We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS BOOK CITATION INDEX INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

1

On Cross-Media Correlation and Its Applications

Wei-Ta Chu
National Chung Cheng University
Taiwan

1. Introduction

Multimedia researches integrate and manipulate different media to facilitate media management, browsing, or presentation such that the developed multimedia systems provide promising learning, entertainment, or information access functionalities. The essence of multimedia is that the processed media are often correlated. Cross-media correlation not only defines the uniqueness of a multimedia system, but also provides clues for efficient/effective processing.

Cross-media correlation comes from related properties between different modalities. Maintenance of these properties is often required to facilitate appropriate presentation and to convey correct information. For example, in a news video, the anchorperson's lip motion should temporally matches with the pronounced sound, or it would be very annoying for us to realize meanings of news stories (Chen, 2001). We call this requirement temporal synchronization, and we say that the anchorperson's lip motion has temporal correlation with the pronounced sound. Another common correlation lies on spatial relationship between different modalities. For example, it's very often that a web page contains text, images, animation, or even videos. How these media arranged on a page, i.e., layout, apparently affects whether information is correctly presented. Images should not occlude the text region, for example. This type of spatial correlation is automatically tackled by web browsers and is transparent to users.

As aforementioned, cross-media correlation apparently or subtly exists in various multimedia systems and plays important roles in many aspects. This article investigates four types of cross-correlations: temporal correlation, spatial correlation, content correlation, and social correlation. We will first give a few brief examples to introduce each type of correlation, and then demonstrate a main system that elaborately exploits one or more cross-media correlations to facilitate media management.

Among four types of correlations, temporal correlation and spatial correlation have been studied for years. For example, in audiovisual display and video compression, temporal synchronization guarantees correct presentation and processing, and different visual media should be placed at the right position to avoid misunderstanding. Moreover, some

multimedia standards such as SMIL (Synchronized Multimedia Integration Language) are proposed to describe multimedia documents. In Section 2, we will describe a multimedia lecturing system that integrates tutor's speech, HTML-based lectures, and guided events to assist English learning.

One example for content correlation is the spreadsheet application. We have numerical data and the corresponding charts. If numerical values change, the corresponding chart changes accordingly. Although numerical data and charts present in different modalities, they convey the same meaning and thus have close content correlation. In Section 3, we will describe a video scene detection system that exploits content correlation between photos and videos taken in journeys. On the basis of video scenes, we propose novel photo summarization and video summarization methods, in which photo summarization is influenced by the correlated video segments, and video summarization is influenced by the characteristics of correlated photos.

Finally, we propose a new kind of cross-media correlation, i.e., social correlation, which emerges in recent years and facilitates multimedia content analysis from a new perspective. In Section 4, social relations between people are described as a network structure. We treat a movie as a small society, and explore the relationships between characters to facilitate movie video understanding.

2. Temporal Correlation and Spatial Correlation

2.1 Introduction

Among various corrleations between media, temporal and spatial correlatons have been studied for several decades. A simple example is slide presentation in Powerpoint. If there are animations to display objects, objects should be shown in correct temporal order to convey correct information. Another interesting example is automatic page turning or score following (Arzt et al., 2008). Music and text-based scores are temporally synchronized such that score pages can automatically change when the played music reaches the end of a page.

In addition to temporal order, objects should be located appropriately to prevent confusion. For objects at the same location or overlap, introduction of depth, or the so-called z-index, facilitates spatial description objects. Therefore, even Powerpoint, which is not thought as a complex multimedia application, needs careful consideration of temporal and spatial correlations between objects.

Recently, integration of various media into the same layout has been the most popular presentation. Synchronized multimedia integration language (SMIL) was proposed to enable authoring and interaction in audiovisual presentations. Through description by the markup language, we can specify media types, time information, spatial information, hyperlinks, and interactive functions for a multimedia document. Due to large-scale of applications and potential commercial benefits, many software developers propose multimedia document format to achieve vivid audiovisual presentations and enrich web-based browsing, such as Adobe's Flash and Microsoft's Silverlight.

Temporal and spatial correlations also present in one of the most important applications – video compression by MPEG and related coding standards. In these standards, audio and

video are compressed separately, and correct decoding is achieved only if audio-video synchronization can be guaranteed. Temporal synchronization is especially important in coding schemes. Moreover, in the MPEG-4 standard, object-based coding is introduced and many coding tools such as sprite coding are proposed to describe spatial relationships among different objects.

In this section, we will introduce a multimedia lecturing system, which integrates text, images, audio, and animation to teach English as a second language. Correlations between different modalities would be discovered and manipulated to facilitate effective tutoring.

2.2 Web-based Synchronized Multimedia Lecturing System

The Web-based Synchronized Multimedia Lecturing system (WSML) aims to capture teaching activities and provides web-based lecturing, which is totally developed by dynamic HTML techniques and can be easily browsed via any webpage browser (Chu & Chen, 2005). Figure 1 show the system architecture composed of three main components: the WSML recorder, WSML document servers, and the WSML browser.

Lectures in the WSML system are all stored in the form of web pages. This characteristic facilitates tightly integration of distance learning and web browsing. To produce a WSML document, the WSML recorder retrieves HTML lectures prepared in advance, and records teacher's tutoring activities, such as teacher's speech and guided events on web pages. The guided events include cursor's trajectory, web page scrolling, highlighting, pen stroke, annotation, and hyperlink. These guided events animate still web pages, and provide vivid and practical tutoring activities on web pages.

In a WSML document, teacher's speech plays as foundation of the time axis, and all other guided events are associated to it by relative timestamps. For example, a scrolling event would occur at 1 minute and 23 seconds relative to the start of teacher's speech, and a highlight event would occur at 2 minute and 45 seconds. Therefore, temporal correlation exists between guided events and teacher's speech. In addition, it's obvious that spatial information of guided media is very important. For example, a pen stroke event should be displayed at the right place in a web page only if the coordinates of pen stroke pixels are appropriately stored. Similarly, the WSML recorder captures coordinates relative to the left-top corner of a web page. Cursor trajectory, highlight event, pen stroke, annotation, and scrolling event correlate to web pages with spatial information.

As shown in Figure 1, all guided events associated with temporal and spatial information are captured by the WSML recorder. At the client side, various media and their correlations should be integrated to present correct lecturing information. The WSML browser at the client side is a web browser equipped with dynamic HTML functionalities. It synchronizes various media and implements audiovisual presentations that were invoked by teachers.

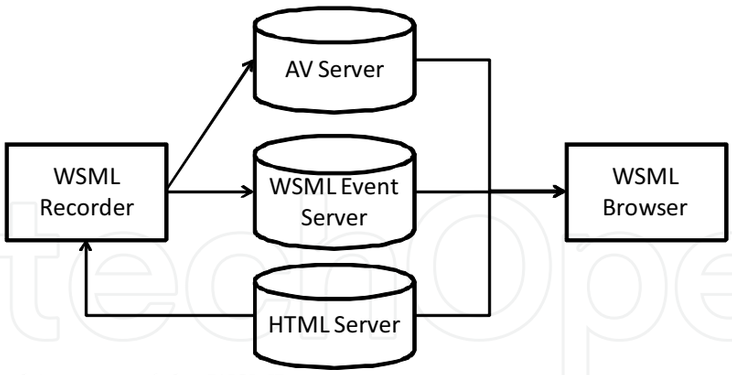

Fig. 1. System architecture of the WSML system.

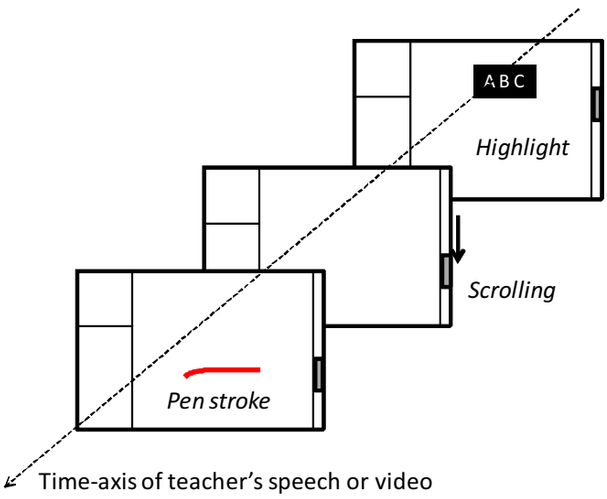

Fig. 2. Presentation scenario of the WSML browser.

Figure 2 shows the presentation scenario of the WSML browser. Along the time-axis of teacher's speech or video, various guided events specially designed for teaching English are displayed in appropriate temporal order and at appropriate positions. In this example, the highlight event is often used to indicate a new vocabulary. The scrolling event is the unique action of web-based lectures. The pen stroke event can be used to highlight a sentence, or to mark corrections on students' articles. In (Chen & Liu, 2009), the authors extend this function by visualized annotations that are commonly used by English teachers to correct students' composition homeworks. Animated visual annotations accompanied with the teacher's speech more accurately point out composition errors and show how to correct.

2.3 Explicit and Implicit Correlation

We can see various correlations exist between media in the WSML system. Nonetheless, according to whether the cross-media correlation can be easily captured or not, correlations in this system can be divided into two categories – explicit correlations and implicit correlations. Explicit correlations are able to be directly captured by the WSML recorder, such as position of the cursor, timestamp of a pen stroke, and the extent of scrolling. For example, the WSML recorder periodically stores the position of cursor, which can be

implemented by dynamic HTML programming (Goodman, 2006). Actually, all spatial and temporal correlations of guided events described above can be explicitly captured.

On the other hand, some correlations are hidden between media and cannot be directly captured by the WSML recorder. In studying English, listening to teacher's reading and comparing it with text in lecture is an important process. Teacher's speech is related to text in web pages, but this relationship cannot be explicitly recorded. Therefore, we would like to develop a speech-text alignment module to discover correlations between two modalities.

Figure 3 shows the system flowchart of the speech-text alignment module. First, we apply a speech recognition module to recognize teacher's reading into text. Note that the recognition module not only transforms speech into text, but also stores timestamp of each recognized word. Through this process, aligning speech and text has been transformed into aligning two text sequences. However, recognized text is imperfect, and many words pronounced similarly would be erroneously recognized. Therefore, we further transform recognized words and text in lectures into phonetic sequences by the CMU pronunciation dictionary (CMU, 2009). Through this process, words with similar pronunciation are encoded as the same string. For example, both *through* and *threw* are converted to "*TH R UW*", in which each term indicates a phoneme symbol.

Based on phonetic strings, we find the longest common subsequence (LCS) to determine the correspondence between recognized text and text in lectures. Due to variations of tense or plurality, words having the same meaning are not necessarily converted into exactly the same phonetic string. For example, the word *student* is converted to "*S T UW D AH N T*", while the word *students* are converted to "*S T UW D AH N T S*". Therefore, we calculate the edit distance between two phonetic strings, and claim them as matched strings if their distance is less than a threshold.

We apply a dynamic programming strategy to find the optimal LCS, which represents the best correspondence between two text sequences. The timestamps of the words matched with recognized words are then determined. For those words that are not aligned, we estimate their timestamps by interpolation or extrapolation. Finally, a fully timestamped sequence is obtained.

The proposed process discovers implicit temporal correlation between teacher's reading and text in lectures. With this correlation, a sentence can be especially highlighted when the teacher speaks it. It is very helpful for students with bad English listening comprehension.

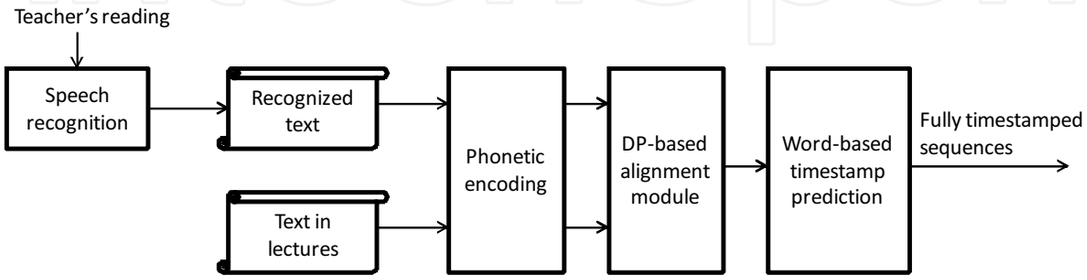

Fig. 3. System flowchart of the speech-text alignment module.

2.4 Evaluation

To evaluate the propose speech-text alignment process, one subjective experiment and one objective experiment are designed. In the subjective experiment that is similar to the investigation in (Steinmetz, 1996), we study how synchronization skews between highlighted sentences and teacher's reading affect human perception. We invited several males and females to evaluate sentence-based sync skews by giving scores from 1 to 5. Smaller score means higher degree of perceiving sync skew. The assessors were asked to see highlighted sentences and listen to corresponding speech, and then evaluate whether a sync error is perceived. In this work, the sentences with scores less than three are claimed to be out-of-sync.

Figure 4 that is by courtesy of (Chu & Chen, 2005) shows the relationships between human perception and sentence-based sync skews. The horizontal axis denotes the sync skews in milliseconds (ms), in which positive values mean sentences show after speech, and negative values mean sentences show before speech. The vertical axis denotes the percentage of assessors who are able to perceive sync errors. This figure can be divided into three regions: 1) The hardly-detected region ranges from -750 ms to +300 ms, in which few assessors can perceive sync errors. 2) The transient region ranges from +300 ms to +1000 ms and from -750 ms to -1350 ms. Synchronization skews in this region can be perceived by more than half of assessors, but they don't lead to misunderstanding. 3) The out-of-sync region spans beyond -1300 ms and +1000 ms. All assessors perceive sync errors, and this presentation largely annoys viewers and may cause misunderstanding.

From Figure 4, human perception is not symmetric when sentences are highlighted before or after the speech. Basically, humans have higher tolerance of speech-text sync skews when sentences are highlighted before start of the corresponding speech. This again confirms the non-uniformity nature of human perception.

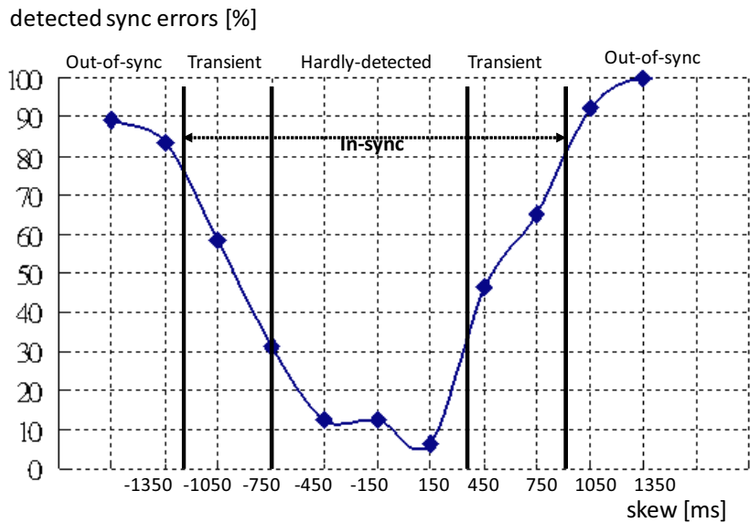

Fig. 4. Relationships between human perception and sync skews.

In the objective experiment, we measure the percentage of in-sync sentences under different speech recognition accuracy. From Figure 4, we define that sentences with sync skews in

"hardly-detected" region or in "transient" region are in-sync. We adopt SPHINX speech recognition engine (Huang, 1993) in real implementation, and achieve different recognition accuracy in different teachers' speeches and in different documents. Because the recognition engine is not specially trained by any specific teacher, the recognition accuracy is generally not high. But fortunately, the proposed speech-text alignment process is still able to determine most of the correspondence based on partially correct text. In this experiment, 80% of sentences are in-sync when the recognition accuracy is 25%, i.e., only quarter of words are correctly recognized. All sentences can be correctly synchronized with speech when the recognition accuracy is higher than 40%. In summary, we effectively discover implicit temporal correlation between speech and text and facilitate teaching English as a second language.

3. Content Correlation

3.1 Introduction

Content correlation between different media or different data is also an important area of multimedia research. A straightforward example is content-based image retrieval. Given an image containing a specific scene or an object, we would like to find images with the same scene or object from image database or from the web. Generally these images have similar appearance or convey similar semantics. Extending from this study, multimedia indexing (Snoek & Worring, 2005), video concept detection (Snoek et al, 2007; Jiang et al, 2008), and video copy detection (Law-To et al., 2007) have been widely investigated in recent years.

Another perceptive of content correlation comes from spreadsheet applications. If we change numerical values, the corresponding chart changes dynamically. Numbers and charts present the same meanings but are in different presentations. In the WSML system described in the previous section, guided events are attached to the corresponding HTML pages. If the HTML lectures are removed, all corresponding guided events should be removed, too. Tight content correlations exist between different media.

In this section, we focus on media captured in journeys and find content correlations between images and videos to facilitate efficient media management, including video scene detection and video/photo summarization. We demonstrate the assist of cross-media correlation on multimedia content analysis.

3.2 Travel Media Analysis

There are at least two challenges in travel media analysis. First, there is no clear structure in travel media. Unlike scripted videos such as news and movies, videos captured in journeys just follow the travel schedule, and content in video may consist of anything people willing or unwilling to capture. Second, because amateur photographers don't have professional skills, the captured photos and videos often suffer from bad quality. The same objects in different photos or video segments may have significant appearance. Due to these characteristics, conventional image/video analysis techniques cannot be directly applied to travel media.

People often take both digital cameras and digital camcorders in journeys. Even with only one of these devices, digital cameras have been equipped with video capturing functions,

and on the other hand, digital camcorders have the "photo mode" to facilitate taking high-resolution photos. Therefore, photos and videos in the same journey often have similar content, and the content correlation can be utilized to analyze travel media.

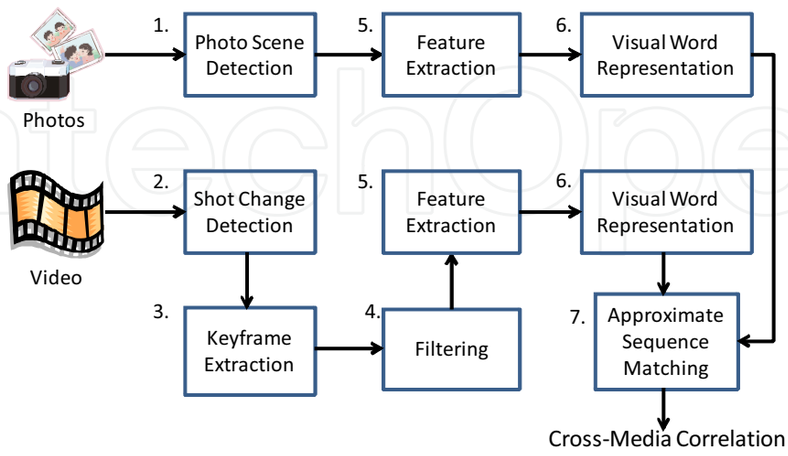

Fig. 5. Flowchart for finding cross-media correlation between photo and video.

3.3 Content Correlation Between Photos and Videos

We assume that travelers alternately take photos and videos when they visit each scenic spot. Along the travel schedule, photos and videos are captured in the same temporal order. Figure 5 shows the flowchart of finding cross-media correlation between a photo set and a video. Note that all video segments captured in the same journey are concatenated as a single video stream according to the temporal order.

- Photo Scene Detection

There are large time gaps between photos in different scenic spots because of transportation. We check time gaps between temporally adjacent photos, and claim a scene boundary exists between two photos if their time gap exceeds a dynamic threshold (Platt et al., 2003). The method proposed in (Platt et al., 2003) has been widely applied in photo clustering, and has been proven very effective. After this time-based clustering, photos taken at the same scenic spot (scene) are clustered together.

- Keyframe Extraction

For the video, we segment it into shots based on difference of HSV color histograms in consecutive video frames (Hanjalic, 2002). To efficiently represent each video shot, one or more keyframes are extracted. We adopt the method proposed in (Chasanis et al., 2007), which automatically determines the most appropriate number of keyframes based on an unsupervised global k-means algorithm (Likas et al, 2003). The global k-means algorithm is an incremental deterministic clustering algorithm that iteratively performs k-means clustering while increasing k by one at each step. The clustering process ends until the clustering results converge.

- Keyframe Filtering

Video shots with blurred content often convey less information, and would largely degrade the performance of correlation determination. To detect blurred keyframes, we check edge information in different resolutions (Tong et al, 2004). The video shots with blurred keyframes are then put aside from the following processes. After video shot filtering, fewer video shots (keyframes) are needed to be examined in the matching process. This filtering reduces influence of blurred content, which may cause false matching between keyframes and photos.

- Visual Word Representation

After the processes above, correlation between photos and videos is determined by matching photos and keyframes. Image matching is an age-old problem, and is widely conducted based on color and texture features. However, especially in travel media, the same place may have significantly different appearances, which may be caused by viewing angles, large camera motion, and overexposure/underexposure. On the other hand, landmarks or buildings with apparent structure are often important clues for image matching. Therefore, we need features that resist to luminance and viewpoint changes, and are able to effectively represent local structure.

We extract SIFT (Scale-Invariant Feature Transform) features (Lowe, 2004) from photos and keyframes. SIFT features from a set of training photos and keyframes are clustered by the k-means algorithm. Feature points belong to the same cluster are claimed to belong to the same *visual word* (Sivic & Zisserman, 2003). Before matching photos with keyframes, visual words in photos and keyframes are collected as visual word histograms. Based on this representation, the problem of matching two image sequences has been transformed into matching two sequences of visual word histograms.

Conceptually, each SIFT feature point represents texture information around a small image patch. After clustering, a visual word presents a concept, which may correspond to corner of building, tip of leaves, and so on. A visual word histogram presents what concepts compose an image. To discover cross-media correlation, we would like to find photos and keyframes that have similar concepts.

- Approximate Sequence Matching

To find the optimal matching between two sequences, we exploit the dynamic programming strategy to find the longest common subsequence (LCS) between them. Given two visual word histogram sequences, $X = \langle \boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_m \rangle$ and $Y = \langle \boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_n \rangle$, which correspond to photos and keyframes, respectively. Each item in these sequences is a visual word histogram, i.e., $\boldsymbol{x}_i = h[j]$, $0 \leq j \leq N-1$, where N is the number of visual words. The LCS between two subsequences X_m and Y_n is described as follows.

$$LCS(X_m, Y_n) = \begin{cases} LCS(X_{m-1}, Y_{n-1}) + 1, & \text{if } x_m = y_n, \\ \max(LCS(X_{m-1}, Y_n), LCS(X_m, Y_{n-1})), & \text{otherwise,} \end{cases} \tag{1}$$

where X_i denotes the ith prefix of X, i.e., $X_i = \langle \boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_i \rangle$, and $LCS(X_i, Y_j)$ denotes the length of the LCS between X_i and Y_j. This recursive structure facilitates usage of the dynamic programming approach.

Based on visual word histograms, the equality in eqn. (1) occurs when the following criterion is met:

$$x_i = y_j \quad \text{if } \sum_{k=0}^{N-1} |(h_i(k) - h_j(k))| < \delta, \tag{2}$$

where h_i and h_j are the visual word histograms corresponding to the images x_i and y_j. According to this measurement, if visual word distributions are similar between a keyframe and a photo, we claim that they are conceptually "common" and contain similar content.

3.4 Video Scene Detection

Figure 6 shows an illustrated example of video scene detection based on cross-media correlation. The double arrows indicate the determined correlations between photos and keyframes. If a video shot's keyframe matches the photo in the *i*th photo scene, this video shot is assigned as in *i*th video scene as well. For those video shots without any keyframe matched with photos, we apply interpolation and nearest neighbor processing to assign them (Chu et al., 2009).

Because photo scene detection is much easier than video scene detection, this method first solves an easier problem, finds the correlation between two problems, and then solves the harder problem by consulting with cross-media correlation. This idea brings a new perspective to conduct travel media analysis.

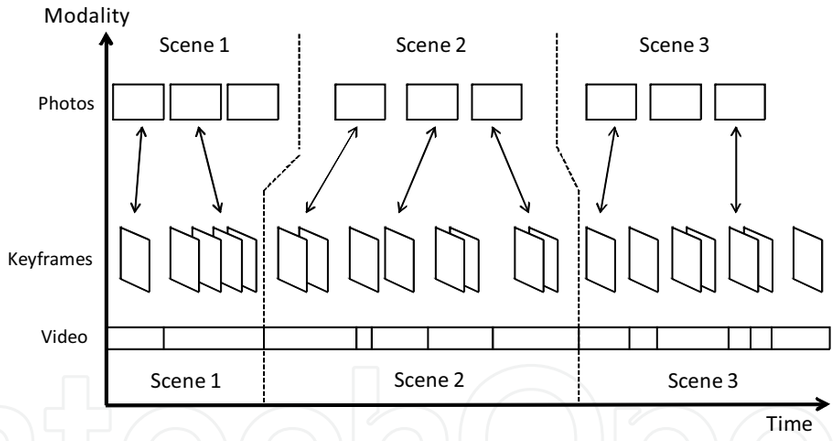

Fig. 6. Illustration of video scene detection using cross-media correlation.

3.5 Photo Summarization and Video Summarization

Correlation determined by the previous process suffices for scene boundary detection. However, to generate summaries, finer cross-media correlation is needed to define importance value of each photo and each keyframe. We call the correlation described above *global cross-media correlation*, which describes matching in terms of visual concepts. In this subsection, we need further analyze *local cross-media correlation* to find matching in terms of objects.

For photos and keyframes in the same scene, we perform finer matching between them by the SIFT matching algorithm (Lowe, 2004). For the photo p_m and the keyframe k_n, we claim

they contain the same object if the number of matched feature points exceeds a predefined threshold τ. This threshold can be adjusted dynamically according to the requirements of users. Note that local cross-media correlation is determined based on SIFT features rather than visual word histograms. Visual word histograms describe global distribution of concepts (visual words), while feature points conveying local characteristics more appropriately describe whether two images have the same building or objects.

- Photo Summarization

The idea of defining each photo's importance comes from two perspectives. First, when a view or an object is both captured in photos and videos, the captured content must attract people more and is likely to be selected into summaries. The second factor is involved with considering characteristics of correlated videos to define photo's importance. When people take a closeup shot on an object, this object must attract people more. Therefore, a photo's importance is set higher if it matches with a keyframe that is between a zoom in action and a zoom out action, or is between a zoom in action and camera turning off. Two factors defining importance values can be mathematically expressed as follows.

Factor 1:
The first importance value of the photo p_m is defined as

$$PT_{1,m} = \frac{I_{1,m}}{\max_{i=1,2,\ldots,M} I_{1,i}}, \tag{3}$$

where M is the number of photos in this dataset. The value $I_{1,m}$ is calculated as

$$I_{1,m} = \begin{cases} L_1(h_{p_m}, h_{k_n}), & \text{if the photo } p_m \text{ matches with the keyframe } k_n, \\ 0, & \text{otherwise,} \end{cases}$$

where h_{p_m} and h_{k_n} are visual word histograms of the photo p_m and the keyframe k_n, respectively. The value $L_1(\cdot,\cdot)$ denotes L_1-distance between two histograms.

Factor 2:
The second importance value of the photo p_m is defined as

$$PT_{2,m} = \frac{I_{1,m} \times ZoomIn(p_m)}{\max_{i=1,2,\ldots,M} I_{1,i} \times ZoomIn(p_i)}, \tag{4}$$

where $ZoomIn(p_m) = 1$ if the keyframe k_n that matches with p_m locates between a zoom in action and a zoom out action, or between a zoom in action and camera turning off. The value $ZoomIn(p_m) = 0$, otherwise.

Two importance values are normalized and integrated to form the final importance value:

$$PT_m = \alpha \times PT_{1,m} + \beta \times PT_{2,m}. \tag{5}$$

Currently, the values α and β are set as 1.

Users can set the desired number of photos in summaries. To ensure that the generated summary contains photos of all scenes (scenic spots), we first pick the most important photo in each scene to the summary. After that, we sort photos according to their corresponding importance values in descending order, and pick photos sequentially until the desired number is achieved.

According to the definitions above, only photos that are matched with keyframes have importance values larger than zero. If all photos with importance values larger than zero are picked but the desired number hasn't achieved, we define the importance value of a photo p_i not picked yet by calculating the similarity between p_i and its temporally closest photo p_j that has nonzero importance value, i.e.,

$$T_i = L_1(h_{p_i}, h_{p_j}). \tag{6}$$

We sort the remaining photos according to the alternative importance values in descending order, and pick photos sequentially until the desired number is achieved.

- Video Summarization

Similar to photo summarization, we advocate that photo taking characteristics in a scene affect selection of important video segments. Two factors are also designed. The first factor is the same as that in photo summarization, i.e., video shots whose content also appears in photos are more important. Moreover, a video shot in which more keyframes match with photos is relatively more important. Two factors can be mathematically expressed as follows.

Factor 1:
The first importance value of a keyframe k_m is defined as

$$KT_{1,m} = \frac{I_{1,m}}{\max_{i=1,2,\ldots,M} I_{1,i}}, \tag{7}$$

where *M* is the number of keyframes in this dataset. The value $I_{1,m}$ is calculated as

$$I_{1,m} = \begin{cases} L_1(h_{k_m}, h_{p_n}), & \text{if the keyframe } k_m \text{ matches with the photo } p_n, \\ 0, & \text{otherwise,} \end{cases}$$

where h_{k_m} and h_{p_n} are visual word histograms of keyframe k_m and the photo p_n, respectively.

Factor 2:
The second importance value of the keyframe k_m is defined as

$$KT_{2,m} = \frac{I_{2,m}}{\max_{i=1,2,\ldots,M} I_{2,i}}, \tag{8}$$

where the value $I_{2,m}$ is defined as

$$I_{2,m} = \sum_{j=1}^{J} L_1(h_{k_j}, h_{p_{j^*}}). \tag{9}$$

This expression means there are J keyframes in the shot containing k_m, and the notation p_{j^*} denotes the photo matched with the keyframe k_j.

These two importance values are integrated to form the final importance value of k_m:

$$KT_m = \alpha \times KT_{1,m} + \beta \times KT_{2,m}. \tag{10}$$

Users can set the desired length of video summaries. To ensure that the generated summary contains video segments of all scenes (scenic spots), we first pick the most important keyframe of each scene. Assume that the keyframe k_i is selected, we determine length and location of the video segment S_i corresponding to k_i as

$$S_i = \left(\frac{t(k_{i-1}) + t(k_i)}{2}, \frac{t(k_i) + t(k_{i+1})}{2} \right), \tag{11}$$

where $t(k_i)$ denotes the timestamp of the keyframe k_i, and k_{i-1} and k_{i+1} are two nearest keyframes that are before and after k_i, and with nonzero importance values. Two values in the parentheses respectively denote start time and end time of the segment S_i.

We pick keyframes and their corresponding video segments according to keyframe's importance values until the desired length is achieved. If all keyframes with nonzero importance values are picked but the desired length hasn't achieved, we utilize a method similar to that in eqn. (6) to define remaining keyframes' importance values, and pick appropriate number of keyframe accordingly.

3.6 Evaluation

We evaluate the proposed methods based on seven data sets. Each dataset includes a video clip and a set of photos, and there are totally 85 minutes of videos with 1084 keyframes and 423 photos. Photos are rescaled to smaller sizes due to the efficiency of feature points processing and visual word construction. Videos and photos are captured by different amateur photographers, with different capturing devices.

To measure video scene detection, we calculate purity value (Vinciarelli & Favre, 2007) by comparing detected video scenes and the ground truths. A purity value ranges from 0 to 1, and a larger purity value means that the detection result is closer to the ground truth. We averagely obtain a purity value of 0.95, which is very promising in scene detection, especially in the uncontrolled travel media.

To further show that the proposed method is more appropriate to travel videos, we compare it with the method proposed in (Chasanis et al, 2007). One of the major challenges in scene detection is the over-segmentation problem. We measure this effect in two methods and list the results in Table 1. In each cell of this table, the value (*m*, *n*) denotes that a scene is segmented into *m* and *n* scenes, by the method in (Chasanis et al, 2007) and our method, respectively. For example, in the S2 column for Video 1, (4,1) means that the second scene in Video 1 is segmented into four scenes and one scene by the method in (Chasanis et al, 2007) and our approach, respectively. There are totally 6 scenes in Video 1, but it is segmented into 27 and 8 scenes by two methods. We see the effect of over-segmentation is severe in Chasanis's approach, and our approach works much better from this perspective.

To objectively demonstrate summarization results, we ask content owners to manually select subset of keyframes and photos as summarization ground truth. In generating video summaries or photo summaries, we set the number of keyframes or photos in manual summaries as the targeted number to be achieved. Summarization results are measured by precision values, i.e.,

$$\text{Precision} = \frac{\#\text{ correctly selected keyframes}}{\#\text{ selected keyframes}} \text{ and } \text{Precision} = \frac{\#\text{ correctly selected photos}}{\#\text{ selected photos}}. \tag{12}$$

Note that precision and recall rates are the same due to the selection policy.

	S1	S2	S3	S4	S5	S6	Overall
Video 1	(1,1)	(4,1)	(7,2)	(3,1)	(9,2)	(3,1)	(27,8)
Video 2	(6,1)	(3,1)	(1,1)	(1,1)			(11,4)
Video 3	(1,1)	(1,1)	(1,1)	(3,1)	(2,1)		(8,5)
Video 4	(1,1)	(2,2)	(1,1)	(5,2)	(1,1)		(10,7)
Video 5	(4,2)	(2,2)	(3,1)				(9,5)
Video 6	(3,1)	(2,1)					(5,2)
Video 7	(5,1)	(2,2)	(2,2)	(1,1)	(3,2)	(1,1)	(14,9)

Table 1. Over-segmentation situations in different videos.

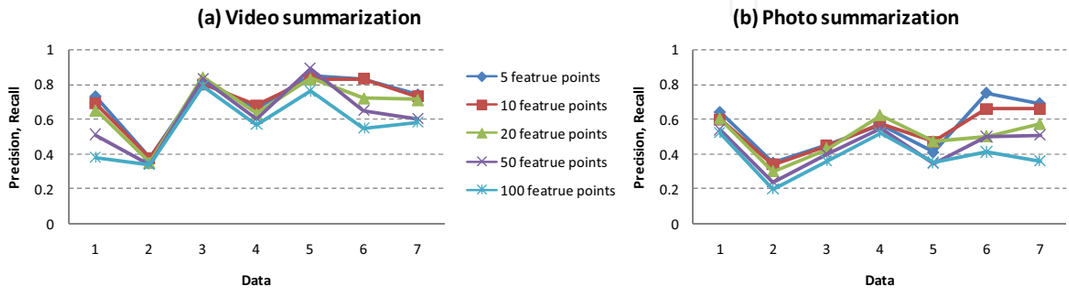

Fig. 7. Performance of (a) video summarization and (b) photo summarization.

Figure 7(a) shows precision/recall rates of video summarization under different matching thresholds τ, while Figure 7(b) shows precision/recall rates of photo summarization. Generally, we see that using five or ten matched points as the threshold we can obtain better summarization results, i.e., looser thresholds draw slightly better performance. Summarization performance of the second dataset is especially worse for both video and photos. This reason is that photos and video in this dataset have less similar content, thus content correlation between them is weak. Except for the second dataset, the proposed methods overall achieves 80% of accuracy in video summarization and 60% of accuracy in photo summarization.

4. Social correlation

4.1 Introduction

In addition to the aforementioned correlations, we introduce a new type of cross-media correlation that hasn't been studied widely before. We can view objects or humans as instances of a modality. The appearance of this modality would be presented by sound, such as anchorperson and guest in radio broadcast programs, or would be presented by visual content, such as actors' faces in TV programs. How objects or humans interact embeds semantics, which facilitates multimedia content management and retrieval.

One of the earliest works about investigating social relationships between humans in multimedia content analysis was proposed in (Vinciarelli, 2007). For radio broadcastings, this work segments audio recordings into segments of different speakers, and utilizes characteristics of duration and interaction between speakers to recognize speaker's role as

an anchorperson or a guest. Correlation between roles is conveyed by speaking duration and order. Another interesting work was proposed in (Rienks et al., 2006). By using features of speech behaviour, interaction, and topics in group meeting videos, they analyze the influence of each participant. With audiovisual features such as speaking length and motion activity, the work in (Hung et al., 2007) estimates the most dominant person in a group meeting. Also for meeting recordings, the work in (Garg et al., 2008) performs role recognition based on lexical information and social network analysis.

In this section, we exploit co-occurrence information of actors in movies to approach movie understanding. We treat a movie as a small society, and model social correlation between actors by a network called *RoleNet* (Weng et al., 2009). Based on this network, the leading roles are automatically determined and community structure of actors is discovered as well. With this information, we are able to develop an accurate story segmentation system.

4.2 RoleNet

A RoleNet is a weighted graph expressed by $G = \langle V, E, W \rangle$, where $V = \{v_1, v_2, ..., v_n\}$ represents a set of roles in a movie, $E = \{e_{ij} | \text{ if } v_i \text{ and } v_j \text{ have relationship}\}$, and the element w_{ij} in W represents strength of the relationship between v_i and v_j. Relationship between roles is developed when they interact with each other. More often two roles appear in the same scenes, more chances they can interact, and closer relationship is built.

For a movie that consists of m scenes and n different roles, we can express the status of occurrence by a matrix $A = [a_{ij}]_{m \times n}$, where the element

$$a_{ij} = \begin{cases} 1, & \text{if the } j\text{th role appears in the } i\text{th scene,} \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

The *j*th column vector, $\boldsymbol{a}_j = (a_{1j}, a_{2j}, ..., a_{mj})$, of A denotes the scenes where the *j*th role appeared. The co-occurrence of the *i*th role and the *j*th role is identified by

$$w_{ij} = \sum_{k=1}^{m} a_{ki} a_{kj} = \boldsymbol{a}_i^T \boldsymbol{a}_j, \text{ for } i \neq j. \tag{14}$$

The value of w_{ij} is actually the inner product of $\boldsymbol{a}_i$ and $\boldsymbol{a}_j$. This measurement can be generalized to the whole matrix. The co-occurrence status between roles can be expressed by $W_{n \times n} = A^T A$, in which w_{ij} is especially set as 0 when $i = j$.

The left side of Figure 8 shows the RoleNet of the movie "You've Got Mail," in which thickness of an edge represents weights on it. Relationships between roles seem to be complicated, and the goal of this work is to find the leading roles and associated communities.

4.3 Social-Based Analysis

- Leading Role Determination

There is a large gap between the impact of leading roles and that of supporting roles. Based on this observation, the problem of leading role determination can be mathematically expressed as follows.

$$\Gamma^* = \arg\max_{\Gamma}(\min\Theta_1 - \max\Theta_0), \tag{15}$$

where $\Theta_1 = \{c_i|\ell_i = 1\}$ and $\Theta_0 = \{c_i|\ell_i = 0\}$.
$\Gamma = \{\ell_i, i = 1, 2, ..., n\}$

$$\begin{cases} \ell_i = 1 & \text{if the } i\text{th role is assigned as a leading role,} \\ \ell_i = 0 & \text{otherwise,} \end{cases}$$

where n is the number of roles, Γ is a set of binary labels representing which roles are assigned as leading roles. The value c_i is weighted degree centrality, which defines the importance value of the node i, and is calculated by

$$c_i = \sum_{j \neq i} w_{ij}. \tag{16}$$

The set Θ_1 represents centrality values of the roles assigned to leading roles. The physical meaning of $(\min\Theta_1 - \max\Theta_0)$ is the difference of centrality values between the least important leading role and the most important supporting role. The result Γ^* we want is the labels that cause the largest centrality difference.

Taking "You've Got Mail" as an example, we calculate the centrality value of each role, and then sort the centrality values in descending order, as shown in Figure 8(a) and Figure 8(b). Next, the differences of centrality values between two adjacent roles in Figure 8(b) are calculated. Figure 8(c) shows the centrality difference distribution, in which each point represents a boundary between two roles. Finally, we find the maximum point in the difference distribution, which represents the largest gap in centrality. In this example, the difference between the role no. 1 and the role no. 7 is maximal, and therefore the role no. 2 and role no. 1 are claimed as leading roles.

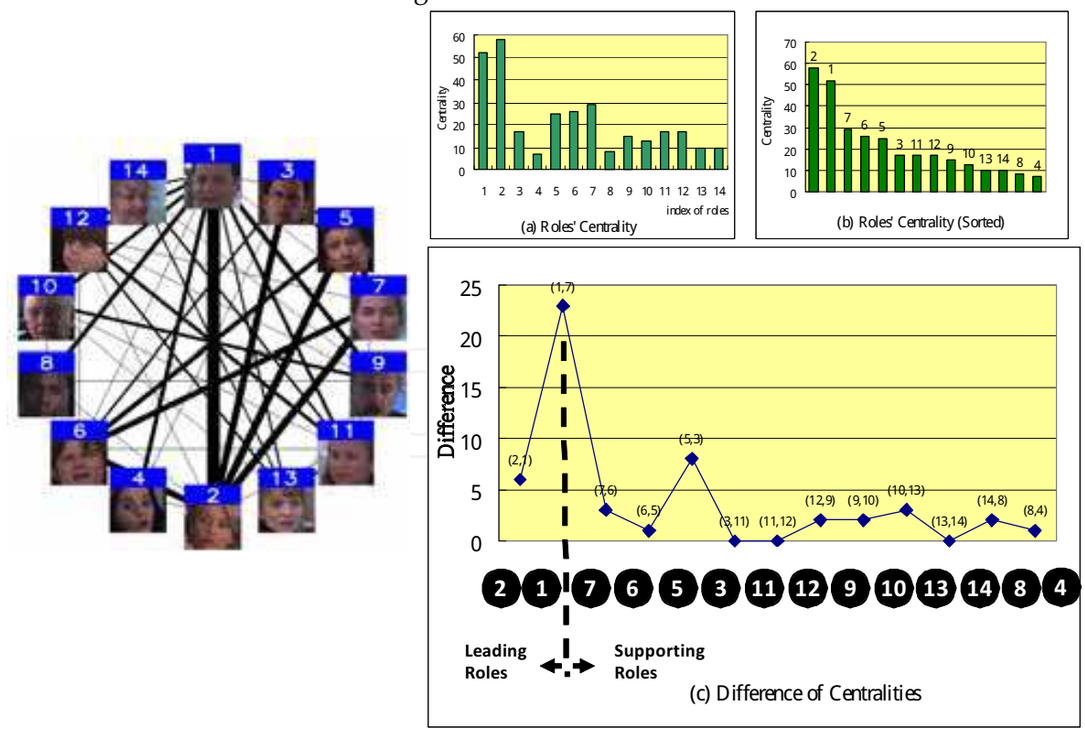

Fig. 8. Left: the RoleNet of "You've Got Mail"; Right: The process of leading role determination.

- Community Identification

After determining the leading roles, we devise a method to appropriately group certain roles into communities. Because leading roles may pass through several communities, it's not reasonable to assign them into any community. Therefore, we first remove leading roles and the edges linked to them from the RoleNet. Then, the following iterative algorithm is applied to the modified RoleNet. We use the value t to index the community's evolution situation. The value t is initialized as 0 in the beginning and increases by one when the community situation changes.

Algorithm 1: Community Identification

1. Initialize every individual node as a community. The set of community is denoted as $\Pi_t = \{T_1^t, T_2^t, ..., T_n^t\}$, t = 0, if there are initially n individual nodes. The size of the pth community in Π_t is denotes as $|T_p^t|$, which is the number of nodes included in this community.
2. From the modified RoleNet, find the edge that has the largest weight, say the edge e_{ij} between the node v_i and the node v_j, $v_i \in T_p^t$ and $v_j \in T_q^t$, then
   - If $|T_p^t| \geq 1$ and $|T_q^t| = 1$, then $T_p^{t+1} = T_p^t \cup T_q^t$, $\Pi_{t+1} = \Pi_t - \{T_q^t\}$, and t = t+1.
   - If $|T_p^t| > 1$ and $|T_q^t| > 1$, then keep current community situation.
3. Remove the edge e_{ij} from the modified RoleNet and go to Step 2 until all edges have been removed.

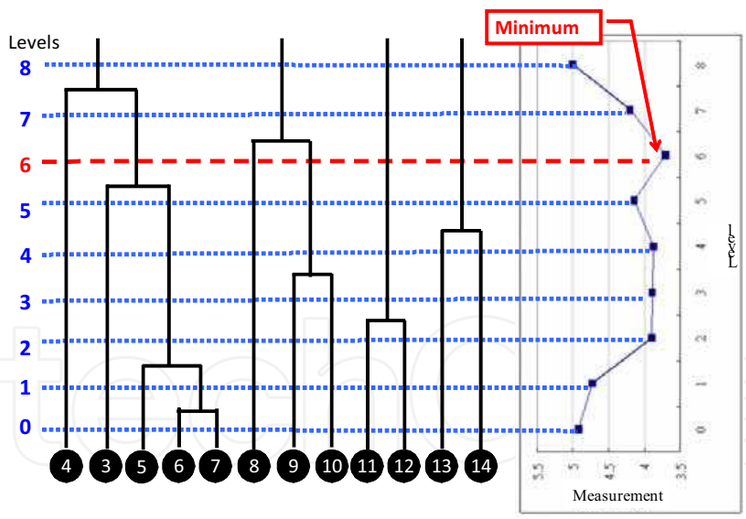

Fig. 9. Dendrogram of the clustering process.

The progress of this algorithm can be illustrated as a dendrogram, which describes how we cluster communities at each step. As shown in Figure 9, the roles no. 6 and no. 7 are first categorized together (t=1), then the role no. 5 is merged into this community at the second level (t=2). (We say "level" but not "iteration" because the community situation may not change at each iteration.) The same process can be iteratively applied until all nodes have been examined.

Each level in the dendrogram represents a case of community situation. Now the problem is to determine which level in the dendrogram is the best. We design a measurement to evaluate communities at different levels. For the level *t*, the measurement is defined as

$$AvgW_t = \frac{\sum w_{ij}}{||\Pi_t||}, \forall v_i \in T_p^t, v_j \in T_q^t, p \neq q, \tag{17}$$

where Π_t denotes the community situation at the level *t*, and $||\Pi_t||$ denotes the number of communities in this case. The value $AvgW_t$ represents the average weight between different communities at level *t*. The right part of Figure 9 shows the measures at different levels.

In community identification, we prefer that roles in different communities are least related. Therefore, we pick the community case that causes the minimal $AvgW$. In Figure 9, the minimal $AvgW$ value occurs at the sixth level, in which six communities are found to include roles {4}, {3, 5, 6, 7}, {8}, {9, 10}, {11, 12}, and {13, 14}, respectively.

To our knowledge, there is no prior study to conduct movie understanding from social perspective. How characters interact affect us to understand a movie. By analyzing social correlation between roles, we not only can determine characters with high impacts, but also determine construct community structure. In the next subsection, we further demonstrate using social correlation to facilitate story segmentation.

4.4 Story Segmentation

- Scene Representation

We first define the representation of scenes. The major difference between the proposed representation and conventional ones is that we describe scenes by "the context of roles" rather than audiovisual features. Story segmentation is achieved by comparing the role's context in successive scenes.

Let $r(k)$ denote the identification of the kth character in a specific scene. The relationship between this role and others can be expressed by a "profile vector" $\boldsymbol{w}_{r(k)} = (w_{1r(k)}, w_{2r(k)}, ..., w_{nr(k)})$, which is the $r(k)$-th column vector of the matrix W. The vector $\boldsymbol{w}_{r(k)}$ denotes the closeness between the role no. $r(k)$ and others. It is normalized into a unit vector. For the *i*th scene, we collect the profile vectors of the roles appearing in this scene to form a matrix $CM_i = [\boldsymbol{w}_{r(1)}\boldsymbol{w}_{r(2)}...\boldsymbol{w}_{r(p)}]$. It is an *n* by *p* matrix if there are *p* roles in this scene and there are totally *n* roles in the movie. Similarly, the matrix CM_j for the *j*th scene is denoted by $CM_j = [\boldsymbol{w}_{r(1)}\boldsymbol{w}_{r(2)}...\boldsymbol{w}_{r(q)}]$, if there are *q* roles in this scene. Note that the vector $\boldsymbol{w}_{r(k)}$ in the *i*th scene and the vector $\boldsymbol{w}_{r(k)}$ in the *j*th scene would be different, i.e., the identifications of the *k*th characters in these two scenes are different. Using the notations of $\boldsymbol{w}^i_{r(k)}$ and $\boldsymbol{w}^j_{r(k)}$ is more precise but dazzles readers. Therefore, we use the simplified notation in the following description.

The context-based similarity between "the *s*th role in the *i*th scene" and "the *t*th role in the *j*th scene" is defined as the inner product of two corresponding profile vectors: $\boldsymbol{w}_{r(s)} \cdot \boldsymbol{w}_{r(t)} = \boldsymbol{w}^T_{r(s)}\boldsymbol{w}_{r(t)}$. By calculating the context-based similarities between every two roles in two successive scenes, the similarity between the *i*th scene and the *j*th scene can be

expressed in a matrix form: $CM_{ij} = CM_i^T CM_j$. Finally, the context-based difference between the *i*th scene and the *j*th scene is defined as

$$d_{ij} = 1 - \frac{1}{pq} \sum_{s=1}^{p} \sum_{t=1}^{q} CM_{ij}(s,t), \tag{18}$$

which represents the average difference between pairs of roles in two different scenes. The value is between 0 and 1.

- Story Segmentation

Going through the whole movie, we can plot a "difference curve" that represents difference between adjacent scenes. Based on this curve, the goal of story segmentation is to find appropriate scene boundaries that represent changes of stories. We propose a story segmentation method called "storyshed," which is motivated by the watershed algorithm for image segmentation but is modified to meet the need of this task.

Denote the set of scene boundaries as $B = \{b_1, b_2, ..., b_{N-1}\}$, in which the element b_i denotes the boundary between the *i*th and the (*i*+1)-th scenes, and N is the total number of scenes. The proposed storyshed algorithm is to determine whether a scene boundary is a story boundary, based on the context-based difference between adjacent scenes. The context-based difference corresponding to b_i is denoted by d_i. In the first step of the segmentation process, we first find the valleys and peaks from the difference curve by checking d_i. That is,

$$\begin{cases} b_i \in Y, & \text{if } d_i < d_{i-\alpha_1} \text{ and } d_i < d_{i+\alpha_2}, \\ b_i \in P, & \text{if } d_i > d_{i-\alpha_1} \text{ and } d_i > d_{i+\alpha_2}, \\ b_i \in OT, & \text{otherwise,} \end{cases} \tag{19}$$

$\alpha_1 = \min\{j | j \in A_1\}, \quad A_1 = \{k | (d_i - d_{i-k}) \neq 0, 1 \leq k \leq i-1\},$
$\alpha_2 = \min\{j | j \in A_2\}, \quad A_2 = \{k | (d_i - d_{i+k}) \neq 0, 1 \leq k \leq (N-1-i)\},$
where $2 \leq i \leq N-2$, Y denotes the set of scene boundaries in valleys, P denotes the set of boundaries in peaks, and the set OT includes all other boundaries. Thus, $Y \cup P \cup OT = B$.

Initialize an empty set SB that will store the story boundaries. For each valley y_j in Y, find the nearest peaks to it. Let's denote the left peak of y_j as p_1 and the right peak of y_j as p_2, $p_1 \in P$ and $p_2 \in P$. Fill water into this valley until the height of the horizontal just floods p_1 or p_2. Without loss of generality, assume that p_1 is flooded first and the height of p_1 is H. Pick the scene boundaries b_k between the peaks p_1 and p_2, for which the corresponding context-based difference d_k is no less than H. Therefore, the set of story boundaries would be $SB = SB \cup \{b_k\}$.

The results reveal the boundaries that scenes around them have significantly different context information. However, the storyshed algorithm only considers local characteristics and may miss the ones that are indeed story boundaries. To consider the global characteristics, we take average of the context-based difference values of all peaks as a global threshold. If the difference value corresponding to a scene boundary is larger than this threshold, it's viewed as a story boundary. The global threshold is adaptively calculated according to the social relationships between roles in different movies.

Figure 10 shows an example of storyshed segmentation with the global threshold. Two valleys (solid black circles) in this example are at b_{t+2} and b_{t+6}. The nearest peaks to b_{t+2} are at b_{t+1} and b_{t+4}, and that to b_{t+6} are at b_{t+4} and b_{t+7}. With the global threshold, the boundary b_{t+5} is

detected as a story boundary as well. Processing with this global threshold is the same as applying a constraint so that height of water is limited to be lower than the threshold.

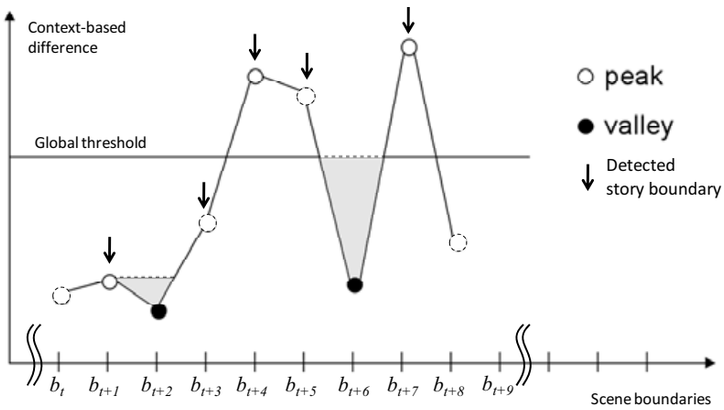

Fig. 10. An example of the storyshed segmentation method with a global threshold.

4.5 Evaluation

We use ten Hollywood movies and three TV shows to evaluate the proposed methods. Total length of evaluation data is over 21 hours, and 428 story segments are included. For each movie, we asked three persons to manually label leading roles. The ones that were labeled as the leading ones by all the three persons are treated as the ground truth. Table 2 shows performance of leading role determination. The numbers in each cell denote the indices of roles. Because the trend of appearance of leading roles is often apparent, the proposed method for leading role determination achieves very promising performance.

To evaluate story segmentation, the ground truths of story boundaries were also decided manually. We invited several subjects who were not familiar the goal and process of our work and knew nothing about the chapter information in advance. They were asked to examine every scene change boundary and decide whether it's a story boundary. We measure story segmentation performance in terms of purity value again (Vinciarelli & Favre, 2007). In addition, to show the superiority of utilizing social correlation in story segmentation, we compare our approach with a conventional tempo-based approach (Chen et al., 2004).

MovieID	M1	M2	M3	M4	M5	M6	M7
GT	1	1, 2	1, 2, 6	1, 2	1	1	1
Det.	1	1, 2	1, 2, 6	1, 2	1	1	1
MovieID	M8	M9	M10	S1	S2	S3	
GT	1	1	1	1	1, 2, 4, 5, 6, 7	1, 2, 3, 4	
Det.	1	1	1	1	1, 2, 3, 4, 5, 6, 7, 9	1, 2, 3	

Table 2. Performance of leading role determination. GT: Ground Truth; Det.: Determined resuls.

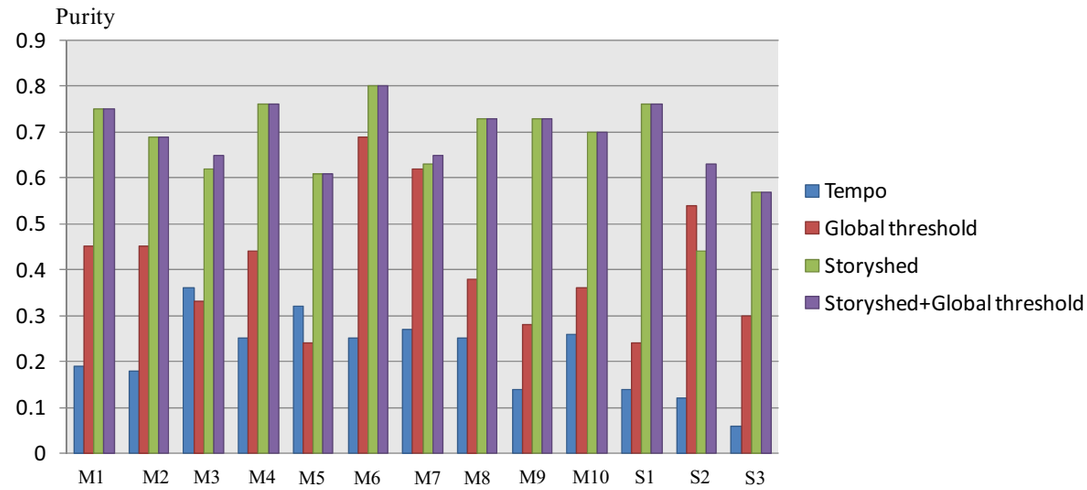

Fig. 11. Performance of story segmentation.

Figure 11 shows that the social-based approach works much better than the tempo-based one. Although the performance improvement varies in different movies, the storyshed algorithm has significantly better performance than thresholding. After combining storyshed with thresholding, the result is even slightly better than the storyshed algorithm, especially in M3, M7, and S2.

5. Conclusion

This article presents cross-media correlation and its applications in various multimedia systems. Four perspectives of correlations are investigated: temporal correlation, spatial correlation, content correlation, and social correlation. Regarding temporal and spatial correlations, we introduce a web-based multimedia lecturing system, in which various guided events have tight temporal correlation to teacher's speech, and have spatial correlation to HTML-based lectures. In addition, we design a speech-text alignment module to discover hidden correlation between teacher's reading and text in lectures. Elaborate usage of temporal and spatial correlation facilitates vivid multimedia lecturing for teaching English as a second language.

Regarding content correlation, a travel media analysis system is introduced to find correlation between photos and videos. The essential idea of this work is to solve a harder problem (video scene detection) by first solving an easier problem (photo scene detection) and then consulting with the correlation between two modalities. Besides, we argue that characteristics of photo taking can be exploited to conduct video summarization, and vice versa. We show that appropriately utilizing cross-media correlation facilitates effective multimedia content analysis.

Regarding social correlation, we analyze relationships between characters to determine leading roles and community structure. Community information makes us approach movie understanding and efficiently movie video management/retrieval. Evolution of social relationships is also studied to conduct story segmentation. We have demonstrated this

approach more matches human's cognition and works better than conventional content-based segmentation methods.

Benefits of employing cross-media correlation in multimedia content analysis are evident. Although this article describes three practical applications, correlations are often specially extracted according to characteristics of different environments or applications. Systematic description and unified framework for cross-media correlation are needed to extend its applicable domain. Finally, this article introduces the new social correlation in multimedia content analysis. We wish to draw more attention on this issue; especially community websites and online video sharing have explosively grown in recent years.

Acknowledgement
This work was partially supported by the National Science Council of the Republic of China under grants NSC 98-2221-E-194-056 and NSC 97-2221-E-194-050.

6. References

Arzt, A.; Widmer, G. & Dixon, S. (2008). Automatic Page Turning for Musicians via Real-Time Machine Listening. *Proceedings of European Conference on Artificial Intelligence.*

Chasanis, V.; Likas, A. & Galatsanos, N. (2007). Scene Detection in Videos Using Shot Clustering and Symbolic Sequence Segmentation. *Proceedings of IEEE International Conference on Multimedia Signal Processing*, pp. 187-190.

Chen, T. (2001). Audiovisual Speech Processing. *IEEE Signal Processing Magazine*, Vol. 18, No. 1, pp. 9-21.

Chen, H.-W.; Kuo, J.-H.; Chu, W.-T. & Wu, J.-L. Action Movies Segmentation and Summarization Based on Tempo Analysis. *Proceedings of the ACM SIGMM International Workshop on Multimedia Information Retrieval*, pp. 251-258.

Chen, H.-Y. & Liu, K.-Y. (2009). WMA : A Marking-Based Synchronized Multimedia Tutoring System for English Composition Studies. *IEEE Transactions on Multimedia*, Vol. 11, No. 2, pp. 324-332.

Chu, W.-T. & Chen, H.-Y. (2005). Towards Better Retrieval and Presentation by Exploring Cross-Media Correlations. *ACM Multimedia Systems Journal*, Vol. 10, No. 3, pp. 183-198.

Chu, W.-T.; Lin, C.-C. & Yu, J.-Y. (2009). Using Cross-Media Correlation for Scene Detection in Travel Videos. *Proceedings of ACM International Conference on Image and Video Retrieval*.

Carnegie Mellon University. (2009). CMU Pronouncing Dictionary. http://www.speech.cs.cmu.edu/cgi-bin/cmudict

Garg, N.P.; Favre, S.; Salamin, H.; Hakkani Tur, D. & Vinciarelli, A. (2008). Role Recognition for Meeting Participants: An Approach Based on Lexical Information and Social Network Analysis. *Proceedings of ACM Multimedia Conference*, pp. 693-696.

Goodman, D. (2006). *Dynamic HTML : The Definitive Reference*. O'Reilly Media, Inc.

Hanjalic, A. (2002). Shot-Boundary Detection : Unraveled or Resolved? *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 12, No. 2, pp. 90-105.

Huang, X.; Alleva, F.; Hon, H.W.; Hwang, M.Y. & Rosenfeld, R. (1993). The SPHINX II Speech Recognition System : An Overview. *Computer Speech and Language*, Vol. 2, No. 7, pp. 137-148.

Hung, H.; Jayagopi, D.; Yeo, C., Friendland, G., Ba, S., Ramchandran, J.; Mirghafori, N. & Gatica-Perez, D. (2007). Using Audio and Video Features to Classify The Most Dominant Person in A Group Meeting. *Proceedings of ACM Multimedia Conference*, pp. 835–838.

Jiang, Y.-G.; Yanagawa, A.; Chang, S.-F. & Ngo, C.-W. (2008). CU-VIREO374: Fusing Columbia374 and VIREO374 for Large Scale Semantic Concept Detection. *Columbia University ADVENT Technical Report* #223-2008-1.

Law-To, J.; Chen, L .; Joly, A.; Laptev, I.; Buisson, O.; Gouet-Brunet, V.; Boujemaa, N. & Stentiford, F. (2007). Video Copy Detection : A Comparative Study. *Proceedings of ACM International Conference on Image and Video Retrieval*, pp. 371-378.

Likas, A.; Vlassis, N. & Verbeek, J.J. (2003). The Global K-means Clustering Algorithm. *Pattern Recognition*, Vol. 36, pp. 451-461.

Lowe, D. (2004) Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, Vol. 60, No. 2, pp. 91-110.

Platt, J.C.; Czerwinski, M. & Field, B.A. (2003). PhotoTOC: Automating Clustering for Browsing Personal Photographs. *Proceedings of IEEE Pacific Rim Conference on Multimedia*, pp. 6-10.

Rienks, R.; Zhang, D. & Post, W. (2006). Detection and Application of Influence Rankings in Small Group Meetings. *Proceedings of International Conference on Multimodal Interfaces*, pp. 257-264.

Sivic, J. & Zisserman, A. (2003). Video Google: A Text Retrieval Approach to Object Matching in Videos. *Proceedings of the International Conference on Computer Vision*, Vol. 2, pp. 1470-1477.

Snoek, C.G.M. & Worring, M. (2005). Multimodal Video Indexing: A Review of the State-of-the-art. *Multimedia Tools and Applications*, Vol. 25, No. 1, pp. 5-35.

Snoek, C.G.M.; Huurnink, B.; Hollink, L.; de Rijke, M.; Schreiber, G. & Worring, M. (2007). Adding Semantics to Detectors for Video Retrieval. *IEEE Transactions on Multimedia*, Vol. 9, No. 5, pp. 975-986.

Steinmetz, R. (1996). Human Perception of Jitter and Media Synchronization. *IEEE Journal on Selected Areas in Communications*, Vol. 14, No. 1, pp. 61-72.

Tong, H.; Li, M.; Zhang, H.-J. & Zhang, C. (2004). Blur Detection for Digital Images Using Wavelet Transform. *Proceedings of IEEE International Conference on Multimedia & Expo*, pp. 17-20.

Vinciarelli, A. (2007). Spearkers Role Recognition in Multiparty Audio Recordings Using Social Network Analysis and Duration Distribution Modeling. *IEEE Transactions on Multimedia*, Vol. 9, No. 6, pp. 1215-1226.

Vinciarelli, A. & Favre, S. (2007). Broadcast News Story Segmentation Using Social Network Analysis and Hidden Markov Models. *Proceedings of ACM Multimedia*, pp. 261-264.

Weng, C.-Y.; Chu, W.-T. & Wu, J.-L. (2009). RoleNet: Movie Analysis from the Perspective of Social Networks. *IEEE Transactions on Multimedia*, Vol. 11, No. 2, pp. 256-271.

IntechOpen

Multimedia

Edited by Kazuki Nishi

ISBN 978-953-7619-87-9

Hard cover, 452 pages

Publisher InTech

Published online 01, February, 2010

Published in print edition February, 2010

Multimedia technology will play a dominant role during the 21st century and beyond, continuously changing the world. It has been embedded in every electronic system: PC, TV, audio, mobile phone, internet application, medical electronics, traffic control, building management, financial trading, plant monitoring and other various man-machine interfaces. It improves the user satisfaction and the operational safety. It can be said that no electronic systems will be possible without multimedia technology. The aim of the book is to present the state-of-the-art research, development, and implementations of multimedia systems, technologies, and applications. All chapters represent contributions from the top researchers in this field and will serve as a valuable tool for professionals in this interdisciplinary field.

INTECH

open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen