voice, Video, and Data Communications*, pages 58–67, 1998. 20

[26] S.-F. Chang, W. Hsu, L. Kennedy, and L. Xie. Video search and high-level feature extraction. *Proceedings of NIST TREC Video Retrieval Evaluation*, 2005. 21, 41

[27] B.-W. Chen, J.-C. Wang, and J.-F. Wang. A novel video summarization based on mining the story-structure and semantic relations among concept entities. *IEEE Transactions on Multimedia*, 11:295–312, 2009. 62, 67

[28] X. Cheng, Y. Hu, and L.-T. Chia. Image near-duplicate retrieval using local dependencies in spatial-scale space. In *MM '08: Proceedings of the 16th ACM International Conference on Multimedia*, pages 627–630, New York, NY, USA, 2008. ACM. 22, 68, 71, 73

[29] S. Cheung and A. Zakhor. Efficient video similarity measurement with video signature. In *ICIP '02: Proceedings of IEEE International Conference on Image Processing*, volume 1, pages 59–74, 2002. 26

[30] S. S. Cheung and A. Zakhor. Efficient video similarity measurement with video signature. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(1):59–74, 2003. 26

[31] L. Ching-Yung, B. Tseng, M. Naphade, A. Natsev, and J. Smith. VideoAL: A novel end-to-end MPEG-7 video automatic labeling system. In *ICIP '03: Proceedings of IEEE International Conference on Image Processing*, volume 3, pages 53–59, 2003. 20

[32] O. Chum, J. Philbin, M. Isard, and A. Zisserman. Scalable near identical image and shot detection. In *CIVR '07: Proceedings of the 6th ACM International Conference on Image and Video Retrieval*, pages 549–556, New York, NY, USA, 2007. ACM. 29

[33] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *CVPR '00: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 142–149. IEEE Computer Society, 2000. 57

[34] U. Damnjanovic, T. Piatrik, D. Djordjevic, and E. Izquierdo. Video summarisation for surveillance and news domain. In *SAMT '07: Proceedings of the Semantic and Digital Media Technologies 2nd International Conference on Semantic Multimedia*, pages 99–112, Berlin, Heidelberg, 2007. Springer-Verlag. 24

[35] D. Das, D. Chen, and A. G. Hauptmann. Improving multimedia retrieval with a video OCR. In *SPIE '08: Society of Photo-Optical Instrumentation Engineers Conference Series*, volume 6820, page 68200B. SPIE, 2008. 15, 32

[36] D. Ding, F. Metze, S. Rawat, P. F. Schulam, S. Burger, E. Younessian, L. Bao, M. G. Christel, and A. Hauptmann. Beyond audio and video retrieval: towards multimedia summarization. In *ICMR '12: Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*, pages 2:1–2:8, New York, NY, USA, 2012. ACM. 161

[37] Q. Do, D. Roth, M. Sammons, Y. Tu, and V. Vydiswaran. Robust, light-weight approaches to compute lexical similarity. Technical report, University of Illinois, 2009. 34, 124, 125

[38] K. Donald and A. Smeaton. A comparison of score, rank and probability-based fusion methods for video shot retrieval. In *CIVR '05: Proceedings of the 1st International Conference on Image and Video Retrieval*, volume 3568 of *Lecture Notes in Computer Science*, pages 61–70. Springer Berlin Heidelberg, 2005. 37, 146

[39] W. Dong, Z. Wang, M. Charikar, and K. Li. Efficiently matching sets of features with random histograms. In *MM '08: Proceedings of the 16th ACM International Conference on Multimedia*, pages 179–188, New York, USA, 2008. ACM. 26

[40] M. Douze, A. Gaidon, H. Jegou, M. Marszalek, and C. Schmid. INRIA-

LEARs video copy detection system. In *Proceedings of the TREC Video Retrieval Evaluation (TRECVID)*, 2008. 84

[41] A. Farahat and M. Kamel. Document clustering using semantic kernels based on term-term correlations. In *ICDMW '09: Proceedings of IEEE International Conference on Data Mining Workshops*, pages 459 –464, 2009. 16

[42] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *CVPR '03: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 264–271, 2003. 21

[43] N. Fernández, J. M. Blázquez, J. A. Fisteus, L. Sánchez, M. Sintek, A. Bernardi, M. Fuentes, A. Marrara, and Z. Ben-Asher. NEWS: Bringing semantic web technologies into news agencies. In *ISWC '06: Proceedings of the 5th international conference on The Semantic Web*, pages 778–791, Berlin, Heidelberg, 2006. Springer-Verlag. 162

[44] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. Query by image and video content: the QBIC system. *Computer*, 28(9):23–32, 1995. 2

[45] Y. Gao and Q.-H. Dai. Shot-based similarity measure for content-based video summarization. In *ICIP '08: Proceedings of the 15th IEEE International Conference on Image Processing*, pages 2512–2515, 2008. 61

[46] D. Gatica-Perez, A. Loui, and M.-T. Sun. Finding structure in home videos by probabilistic hierarchical clustering. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(6):539–548, 2003. 24

[47] K. Grauman and T. Darrell. The pyramid match kernel: Discriminative classification with sets of image features. In *ICCV '05: Proceedings of IEEE International Conference on Computer Vision*, pages 1458–1465, 2005. 21

[48] M. R. Gupta, N. P. Jacobson, and E. K. Garcia. OCR binarization and image pre-processing for searching historical documents. *Pattern Recognition*, 40(2):389–397, 2007. 116

[49] L. Hagen and A. B. Kahng. New spectral methods for ratio cut partitioning and clustering. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 11(9):1074–1085, 1992. 62

[50] A. Hampapur and R. M. Bolle. Comparison of distance measures for video copy detection. In *ICME '01: Proceedings of IEEE International Conference on Multimedia and Expo*, pages 22–25. IEEE, 2001. 20

[51] D. R. Hardoon, S. R. Szedmak, and J. R. Shawe-taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural Computation*, 16(12):2639–2664, 2004. 145

[52] A. Hauptmann, Y. Chen, M. Christel, C. Huang, W.-H. Lin, T. Ng, N. Papernick, A. Velivelli, J. Yang, R. Yan, H. Yang, and H. D. Wactlar. Confounded expectations: Informedia at trecvid 2004. In *Proceedings of NIST TREC Video Retrieval Evaluation*, 2004. 2

[53] A. Hauptmann, R. Yan, W.-H. Lin, M. Christel, and H. Wactlar. Can high-level concepts fill the semantic gap in video retrieval? a case study with broadcast news. *IEEE Transactions on Multimedia*, 9(5):958–966, 2007. 165

[54] A. G. Hauptmann, R. Jin, and T. D. Ng. Multi-modal information retrieval from broadcast video using OCR and speech recognition. In *JCDL '02: Proceedings of the 2nd ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 160–161, New York, NY, USA, 2002. ACM. 15, 32, 115, 131

[55] T. C. Hoad and J. Zobel. Video similarity detection for digital rights management. In *ACSC '03: Proceedings of the 26th Australasian Computer Science Conference*, pages 237–245, Darlinghurst, Australia, 2003. Australian Computer Society, Inc. 7

[56] C.-R. Huang and C.-S. Chen. Video scene detection by link-constrained affinity-propagation. *ISCAS '09: Proceedings of IEEE International Symposium on Circuits and Systems*, pages 2834–2837, 2009. 27

[57] Z. Huang, H. T. Shen, J. Shao, B. Cui, S. Member, and X. Zhou. Practical online near-duplicate subsequence detection for continuous video streams. *IEEE Transactions on Multimedia*, 12(5):386–398, 2010. 30

[58] W. IJntema, F. Goossen, F. Frasincar, and F. Hogenboom. Ontology-based news recommendation. In *Proceedings of the EDBT/ICDT Workshops*, pages 16:1–16:6, New York, NY, USA, 2010. ACM. 162

[59] E. Izquierdo, K. Chandramouli, M. Grzegorzek, and T. Piatrik. K-space content management and retrieval system. In *ICIAPW '07: Processing of the 14th International Conference on Image Analysis and Processing Workshops*, pages 131–136, 2007. 160

[60] Y. G. Jiang, J. Yang, C. W. Ngo, and A. G. Hauptmann. Representations of keypoint-based semantic concept detection: A comprehensive study. *IEEE Transactions on Multimedia*, 12(1):42–53, 2009. 135

[61] N. Jojic, N. Petrovic, and T. Huang. Scene generative models for adaptive video fast forward. In *ICIP '03: Proceedings of IEEE International Conference on Image Processing*, volume 2, pages 619–22, 2003. 21

[62] K. Jung. Text information extraction in images and video: A survey. *Pattern Recognition*, 37(5):977–997, 2004. 32

[63] Y. Ke, R. Sukthankar, and L. Huston. An efficient parts-based near-duplicate and sub-image retrieval system. In *MM '04: Proceedings of the 12th Annual ACM International Conference on Multimedia*, pages 869–876, New York, NY, USA, 2004. ACM. 22, 23, 41

[64] J. Kittler. On the accuracy of the sobel edge detector. *Image and Vision Computing*, 1(1):37–42, 1983. 85, 86

[65] P. Kolb. Experiments on the difference between semantic similarity and relatedness. In *NODALIDA '09: Proceedings of the 17th Nordic Conference of Computational Linguistics*, volume 4, pages 81–88. Northern European Association for Language Technology, 2009. 34, 125

[66] J. Law-To, O. Buisson, V. Gouet-Brunet, and N. Boujemaa. Robust voting algorithm based on labels of behavior for video copy detection. In *MM '06: Proceedings of the 14th Annual ACM International Conference on Multimedia*, pages 835–844, New York, NY, USA, 2006. ACM. 29

[67] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR '08: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2169–2178, 2006. 21, 70, 71, 73

[68] C. Leacock, G. A. Miller, and M. Chodorow. Using corpus statistics and WordNet relations for sense identification. *Computing Linguistic*, 24:147–165, 1998. 34, 124

[69] L. Lee. Measures of distributional similarity. In *ACL '99: Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 25–32, Stroudsburg, PA, USA, 1999. Association for Computational Linguistics. 34, 125

[70] F.-F. Li and P. Perona. A Bayesian hierarchical model for learning natural scene categories. In *CVPR '05: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 524–531, Washington, DC, USA, 2005. IEEE Computer Society. 38

[71] G. Li, Z. Ming, H. Li, and T. Chua. Spatio-temporal features for robust content-based video copy detection. In *MM '09: Proceedings of the 17th ACM International Conference on Multimedia*, pages 773–776. ACM. 30

[72] G. Li, Z. Ming, H. Li, and T. Chua. SIFT-bag kernel for video event analysis. In *MM '09: Proceedings of the 17th ACM International Conference on Multimedia*, pages 773–776. ACM, 2009. 31

[73] W.-H. Lin and A. Haputmann. Identifying news videos' ideological perspectives using emphatic patterns of visual concepts. In *MM '09: Proceedings of the 17th ACM International Conference on Multimedia*, pages 261–270, New York, NY, USA, 2009. ACM. 33

[74] H. Liu and P. Singh. ConceptNet – A practical commonsense reasoning tool-kit. *BT Technology Journal*, 22:211–226, 2004. 34

[75] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004. 27, 40, 46, 49, 84

[76] U. Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007. 63, 64, 65

[77] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval: Evaluation of Clustering*. Cambridge University Press, 2008. 76

[78] J. Matas and O. Chum. Randomized RANSAC with sequential probability ratio test. In *ICCV '05: Proceedings of IEEE International Conference on Computer Vision*, volume 2, pages 1727–1732, Los Alamitos, CA, USA, 2005. IEEE Computer Society. 23, 88

[79] K. Mikolajczyk and C. Schmid. Scale & affine invariant interest point detectors. *International Journal Computer Vision*, 60:63–86, 2004. 46

[80] G. A. Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995. 16, 34, 36, 122

[81] H.-S. Min, J. Choi, W. De Neve, and Y. M. Ro. Near-duplicate video detection using temporal patterns of semantic concepts. In *ISM '09: Proceedings of IEEE International Symposium on Multimedia*, pages 65–71. IEEE, 2009. 30

[82] C.-W. Ngo, Y.-F. Ma, and H.-J. Zhang. Video summarization and scene detection by graph modeling. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(2):296–305, 2005. 112

[83] C.-W. Ngo, W.-L. Zhao, and Y.-G. Jiang. Fast tracking of near-duplicate keyframes in broadcast domain with transitivity propagation. In *MM '06: Proceedings of the 14th Annual ACM International Conference on Multimedia*, pages 845–854, New York, NY, USA, 2006. ACM. 23, 42, 43, 44, 45, 48, 50, 52, 53, 71, 97, 98

[84] J.-M. Odobez, D. Gatica-Perez, and M. Guillemot. Spectral structuring of home videos. In *CIVR '03: Proceedings of the 2nd International Conference on Image and Video Retrieval*, pages 310–320, Berlin, Heidelberg, 2003. Springer-Verlag. ix, 24, 25, 59, 62, 75, 76, 77, 78, 79

[85] A. Ogawa, T. Takahashi, I. Ide, and H. Murase. Cross-lingual retrieval of identical news events by near-duplicate video segment detection. In *MMM '08: Proceedings of the 14th International Conference on Advances in Multimedia Modeling*, pages 287–296, Berlin, Heidelberg, 2008. Springer-Verlag. 34

[86] M. Ortega, Y. Rui, K. Chakrabarti, S. Mehrotra, and T. S. Huang. Supporting similarity queries in MARS. In *MM '97: Proceedings of the 5th ACM international conference on Multimedia*, pages 403–413, New York, NY, USA, 1997. ACM. 2

[87] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1):62–66, 1979. 116

[88] K. A. Peker and F. I. Bashir. Content-based video summarization using spectral clustering. In *VLVB '05: Proceedings of Workshop on Very Low-Bitrate Video*, 2005. 25

[89] R. A. Peters, R. Alan, P. Ii, and R. N. Strickland. Image complexity metrics for automatic target recognizers. In *Proceedings of Automatic Target Recognizer System and Technology Conference*, pages 1–17, 1990. 59

[90] W. Qi, L. Gu, H. Jiang, X.-R. Chen, and H.-J. Zhang. Integrating visual, audio and text analysis for news video. In *ICIP '00: Proceedings of IEEE International Conference on Image Processing*, pages 10–13, 2000. 15, 32

[91] M. Rahman, P. Bhattacharya, and B. Desai. Statistical similarity measures in image retrieval systems with categorization and block based partition. In *Proceedings of IEEE International Workshop on Imaging Systems and Techniques*, pages 92–97, 2005. 57

[92] Z. Rasheed and M. Shah. Detection and representation of scenes in videos. *IEEE Transactions on Multimedia*, 7(6):1097–1105, 2005. 61

[93] P. Resnik. Using information content to evaluate semantic similarity in a

taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, volume 1, pages 448–453, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc. xi, 34, 122, 123, 124

[94] J. A. Rice. *Mathematical statistic and data analysis (Third edition).* Belmont, CA: Duxbury, 2007. 61, 79, 137

[95] Y. Rui and T. S. Huang. A unified framework for video browsing and retrieval. In *Alan Bovik, Ed., Image and Video Processing Handbook*, pages 705–715. Academic Press, 2000. 61

[96] M. Sargin, Y. Yemez, E. Erzin, and A. Tekalp. Audiovisual synchronization and fusion using canonical correlation analysis. *IEEE Transactions on Multimedia*, 9(7):1396 –1403, 2007. 35

[97] S. Satoh, M. Takimoto, and J. Adachi. Scene duplicate detection from videos based on trajectories of feature points. In *MIR '07: Proceedings of the international workshop on Workshop on multimedia information retrieval*, pages 237–244, New York, NY, USA, 2007. ACM. 19

[98] S. Savarese, J. Winn, and A. Criminisi. Discriminative object class models of appearance and shape by correlatons. In *CVPR '06: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2033–2040. IEEE, 2006. 21

[99] A. Schwering. Hybrid model for semantic similarity measurement. In *ODBASE '05: Proceedings of the 4th International Conference on Ontologies, DataBases, and Applications of Semantics*, volume 3761 of *Lecture*

Notes in Computer Science, pages 1449–1465, Agia Napa, Cyprus, 2005. Springer. 34, 124

[100] H. T. Shen, B. C. Ooi, and X. Zhou. Towards effective indexing for very large video sequence database. In *SIGMOD '05: Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 730–741, New York, NY, USA, 2005. ACM. 31

[101] H. T. Shen, X. Zhou, Z. Huang, J. Shao, and X. Zhou. UQLIPS: A real-time near-duplicate video clip detection system. In *VLDB '07: Proceedings of the 33rd International Conference on Very large Data Bases*, pages 1374–1377. VLDB Endowment, 2007. 31

[102] A. F. Smeaton, W. Kraaij, and P. Over. The trec video retrieval evaluation (trecvid): A case study and status report. In *RIAO '04: Proceedings of Coupling Approaches, Coupling Media and Coupling Languages for Information Retrieval*, pages 25–37, 2004. 2

[103] J. Smith, M. Naphade, and A. Natsev. Multimedia semantic indexing using model vectors. In *ICME '03: Proceedings of IEEE International Conference on Multimedia and Expo*, volume 2, pages 445–453, 2003. 20

[104] J. R. Smith and C.-S. Li. Image classification and querying using composite region templates. *Computer Vision and Image Understanding*, 75:165–174, 1999. 20

[105] M. Smith and T. Kanade. Video skimming for quick browsing based on audio and image characterization. Technical Report CMU-CS-95-186, Computer Science Department, Pittsburgh, PA, 1995. 161

[106] C. G. M. Snoek and M. Worring. Concept-based video retrieval. *Foundations and Trends in Information Retrieval*, 4(2):215–322, 2009. 164

[107] M. Srikanth, M. Bowden, and D. Moldovan. LCC at TRECVID 2005. In *Proceedings of NIST TREC Video Retrieval Evaluation*, pages 3–6, 2005. 35, 36, 155

[108] H. Tan, C. Ngo, R. Hong, and T. Chua. Scalable detection of partial near-duplicate videos by visual-temporal consistency. In *MM '09: Proceedings of the 17th ACM International Conference on Multimedia*, pages 145–154. ACM, 2009. 3, 29

[109] C. Wolf and D. Doermann. Binarization of low quality text using a Markov random field model. In *ICPR '02: Proceedings. 16th International Conference on Pattern Recognition*, volume 3, pages 160–163, 2002. 116

[110] X. Wu, A. Hauptmann, and C. Ngo. Practical elimination of near-duplicates from web video search. In *MM '07: Proceedings of the 15th ACM International Conference on Multimedia*, pages 218–227, New York, NY, USA, 2007. ACM. 3, 26, 81

[111] X. Wu, A. G. Hauptmann, and C.-W. Ngo. Novelty detection for cross-lingual news stories with visual duplicates and speech transcripts. In *MM '07: Proceedings of the 15th ACM International Conference on Multimedia*, pages 168–177, New York, NY, USA, 2007. ACM. 27

[112] X. Wu, C.-W. Ngo, and A. G. Hauptmann. Multimodal news story clustering with pairwise visual near-duplicate constraint. *IEEE Transactions on Multimedia*, 10(2):188–199, 2008. 27, 111

[113] X. Wu, C.-W. Ngo, A. G. Hauptmann, and H.-K. Tan. Real-time near-duplicate elimination for web video search with content and context. *IEEE Transactions on Multimedia*, 11:196–207, 2009. 6, 26, 31, 81

[114] X. Wu, M. Takimoto, S. Satoh, and J. Adachi. Scene duplicate detection based on the pattern of discontinuities in feature point trajectories. In *MM '08: Proceedings of the 16th ACM International Conference on Multimedia*, pages 51–60, New York, NY, USA, 2008. ACM. 26, 29, 81

[115] X. Wu, W. L. Zhao, and C. W. Ngo. Near-duplicate keyframe retrieval with visual keywords and semantic context. In *CIVR '07: Proceedings of the 2nd International Conference on Image and Video Retrieval*, pages 162–169. ACM, 2007. 21, 22, 70, 71

[116] L. Xie, A. Natsev, and J. Testic. Dynamic multimodal fusion in video search. In *ICME '07: Proceedings of IEEE International Conference on Multimedia and Expo*, pages 1499–1502, 2007. 38

[117] D. Xu, T.-J. Cham, S. Yan, and S.-F. Chang. Near duplicate image identification with patially aligned pyramid matching. In *CVPR '08: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1–7, Los Alamitos, CA, USA, 2008. IEEE Computer Society. 17, 56, 73, 74

[118] D. Xu and S.-F. Chang. Video event recognition using kernel methods with multilevel temporal alignment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30:1985–1997, 2008. 73

[119] R. Yan and A. G. Hauptmann. Probabilistic latent query analysis for combining multiple retrieval sources. In *SIGIR '06: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 324–331, New York, NY, USA, 2006. ACM. 37

[120] R. Yan, J. Yang, and A. G. Hauptmann. Learning query-class dependent weights in automatic video retrieval. In *MM '04: Proceedings of the 12th Annual ACM International Conference on Multimedia*, pages 548–555. ACM, 2004. 37, 164

[121] E. Younessian, T. Adamek, X. Anguera, N. Oliver, and D. Marimon. Telefonica research at TRECVID 2010 content-based copy detection. In *Proceedings of the TREC Video Retrieval Evaluation (TRECVID)*, 2010. 3, 87, 94

[122] E. Younessian, T. Mitamura, and A. Hauptmann. Multimodal knowledge-based analysis in multimedia event detection. In *ICMR '12: Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*, pages 51:1–51:8, New York, NY, USA, 2012. ACM. 163

[123] E. Younessian, M. Quinn, T. Mitamura, and A. Hauptmann. Multimedia event detection using visual concept signatures. In *Proceedings of SPIE 8667, Multimedia Content and Mobile Devices*, pages 708–720. 2013. 151

[124] E. Younessian, D. Rajan, and E. S. Chng. Improved keypoint matching method for near-duplicate keyframe retrieval. In *ISM '09: Proceedings of*

IEEE International Symposium on Multimedia, pages 298–303, 2009. 56, 104

[125] D.-Q. Zhang and S.-F. Chang. Detecting image near-duplicate by stochastic attributed relational graph matching with learning. In *MM '04: Proceedings of the 12th Annual ACM International Conference on Multimedia*, pages 877–884, New York, NY, USA, 2004. ACM. viii, 17, 22, 23, 40, 41, 54, 68, 70

[126] J. Zhang, L. Sun, S. Yang, and Y. Zhong. Joint inter and intra shot modeling for spectral video shot clustering. In *ICME '05: IEEE International Conference on Multimedia and Expo*, pages 1362–1365, 2005. 24, 25

[127] W.-L. Zhao and C.-W. Ngo. Scale-rotation invariant pattern entropy for keypoint-based near-duplicate detection. *IEEE Transaction on Image Processing*, 18:412–423, 2009. 26

[128] W. L. Zhao, C. W. Ngo, H. K. Tan, and X. Wu. Near-duplicate keyframe identification with interest point matching and pattern learning. *IEEE Transactions on Multimedia*, 9(5):1037–1048, 2007. 40, 42, 56, 70, 71

[129] Y. Zheng, L. Duan, Q. Tian, and J. Jin. TV commercial classification by using multi-modal textual information. In *ICME '06, Proceedings of IEEE International Conference on Multimedia and Expo*, pages 497–500, 2006. 33

[130] Z. Zhong Lan, L. Bao, S.-I. Yu, W. Liu, and A. G. Hauptmann. Double fusion for multimedia event detection. In *MMM '12: Proceedings of the 18th International Conference on Multimedia and Modeling*, volume 7131

of *Lecture Notes in Computer Science*, pages 173–185. Springer, 2012. 39, 146, 151, 161

[131] X. Zhou and L. Chen. Monitoring near duplicates over video streams. In *MM '10: Proceedings of the 18th ACM International Conference on Multimedia*, pages 521–530, 2010. 30

[132] X. Zhou, X. Zhou, L. Chen, A. Bouguettaya, N. Xiao, and J. Taylor. An efficient near-duplicate video shot detection method using shot-based interest points. *IEEE Transactions on Multimedia*, 11(5):879–891, 2009. 26, 28, 81, 92, 104, 109

[133] J. Zhu, S. C. Hoi, M. R. Lyu, and S. Yan. Near-duplicate keyframe retrieval by nonrigid image matching. In *MM '08: Proceedings of the 16th ACM International Conference on Multimedia*, pages 41–50, New York, NY, USA, 2008. ACM. 40, 45, 50, 57, 71, 73

[134] Q. Zhu, M.-C. Yeh, and K.-T. Cheng. Multimodal fusion using learned text concepts for image categorization. In *MM '06: Proceedings of the 14th Annual ACM International Conference on Multimedia*, pages 211–220, New York, NY, USA, 2006. ACM. 38

[135] J. Zobel and T. C. Hoad. Detection of video sequences using compact signatures. *ACM Transactions on Information System*, 24:1–50, 2006. 31