Classifying Video Documents by Hierarchical Structure of Video Contents

DUEN-REN LIU, CHEN-HSIEN LIN AND JING-JANG HWANG

Institute of Information Management, National Chiao Tung University, 1001 Ta Hsueh Rd., Hsinchu, Taiwan, Republic of China
Email: dliu@iim.nctu.edu.tw

Recent advances in video database technology and increasing use of the World Wide Web allow users to access video documents on the Web. Moreover, advances in digital library technologies necessitate the storing of large quantities of video documents. Consequently, video documents must be categorized to assist the user in browsing and searching them by category. This study presents a novel approach to classifying video documents under classes in a predefined class hierarchy. Video documents are classified according to the structural weight of video contents, capturing their characteristics with respect to the hierarchical structure of the video data. Furthermore, while considering the hierarchical relationship among classes, the proposed technique strives for an appropriate balance between specificity and exhaustivity in video classification.

Received 29 January 1998; revised 8 August 2000

1. INTRODUCTION

Constructing databases to store document files in any format (e.g. text, graphic, image or video) involves classifying each document file into related subjects [1, 2, 3, 4, 5, 6, 7]. Classification enables users to access documents according to their interests through related subject classes (categories). Recent advances in multimedia technologies on the World Wide Web have made it possible to browse video documents on the Web. Furthermore, advances in digital library technologies and applications make it necessary to store large amounts of video documents in digital video libraries. Video classification can provide video databases with a hierarchical semantic (subject) structure, facilitating effective searching and navigation through large collections of video data. An interesting application is to provide the retrieval of video documents on the Web, allowing users to browse and navigate video data hierarchically according to their interest categories. This work investigates how to classify video documents into appropriate subject classes according to the hierarchical structure of video contents.

The classification of visual information and images has received considerable interest. WebSEEk, a web visual information retrieval system [1], provides hierarchical browsing and navigation of visual information based on its categorization. The semantic categories and their hierarchical structure are constructed semi-automatically. Meanwhile, visual data is classified into classes based on mapping keywords into classes, with keywords being extracted from html tags associated with the visual data. However, the classification does not consider the hierarchical structure of video contents. Huang *et al.* [7] have proposed an approach for the hierarchical classification of images based on singular value decomposition with banded color correlogram for modeling and organizing image features. The Informedia Project [8] is developing a large on-line digital video library. The project mainly investigates automatic encoding, segmenting and indexing techniques to support full-content and knowledge-based search and retrieval for video data.

Video classification is performed based on video contents. Considerable work has been done on the automatic parsing and analysis of video contents, as well as the clustering and classification of the contents of a video document [9, 10, 11, 12, 13, 14, 15, 16]. The main focus has been on the automatic partitioning of full-motion videos [9], automatic cut detection methods [10], three-step methodology for automatically detecting film genres [12], and the shot classification method for classifying key-frames so as to select representative key-frames for shots [13]. Other research areas include methods of calculating the similarity between shots to categorize key-frames [14] and a generalized top-down hierarchical clustering method to develop hierarchical representations of videos [15]. Kobla and Doermann [16] have proposed Video Trails for representing and visualizing the structure of video sequences. Most of these studies investigate automatic video content analysis, clustering or classification in order to detect, recognize or classify film genres, shots or scenes in a video document. In general, the classification considered within a video document is a fine-grained classification;

the problem considered herein is taking the whole video document as a unit and classifying it under appropriate subject classes.

Several approaches have also been proposed to detect video segments or news stories in news videos. Nakamura and Kanade [11] have proposed a spotting by association method to extract video contents in news videos. Video segments with different types of topics such as speech/opinion and meeting/conference are detected by associating language clues (keywords and key sentences) and image clues (key images). Meanwhile, keywords and key sentences are obtained by spotting, parsing and lexical analysis of the closed-caption transcripts of news videos. Merlino *et al.* [17] have proposed methods of detecting story segments of news videos by correlating the textual, video and audio segment cues. Huang *et al.* [18] also have used the idea of cues (audio and video cues) in the segmentation of news stories and the reconstruction of the semantic structure of news broadcasting. These works focus mainly on automatic content extraction for news videos; this differs from the objective of this study, which is to classify a given video document under a class hierarchy.

Although feasible, automatic extraction of video contents can only be achieved to a limited extent. Some semantic notions are rather difficult or even impossible to extract automatically; the higher the level of the semantic notion, the more difficulty in automatic extraction. Various techniques for automatic extraction of video contents can be incorporated into content-recognition tools to automatically or semi-automatically generate the video structure and semantic notions, and they can further be used as the basis for video classification.

In addition to the research on the automatic extraction of video contents, several studies of content-based retrieval and indexing [19, 20, 21, 22, 23, 24, 25, 26] have focused on using keywords or annotations (textual descriptions) to describe the semantic notions of video contents, including individuals, things, events and locations [22, 24, 25]. Indexing can thus be created based on keywords or annotations, thereby allowing users to retrieve video data based on the contents. Moreover, a video document comprises levels of video units with a hierarchical structure. The video units include compound units, sequences (or segments), scenes, shots and frames [22, 24]. Rowe *et al.* [24] defined several types of indexes, including bibliographic, structural, keyword and object indexes, to answer bibliographic, object and structural queries, as well as queries about topic content. Hielsvold *et al.* [22] described the modeling of video data. Thematic indexing, including person, event and location annotations, is used to thoroughly describe the contents of a video document.

The growing need to support content-based retrieval, search and filtering has made the standardized description of multimedia content possible. MPEG-7 (multimedia content description interface) [27] is an ISO/IEC standard being developed by MPEG (Moving Picture Experts Group) for describing multimedia content data. MPEG-7 aims to allow efficient search of multimedia content through standardized descriptions. Description schemes (DSs) being developed for MPEG-7 are used to describe and annotate the structural and semantic aspects of audio-visual (AV) contents.

Automatic classification of full-text documents [2, 3, 5, 28] has been extensively studied. Wong *et al.* [5] have proposed an algorithm, ACTION, to classify full-text documents. By measuring the significance of each class defined in a class hierarchy, the algorithm classifies a given document under classes with the highest significance values. ACTION not only considers the frequency with which a keyword appears, but also effects an appropriate balance between specificity and exhaustivity. Specificity measures the degree of precision with which the assigned class represents the content of the document. Exhaustivity measures the extent of the coverage by the assigned class on the subjects contained in the document [5]. Ng *et al.* [3] have proposed an automated learning approach to categorize text documents into a tree of categories. Additionally, the mining association rules technique has been applied to the extraction of classification knowledge from Internet text documents [2].

This work proposes a novel approach to the classification of a video document under appropriate subject classes according to the content-bearing keywords (semantic notions) and the structural weight of the video content. The approach is based on the video content available by extracting manually by human annotation, or automatically (semi-automatically) by means of content processing and analysis tools. The proposed approach classifies a given video document under one or more appropriate classes defined in a given class hierarchy; each class represents a different subject of interest and the video classification value (VCV) and significance value (SV) of each class is measured to determine the appropriateness of the classification results. The VCV measures the appropriateness of a class to represent the subjects of a given video document, while the SV quantitatively measures the significance of a class as the classification result (Section 3 describes their computation in more detail). The proposed method computes the VCV and SV of each class based on the content-bearing keywords and the structural weight of the video contents. Final classification results can be determined according to various classification criteria. For example, the class with the highest SV (or VCV) is selected as the classification result.

Two further aspects are addressed.

- *Hierarchical structure of video documents.* A video document comprises video units with a hierarchical structure, including compound units, sequences, scenes, and shots. The proposed approach assigns keywords used to describe the contents of video units at higher levels of the hierarchical structure with higher weights when measuring the VCV, because they more appropriately represent the main subjects of a video document.
- *Specificity and exhaustivity.* Specificity and exhaustivity are two metrics used to describe the effectiveness of classifying a document under the class or classes

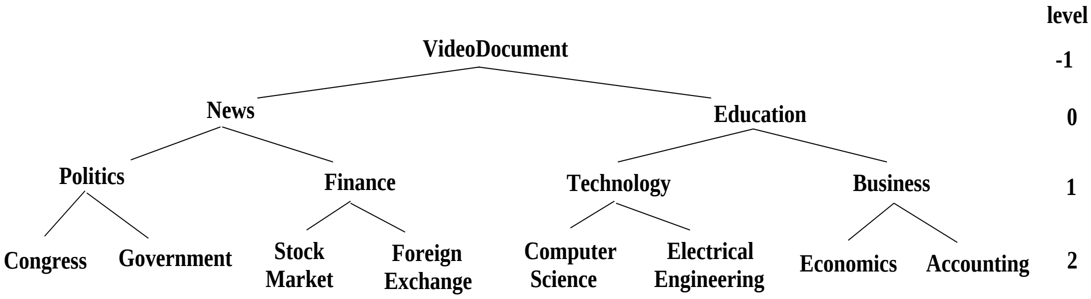

FIGURE 1. Class hierarchy.

defined in a class hierarchy. The two metrics contradict each other. Wong *et al.* [5] proposed an algorithm to classify full-text documents, capable of achieving an appropriate balance between specificity and exhaustivity. In this work, the calculation of significance value is based on the video classification value, and also targets to achieve an appropriate balance between specificity and exhaustivity by considering the hierarchical relationships between classes and subclasses.

The rest of this paper is organized as follows. Section 2 introduces the underlying concept of video classification. Section 3 then discusses the proposed video classification method. Next, Section 4 presents examples to analyze the classification method. Section 5 illustrates the results of experimental evaluations. Concluding remarks and suggestions for future work are finally made in Section 6.

2. UNDERLYING CONCEPT OF VIDEO CLASSIFICATION

Two aspects must be addressed with respect to video classification: video structure and class–subclass relationships in the class hierarchy, as described in the following.

2.1. Class hierarchy

The predefined classes are organized into a hierarchical structure, i.e. the class hierarchy. Class hierarchy defines the contextual and logical relationships between classes and subclasses. A video document is classified under one or more classes defined in the hierarchy. A class contains several subclasses, and a subclass can be divided into further subclasses, and so on. Class names represent categories of subjects of interest.

A tree structure can be used to represent the class hierarchy. Each node in the tree corresponds to a defined class. Parent–child relationships in the tree correspond to class–subclass relationships in the class hierarchy. The root node of the tree represents the virtual class, ‘VideoDocument’, which is a class defined to encompass all video documents. Associated with each class (node) C_i is the level number of class C_i, denoted as $level(C_i)$. To distinguish the virtual class ‘VideoDocument’ from other defined classes, the level number of the root node (virtual class) of the tree is set to -1. The level number of the child of the root node (virtual class) is 0. For all subsequent nodes (classes), the level number is one plus the level number of the node’s parent. Figure 1 provides an illustrative example of class hierarchy. Under the root node, the ‘VideoDocument’ class, there are ‘News’ and ‘Education classes’. The ‘News’ class contains ‘Politics’ and ‘Finance’ subclasses. This figure also depicts the level number of classes at the same level. For instance, the ‘News’ class has level number 0 and the ‘Stock Market’ class has level number 2.

In the class hierarchy, the lower (deeper) the level of the class hierarchy the narrower the coverage of the subjects. In general, a video document may contain several subjects. A low level of class suggests that the class can represent more specific subjects of a video document, i.e. the specificity is high. On the other hand, the higher (upper) the level of the class, the more distinct subjects of a video document the class can cover, i.e. the exhaustivity is high. In addition, the two metrics contradict each other. While considering the two metrics, we apply the notion of full-text documents (as proposed in [5]) as our basis for video classification. In doing so, an appropriate balance can be achieved between specificity and exhaustivity.

2.2. Video structure and keywords

Video documents must be preprocessed to extract the video structure and content-bearing keywords (semantic notions). Several video parsing and processing techniques have been proposed for analyzing video contents [8, 10, 11, 12, 13, 15, 16, 21, 26]. Keywords can also be extracted automatically by processing movie transcripts, or by closed-caption transcripts of news videos as suggested in [11, 17, 18]. Video content may also be represented in the MPEG-7 standard in the future. The structure-based DSs in MPEG-7 can be used to describe the structural aspects of the video content, and develop a hierarchical tree structure of video segments. Meanwhile, the semantics-based DSs can be employed to describe the conceptual aspects (semantic notions) of the video content [27]. For video content represented in the MPEG-7 standard, the structure and keywords can be obtained by processing the structure-based DSs and semantic-based DSs of MPEG-7. Note that this work does not focus on preprocessing video documents; rather, it assumes herein that video documents have already been preprocessed to obtain the video structure and the

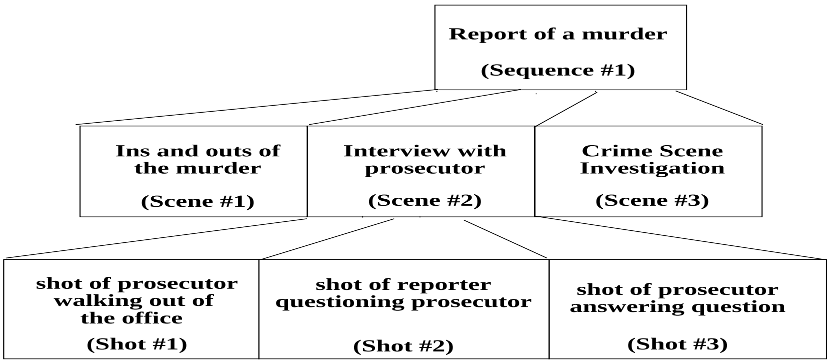
Report of a murder
(Sequence #1)
Ins and outs of the murder
(Scene #1)
Interview with prosecutor
(Scene #2)
Crime Scene Investigation
(Scene #3)
shot of prosecutor walking out of the office
(Shot #1)
shot of reporter questioning prosecutor
(Shot #2)
shot of prosecutor answering question
(Shot #3)

FIGURE 2. An example of a video structure.

TABLE 1. Keywords.

Video unit	Keywords (occurrence frequency)
Sequence #1	Governor(2), Murder(2), Mansion(1), Shot(1), State(1), Police(1), Prosecutor(1), Election(2)
Scene #1	Governor(1), Murder(1), Mansion(1), State(1), Police(1), Prosecutor(1), Election(2)
Scene #2	Governor(1), Murder(1), Mansion(1), Prosecutor(1), Election(1), Witness(1), Suspect(1)
Scene #3	Governor(1), Murder(1), Mansion(1), Shot(1), Police(1), Election(1), Witness(1), Gun(1)

content-bearing keywords. The structure and keywords have been established either manually [22, 23, 24] or by video parsing and processing systems [6, 8, 11, 17, 18, 21, 26]. The above assumption is a limitation of this work.

The hierarchical structure of a video document, i.e. video hierarchy, is constructed by components such as shots, scenes, sequences and compound units [22]. A compound unit consists of sequences, a sequence consists of scenes, and a scene consists of shots. The relationships between levels of video units can be represented as a tree structure. Each node in the tree corresponds to a video unit (i.e. a sequence or a scene). The level number of the tree's root node is 0. For all subsequent nodes, the level number is one plus the level number of the node's parent. The height of a video hierarchy (tree) is one plus the maximum level number of any node in the video hierarchy.

On the basis of necessary details of the contents of the video documents, the classification method can determine the levels of the video contents deemed necessary for the classification process. For instance, the chosen levels of the video contents may range from the compound-unit level to the scene level, or from the sequence level to the scene level. Figure 2 provides an example report of a news item, Governor murdered during election campaign at governor mansion. For simplicity, assume that the given video document is a report of a news item. A sequence-level video unit can represent the news item, possibly containing several scene-level video units.

For different levels of video units, keywords or annotations can be used to describe their contents. Table 1 lists the keywords of the news item for sequence-level and scene-level video units. The number inside the parenthesis is the frequency with which the keyword occurs in the descriptions of the contents of the video unit. For instance, the 'Murder' keyword occurs once in scene 2, the 'Police' keyword occurs once in sequence 1, and the keyword 'Governor' occurs twice in sequence 1.

Note that the sequence-level keywords represent the essential contents of a video document, while scene-level keywords only describe the subjects of a scene which is only a portion of a sequence. Restated, the sequence-level keywords are abstractions of scene-level keywords. Consequently, sequence-level keywords more appropriately represent the main subjects of a video document than the scene-level keywords. Therefore, keywords occurring at higher levels of video structure are assigned higher weights when measuring the video classification value.

3. VIDEO CLASSIFICATION METHOD

In this section, we present the proposed video classification method. The method consists of two main parts, i.e. the calculation of video classification values and the calculation of significance values, as described in subsequent sections. Table 2 lists the symbols and their definitions in this work.

In Section 3.1, we illustrate how to compute video classification value, without considering the hierarchical relationships between the classes and subclasses. Video classification value is computed according to keyword video weight and keyword class weight. Keyword video weight is defined to consider the property of video documents, i.e. video structure. Keyword class weight is defined to consider the flexibility in mapping a keyword to a class. A class with a higher value of VCV more appropriately represents the subjects of a video document.

In Section 3.2, we describe how to calculate the

TABLE 2. Symbols and definitions.

Symbol	Meaning
$Occur(K_p, l_i)$	The occurrence frequency of keyword K_p occurring at level l_i of video hierarchy.
$KVW(K_p)$	The keyword video weight is a measurement of keyword K_p's importance in representing the subjects of video contents.
$KCW(K_p, C_i)$	The keyword class weight is the weight of mapping keyword K_p to class C_i.
$KCV(K_p, C_i)$	The keyword classification value is a measurement of K_p's contribution to quantify the appropriateness of class C_i to represent the subjects of a video document.
$VCV(C_i)$	The video classification value of a class C_i is a measurement of the appropriateness of C_i to represent the subjects of a video document.
$EVCV(C_i)$	The exhaustive video classification value of a subtree rooted at the class C_i measures the exhaustivity of the subtree contributing to the calculation of the significance value.
$SV(C_i)$	The significance value of a class C_i is a measurement to quantify the significance of C_i as the classification result.

significance value, which considers the hierarchical relationships between classes and subclasses. The significance value is calculated on the basis of two metrics, specificity and exhaustivity as proposed by [5] for classifying text-documents. Herein, we redefine the significance value on the basis of the video classification value, in order to adapt to the classification of video documents. If only the VCV is considered, a larger VCV of a class implies a more proper class to represent the subjects of a video document. However, the significant value described in Section 3.2 may be the criterion to determine the final classification results. The approach may select several classes with the highest significance values as the results.

3.1. Video classification: video classification value

This subsection explains the computation of keyword video weight, keyword class weight, keyword classification value and video classification value.

3.1.1. *Keyword video weight*

According to the characteristics of the video structure, we define a metric, keyword video weight (*KVW*), in Definition 1. The higher the *KVW* of a keyword, the more important it will be in representing the subjects of video contents.

DEFINITION 1. *(Keyword video weight) Keyword video weight, denoted by KVW(K_p), is defined as a measurement of the keyword K_p's importance in representing the subjects of video contents*

$$KVW(K_p) = \sum_{l_i} Occur(K_p, l_i) * (h - l_i) \quad \begin{cases} l_i: & 0 \ldots h-1; \textit{video level number}; \\ h: & \textit{the height of video hierarchy}; \end{cases} \tag{1}$$

$$Occur(K_p, l_i) = \sum_{A_j^{l_i}} freq(K_p, A_j^{l_i}) \quad A_j^{l_i} \textit{ is the video unit at level } l_i. \tag{2}$$

$Occur(K_p, l_i)$ is the occurrence frequency of keyword K_p occurring at video level l_i. $freq(K_p, A_j^{l_i})$ is the occurrence frequency of keyword K_p occurring at video unit $A_j^{l_i}$. $KVW(K_p)$ is the summation of the product of $Occur(K_p, l_i)$ and $(h - l_i)$, for all l_i. Equation (1) uses the occurrence frequency of a keyword to calculate the *KVW*. Notably, the points of view of users can also be considered in calculating the weights. Equation (1) can be generalized to $KVW(K_p) = \sum_{l_i} W(K_p, l_i) * (h - l_i)$. $W(K_p, l_i)$ can be the occurrence frequency or simply a user assigned weight, $UW(K_p)$, which is used to describe the importance of the keyword in representing the video content. This equation can also be defined by considering both the occurrence frequency and the user assigned weight, such as $W(K_p, l_i) = Occur(K_p, l_i) * UW(K_p)$.

Considering two keywords, K_p and K_q, if the occurrence frequency of K_p at sequence level is the same as that of K_q at the scene level, k_p is assigned a higher keyword video weight than k_q, since the level number of the sequence level is smaller than the level number of the scene level. Thus, K_p is assigned a higher weight than K_q. As mentioned earlier, sequence level keywords more appropriately represent the main subjects of video contents than scene level keywords. For instance, Figure 3 depicts a video document that contains sequence level and scene level video units in which the associated level number is 0 and 1, respectively. The height of the video hierarchy is 2. For the keyword 'Murder' described in Table 1, its occurrence frequencies at sequence 1, scene 1, scene 2, and scene 3, are 2, 1, 1, and 1, respectively. The occurrence frequencies of 'Police' keyword at sequence 1, scene 1, scene 2, and scene 3, are

level 0	**sequence #1 Governor Murdered; Election**		
level 1	**scence #1 CNN Report**	**scene #2 Interview with Prosecutor**	**scene #3 Crime Scene Investigation**

FIGURE 3. Video units and their corresponding level numbers.

1, 1, 0, and 1, respectively. The keyword video weight of ‘Murder’, *KVW*(‘Murder’), equals $2 * (2 - 0) + (1 + 1 + 1) * (2 - 1) = 7$. The keyword video weight of ‘Police’, *KVW*(‘Police’), equals $1*(2-0)+(1+0+1)*(2-1) = 4$. Keyword ‘Murder’ is more important than ‘Police’ in terms of representing the subjects of the video contents.

3.1.2. *Keyword class weight*

After computing the *KVW* of each keyword, the mappings of keywords to their related classes must be determined. Definition 2 defines the keyword class weight. The higher the value of $KVW(K_p, C_i)$, the higher probability it will be that K_p belongs to class C_i.

DEFINITION 2. *(Keyword class weight) Keyword class weight, denoted by* $KCW(K_p, C_i)$*, is defined as the weight of mapping keyword* K_p *to class* C_i*.*

A keyword can be mapped into one or more classes with different probabilities. The mapping may be executed either as a mapping with probability distributions in the range of 0 to 1, or as a simple matching that has a probability of either 0 or 1 (no match or match). However, the probability values can be scaled to relative values such that their values may exceed the range of 0 to 1. Assume that the mappings and their probabilities can be accumulated based on statistics of historical data. The mapping probability of mapping K_p into subject class C_i can be defined as the total number of occurrences of K_p in all video documents classified under C_i divided by the total number of occurrences of K_p in all video documents. If no such historical information is available, a user-assigned weight can be used in place of *KCW*.

Keywords can serve as cues for analyzing the contents, as suggested in [11, 17, 18], in which frequent words are used as text cues to identify story or video segments with different types of topics in news videos. In addition to their usefulness for analyzing video contents, keywords also play an important role in classifying a video document into appropriate classes. The *KCW* models the importance of keywords in the mapping of the subject classes. Some nondescriptive keywords, which are meaningless in the subject classification, should be assigned a very low or zero mapping probability. Meanwhile, some specific keywords are crucial in determining the mapping to a particular subject class, and require a higher *KCW*. Besides, considering all available keywords may impair classification performance. Selecting an appropriate set of subject-related keywords will ensure the efficiency and quality in classification. The subject-related keywords may be selected automatically or human-assisted. If the historical classification data is available, the set of subject-related keywords can be selected by including frequent keywords from all subject classes. Frequent keywords from a given subject class are those that frequently appear in video documents classified under the subject class. For domain specific applications, domain knowledge can also be used for semantic analysis when detecting subject-related keywords. Further work on the detection and selection of subject-related keywords will be investigated in the future.

Assume that the mappings of keywords ‘Murder’ and ‘Police’ to corresponding classes are the following: *KCW*(‘Murder’, ‘Politics’) = 0.1; *KCW*(‘Murder’, ‘Society’) = 0.4; *KCW*(‘Murder’, ‘Crime’) = 0.7; *KCW*(‘Police’, ‘Politics’) = 0.1; *KCW*(‘Police’, ‘Society’) = 0.3; *KCW*(‘Police’, ‘Crime’) = 0.6. The KCWs referred to in the examples are scaled values. If $KCW(K_p, C_i) > KCW(K_p, C_j)$, $i \neq j$, the probability of K_p being mapped into class C_i surpasses the probability of K_p being mapped into class C_j. For instance, *KCW*(‘Murder’, ‘Crime’) > *KCW*(‘Murder’, ‘Politics’), the possibility of ‘Murder’ being classified under ‘Crime’ class exceeds the possibility of ‘Murder’ being classified under ‘Politics’ class.

3.1.3. *Keyword classification value*

DEFINITION 3. *(Keyword classification value) Keyword classification value,* $KCW(K_p, C_i)$*, is defined as a measurement of keyword* K_p*’s contribution to quantify the appropriateness of class* C_i *to represent the subjects of a video document*

$$KCV(K_p, C_i) = KVW(K_p) * KCW(K_p, C_i). \qquad (3)$$

$KCV(K_p, C_i)$ is proportional to both $KVW(K_p)$ and $KCW(K_p, C_i)$. If the keyword video weight of K_p is larger or the probability of mapping K_p to C_i is larger, the value of $KCW(K_p, C_i)$ is also larger. For instance, the keyword video weights of ‘Murder’ and ‘Police’ are 7 and 4, respectively, as illustrated in Section 3.1.1. The calculations of *KCVs* of keywords ‘Murder’ and ‘Police’ are the following: *KCV*(‘Murder’, ‘Crime’) = *KVW*(‘Murder’) $*$ *KCW*(‘Murder’, ‘Crime’) $= 7 * 0.7 = 4.9$; *KCV*(‘Police’, ‘Crime’) = *KVW*(‘Police’) $*$ *KCW*(‘Police’, ‘Crime’) $= 4 * 0.6 = 2.4$. The *KCV* (4.9) of ‘Murder’ is larger than the *KCV* (2.4) of ‘Police’. This observation implies that the keyword ‘Murder’ contributes more than the keyword ‘Police’ does in terms of measuring the appropriateness of ‘Crime’ class to represent the subjects of the video document.

3.1.4. Video classification value

The definition of video classification value and its computation are provided in Definition 4 and Equation (4), respectively.

DEFINITION 4. *(Video classification value) Video classification value, denoted as $VCV(C_i)$, is defined as a measurement of the appropriateness of class C_i to represent the subjects of a video document*

$$VCV(C_i) = \sum_{p=1,\ldots,m} KCV(K_p, C_i) = \sum_{p=1,\ldots,m} (KVW(K_p) * KCW(K_p, C_i)). \quad (4)$$

p is equal to $1, 2, \ldots, m$, and m is the total number of keywords in the video document. The video classification value of a class C_i is the total of all the measurements of keywords' contributions to quantify the appropriateness of class C_i in representing the subjects of the video document. $VCV(C_i)$ is the summation of the $KCW(K_p, C_i)$, i.e. the summation of the product of $KVW(K_p)$ and $KCW(K_p, C_i)$, for all K_p in the video document. The larger the video classification value of a class the more appropriate is the class to represent the subjects of the given video document.

Assume that the given video document only contains two keywords 'Murder' and 'Police'. The calculations of video classification values of 'Politics', 'Society' and 'Crime' classes are the following: $VCV(\text{'Politics'}) = KVW(\text{'Murder'}) * KCW(\text{'Murder', 'Politics'}) + KVW(\text{'Police'}) * KCW(\text{'Police', 'Politics'}) = 7 * 0.1 + 4 * 0.1 = 1.1$; $VCV(\text{'Society'}) = KVW(\text{'Murder'}) * KCW(\text{'Murder', 'Society'}) + KVW(\text{'Police'}) * KCW(\text{'Police', 'Society'}) = 7 * 0.4 + 4 * 0.3 = 4.0$; $VCV(\text{'Crime'}) = KVW(\text{'Murder'}) * KCW(\text{'Murder', 'Crime'}) + KVW(\text{'Police'}) * KCW(\text{'Police', 'Crime'}) = 7 * 0.7 + 4 * 0.6 = 7.3$. Comparing the three classes, $VCV(\text{'Crime'}) > VCV(\text{'Society'}) > VCV(\text{'Politics'})$, the 'Crime' class is the most appropriate to represent the subjects of the given video document.

3.2. Video classification: significance value

Significance value is a numerical measurement to quantify the significance of each defined class as the classification result for a given video document. The computation of significance value targets to achieve the appropriate balance between the specificity and exhaustivity by considering the hierarchical relationships between classes and subclasses. Selecting a lower-level (larger level number) class as the classification result achieves good specificity, but at the expense of exhaustivity. On the other hand, selecting a higher-level (smaller level number) class achieves good exhaustivity, but at the expense of specificity.

Achieving appropriate balance between the two metrics is an important design perspective of document classification algorithms. To achieve the above goal, Wong *et al.* [5] defined the calculation of significance value and proposed some classification rules for classifying text documents. In this work, we modify their classification rules as well as the calculation of significance value to fulfil the requirement of classifying video documents. The modified classification rules are as follows. Note that Rule 2 and Rule 4 are adapted from [5], while Rule 1 and Rule 3 are newly added.

RULE 1. For a given video document D and class hierarchy H, the class with the larger video classification value should be weighted larger in the calculation of the significance value.

The VCV measures the appropriateness of the class to represent the subjects of the given video document D. A larger VCV of the class implies a more significant class to represent D.

RULE 2. For a given video document D and class hierarchy H, if more than one class having the same video classification value exists, the class with the higher level number should be weighted higher in the calculation of the significance value.

Two classes having the same VCV are equally appropriate in representing the subjects of the given video document. The class with the higher level number is more specific as the representative of the given document and should be weighted higher in the calculation of the significance value, in order to achieve a higher specificity.

RULE 3. For a given video document D and class hierarchy H, if more than one class having the same video classification value exists, the class C_i having the higher sum of the video classification values of all successor classes of C_i, should be weighted higher in the calculation of the significance value.

Two classes having the same VCV are equally appropriate in representing the subjects of the given document D. The class C_i having the higher sum of VCVs of all successor classes of C_i more exhaustively represents the subjects of D, since the higher the summation of VCVs implies a higher degree of the coverage of the subjects of D. To achieve higher exhaustivity, the class C_i should be weighted higher in the calculation of the significance value.

RULE 4. For a given video document D and class hierarchy H, if more than one class having the same significance value exists, the class with the lowest level number is chosen as the classification result.

Two classes having the same SV are equally significant to represent the given document D. To achieve a higher exhaustivity, the class with the lower level number is chosen as the classification result. This is attributed to the fact that the class with the lower level number is more exhaustive than classes with higher level numbers.

According to Rule 2, the level of a class C_i, $level(C_i)$, can be used to reflect the specificity of the class C_i. Within the class hierarchy, the deeper (larger) the level the more specific the class. According to Rule 3, the sum of VCVs of all classes in the sub-hierarchy rooted at the class C_i is used to reflect the exhaustivity. A class at a deeper level of the class hierarchy has a higher level number (higher specificity), but has lower sum of VCVs of all classes in the sub-hierarchy rooted at the class (lower exhaustivity). On the other hand, a class, at an upper level of the class hierarchy, has a lower level number (lower specificity), but has a higher sum of VCVs of all classes in the sub-hierarchy rooted at the class (higher exhaustivity). The measurement of the significance value attempts to balance specificity and exhaustivity. By adapting the definitions of the calculation of significance value proposed by Wong *et al.* [5] for text-document classification, we define an exhaustive video classification value in Definition 5 and redefine the definition of the significance value to be suited for video classification in Definition 6.

DEFINITION 5. *(Exhaustive video classification value (EVCV)) Exhaustive video classification value of a subtree rooted at the class C_i, denoted by $EVCV(C_i)$, measures the exhaustivity of the subtree contributing to the calculation of the significance value*

$$EVCV(C_i) = \sum_J VCV(J) \quad J = C_i \\ \text{or} \quad J \text{ is a successor of } C_i. \tag{5}$$

$EVCV(C_i)$ measures the total VCV of a class C_i and VCVs of all C_i's successor classes. The $EVCV$ of a subtree rooted at the class C_i represents the extent of the coverage of the subjects that the subtree covers. If the class C_i is a leaf node, $EVCV(C_i)$ equals $VCV(C_i)$.

DEFINITION 6. *(Significance value (SV)) Significance value of a class C_i, denoted by $SV(C_i)$, is a measurement to quantify the significance of C_i as the classification result*

$$\begin{aligned} SV(C_i) &= EVCV(C_i) * level(C_i) + \sum_J VCV(J) * level(J) \\ &\quad J : \text{ ancestor of } C_i \qquad (6) \\ &= \sum_I VCV(I) * level(C_i) + \sum_J VCV(J) * level(J) \\ &\quad I : C_i \text{ or successor of } C_i. \qquad (7) \end{aligned}$$

$SV(C_i)$ includes two terms.

- The first term, $EVCV(C_i) * level(C_i)$, is further equal to $\sum VCV(I) * level(C_i)$. This term is the sum of the product of $VCV(I)$ and $level(C_i)$ for all I which is in the subtree rooted at C_i, i.e. I is C_i or I is a successor of C_i. Notably, the level number used in the product is the level number of C_i. This term contains $EVCV$ to measure the exhaustivity contributing to the calculation of the significance value.
- The second term, $\sum VCV(J) * level(J)$, is the sum of the product of $VCV(J)$ and $level(J)$ for all J which is an ancestor class of C_i. This term seeks for specificity contributing to the calculation of the significance value.

If the class C_i is a leaf node, $SV(C_i)$ equals the sum of the product of $VCV(C_i)$ and $level(C_i)$, and the total of the product of $VCV(J)$ and $level(J)$ for all J which is an ancestor class of C_i.

After computing the significance values of all classes, classes can be selected according to different *classification criteria*.

1. The class with the highest significance value is selected as the classification result. If two or more classes have the same significance value, the class with the lowest level number (highest level) is chosen as the classification result.
2. If a given video document must be classified under N number of classes, the classes having a significance value in the top N significance values of all classes are selected as the classification results.
3. A threshold T of the significance value can also be set to select classes as the classification results. For instance, the classification selects all classes with significance values exceeding T as the classification results.
4. The classification criterion can also be defined using the video classification value instead of the significance value if specificity and exhaustivity are not primary concerns. A similar classification criterion based on the video classification values can be obtained by replacing the significance value with the video classification value in the above three criteria.

3.3. Algorithm

In this subsection, we describe the proposed classification algorithm. Figure 4 illustrates the ComputeVCV() algorithm used to calculate the video classification values. The algorithm starts by preprocessing the video document to construct the video hierarchy and keywords. The proposed algorithm calculates the keyword video weight of each keyword by applying Equation (1) and Equation (2). The keyword class weight is obtained by mapping the keyword K_p to class C_i. $KCW(K_p, C_i)$ is retrieved by searching the keyword-class inverted list as described in Section 3.1.2. Finally, the computation of the video classification value is according to the Equation (4).

Figure 5 shows the ComputeSV() algorithm used to calculate the significance values. The algorithm initially computes the video classification value of each class using the ComputeVCV() algorithm described in Figure 4. The algorithm then calculates the exhaustive video classification values and significance values in a depth-first traversal manner, using the DfsTravl() algorithm described in Figure 6. Finally, the algorithm determines the classification results according to the classification criterion described in Section 3.2.

The DfsTravl() algorithm precisely computes the exhaustive video classification values in a bottom-up manner by the depth-first traversal, i.e. it calculates $EVCV(Q)$ by adding $VCV(Q)$ with $\sum EVCV(C_i)$ for all C_i, the children of Q. The calculation easily confirms that the algorithm is the

```
Algorithm: ComputeVCV(D: video document; H: class hierarchy)
V : video hierarchy;
begin
    V <== Preprocessing(D);
    $h$ = the height of the video hierarchy of V;
    for each keyword $K_p$ in V do
        //comments: compute keyword video weight
        $KVW(K_p) = 0$;
        for $l = 0$ to $h - 1$ do
            $Occur(K_p, l) = \sum_{A_j^l} freq(K_p, A_j^l)$;
            $KVW(K_p) = KVW(K_p) + Occur(K_p, l) * (h - l)$;
        endfor;
    endfor;
    for each class $C_i$ in H do
        // comments: compute video classification value
        $VCV(C_i) = 0$;
        for each keyword $K_p$ in V do
            $KCW(K_p, C_i) <== Mapping(K_p, C_i)$;
            $VCV(C_i) = VCV(C_i) + KVW(K_p) * KCW(K_p, C_i)$;
        endfor;
    endfor;
end;
```

FIGURE 4. ComputeVCV algorithm.

```
Algorithm: ComputeSV(D: video document; H: class hierarchy)
begin
    // comments: compute video classification value
    ComputeVCV(D, H);
    // comments: Perform a Depth-First traversal to compute EVCVs and SVs;
    ancestorsSpecificity = 0;
    R = the root of the class hierarchy H; // i.e., the virtual class VideoDocument;
    for each child $C_i$ of R do
        DfsTravl($C_i$, ancestorsSpecificity);
    endfor;
    select classes as the classification results using the classification criterion;
end;
```

FIGURE 5. ComputeSV algorithm.

```
Algorithm: DfsTravl($Q$: node in class hierarchy; ancesS: sum of ancestors' VCV*levelNo )
begin
// comments: newAncesS is the sum of $Q$ and ancestors' VCV*levelNo;
newAncesS = ancesS + $VCV(Q) * level(Q)$;
// comments: Perform a recursive Depth-First traversal to compute EVCVs
// and significance values of all successor nodes of $Q$;
$EVCV(Q) = VCV(Q)$;
for each child $C_i$ of $Q$ do
    DfsTravl($C_i$, newAncesS);
    $EVCV(Q) = EVCV(Q) + EVCV(C_i)$;
endfor;
$SV(Q) = EVCV(Q) * level(Q)$ + ancesS;
end;
```

FIGURE 6. DfsTravl algorithm.

TABLE 3. Sample 1—KVW.

	Occurrence frequency				
Keywords	sequence#1 level 0	scene#1 level 1	scene#2 level 1	scene#3 level 1	KVW
Governor	2	1	1	1	$2 * (2-0) + 1 * (2-1) + 1 * (2-1) + 1 * (2-1) = 7$
Murder	2	1	1	1	$2 * (2-0) + 1 * (2-1) + 1 * (2-1) + 1 * (2-1) = 7$
Mansion	1	1	1	1	$1 * (2-0) + 1 * (2-1) + 1 * (2-1) + 1 * (2-1) = 5$
Shot	1	0	0	1	$1 * (2-0) + 0 * (2-1) + 0 * (2-1) + 1 * (2-1) = 3$
State	1	1	0	0	$1 * (2-0) + 1 * (2-1) + 0 * (2-1) + 0 * (2-1) = 3$
Police	1	1	0	1	$1 * (2-0) + 1 * (2-1) + 0 * (2-1) + 1 * (2-1) = 4$
Prosecutor	1	1	1	0	$1 * (2-0) + 1 * (2-1) + 1 * (2-1) + 0 * (2-1) = 4$
Election	2	2	1	1	$2 * (2-0) + 2 * (2-1) + 1 * (2-1) + 1 * (2-1) = 8$
Witness	0	0	1	1	$0 * (2-0) + 0 * (2-1) + 1 * (2-1) + 1 * (2-1) = 2$
Gun	0	0	0	1	$0 * (2-0) + 0 * (2-1) + 0 * (2-1) + 1 * (2-1) = 1$
Suspect	0	0	1	0	$0 * (2-0) + 0 * (2-1) + 1 * (2-1) + 0 * (2-1) = 1$
the height of the video hierarchy is 2					

TABLE 4. Sample 1—KCW.

	Classes								
Keywords	Society	Crime	Murder	Smuggle	Politics	Congress	Government	Election	Scandal
Governor	0.3	0	0	0	0.6	0.2	0.9	0.4	0.1
Murder	0.4	0.7	0.9	0.1	0.1	0	0	0	0
Mansion	0.2	0	0	0	0.2	0.1	0.4	0.1	0
Shot	0.3	0.6	0.6	0.1	0.0	0	0	0	0
State	0.1	0	0	0	0.4	0.3	0.6	0.1	0
Police	0.3	0.6	0.5	0.3	0.1	0	0	0	0
Prosecutor	0.2	0.6	0.5	0.2	0.3	0	0.2	0.1	0
Election	0.1	0	0	0	0.6	0.3	0.6	0.9	0.1
Witness	0.1	0.5	0.4	0.1	0	0	0.1	0	0
Gun	0.3	0.6	0.6	0.3	0.0	0.0	0.0	0.0	0
Suspect	0.1	0.7	0.6	0.3	0	0	0	0	0

same as the calculation of *EVCV* in Equation (5). The calculation of significance value consists of two terms, $EVCV(C) * level(C)$ and $\sum_J VCV(J) * level(J)$, by Equation (6). The first term is easily calculated using the *EVCV* obtained in the bottom-up manner. The second term is calculated by accumulating the ancestors' $VCV() * level()$, i.e. $\sum_J VCV(J) * level(J)$, in a top-down manner by the depth-first traversal.

4. ANALYSIS

In this section, we use two sample video documents to illustrate the classification process. The video classification value is calculated according to the algorithm ComputeVCV(). The significance value is calculated according to the algorithm ComputeSV().

4.1. Sample 1

Sample 1 is a video document which records a news report of 'Governor murdered during election campaign at governor mansion'. The video document contains three scenes: 'CNN report', 'Interview with Prosecutor' and 'Crime scene investigation'. The subjects of the video document are related to both murder and the election. To simplify our illustration, we only select some keywords to describe the video contents. Also, only some classes are selected in the illustration to clarify the explanation.

Table 3 lists the keywords, occurrence frequency in which a keyword occurs at each video unit, and the video unit's corresponding level number. The height of the video structure is 2. This table also contains the keyword video weight of each keyword calculated according to the algorithm ComputeVCV(). For instance, $KVW(\text{'Governor'}) = 2*(2-0)+1*(2-1)+1*(2-1)+1*(2-1) = 7$. Table 4 lists the keyword class weight. Note that the weights are scaled values; thus the sum of $KCW(K_p, C_i)$ for all C_i does not equal 1. The keyword classification values obtained by the product of KVWs and KCWs, i.e. the values listed in Table 3 and Table 4, respectively, are shown

TABLE 5. Sample 1—VCV.

	Keyword classification value								
Keywords	Society	Crime	Murder	Smuggle	Politics	Congress	Government	Election	Scandal
Governor	2.1	0.0	0.0	0.0	4.2	1.4	6.3	2.8	0.7
Murder	2.8	4.9	6.3	0.7	0.7	0.0	0.0	0.0	0.0
Mansion	1.0	0.0	0.0	0.0	1.0	0.5	2.0	0.5	0.0
Shot	0.9	1.8	1.8	0.3	0.0	0.0	0.0	0.0	0.0
State	0.3	0.0	0.0	0.0	1.2	0.9	1.8	0.3	0.0
Police	1.2	2.4	2.0	1.2	0.4	0.0	0.0	0.0	0.0
Prosecutor	0.8	2.4	2.0	0.8	1.2	0.0	0.8	0.4	0.0
Election	0.8	0.0	0.0	0.0	4.8	2.4	4.8	7.2	0.8
Witness	0.2	1.0	0.8	0.2	0.0	0.0	0.2	0.0	0.0
Gun	0.3	0.6	0.6	0.3	0.0	0.0	0.0	0.0	0.0
Suspect	0.1	0.7	0.6	0.3	0.0	0.0	0.0	0.0	0.0
VCV	10.5	13.8	14.1	3.8	13.5	5.2	15.9	11.2	1.5

TABLE 6. Sample 1—calculation of SV.

Class name	Level no.	VCV	Calculation process	SV
Society	1	10.5	$EVCV = 10.5 + 13.8 + 14.1 + 3.8 = 42.2$ $SV = 42.2 \ast 1 + 0 \ast 0 = 42.2$	42.2
Politics	1	13.5	$EVCV = 13.5 + 5.2 + 15.9 + 11.2 + 1.5 = 47.3$ $SV = 47.3 \ast 1 + 0.0 = 47.3$	47.3
Crime	2	13.8	$EVCV = 13.8 + 14.1 + 3.8 = 31.7$ $SV = 31.7 \ast 2 + 10.5 \ast 1 + 0 \ast 0 = 73.9$	73.9
Congress	2	5.2	$EVCV = 5.2$ $SV = 5.2 \ast 2 + 13.5 \ast 1 + 0 \ast 0 = 23.9$	23.9
Government	2	15.9	$EVCV = 15.9 + 11.2 + 1.5 = 28.6$ $SV = 28.6 \ast 2 + 13.5 \ast 1 + 0 \ast 0 = 70.7$	70.7
Murder	3	14.1	$EVCV = 14.1$ $SV = 14.1 \ast 3 + 13.8 \ast 2 + 10.5 \ast 1 + 0 \ast 0 = 80.4$	80.4
Smuggle	3	3.8	$EVCV = 3.8$ $SV = 3.8 \ast 3 + 13.8 \ast 2 + 10.5 \ast 1 + 0 \ast 0 = 49.5$	49.5
Election	3	11.2	$EVCV = 11.2$ $SV = 11.2 \ast 3 + 15.9 \ast 2 + 13.5 \ast 1 + 0 \ast 0 = 78.9$	78.9
Scandal	3	1.5	$EVCV = 1.5$ $SV = 1.5 \ast 3 + 15.9 \ast 2 + 13.5 \ast 1 + 0 \ast 0 = 49.8$	49.8

in Table 5. The video classification value of each class is the sum of KCVs of each column. The VCVs are listed in the last row of Table 5. The video classification values are calculated according to the algorithm ComputeVCV().

Table 6 lists the class name, level number, video classification value, the calculation process, and the significance value for each class. Figure 7 depicts the corresponding class hierarchy. Each node in the class hierarchy corresponds to a particular class. The VCV of a class appears at the top right corner of a node. The SV appears at the lower right corner of a node. The video classification values of classes are further used to calculate the significance values. The significance value and exhaustive video classification value are calculated according to the algorithm ComputeSV().

The given video document is not only related to a murder case but also related to an election which shows totally different interests of topics. Therefore, classifying the given video document under two classes is quite reasonable. If the classification criterion compares the video classification values without considering the class-subclass relationships, the two classes chosen as the final classification results are the 'Murder' class and the 'Government' class which have the top two highest VCVs. This observation implies that the

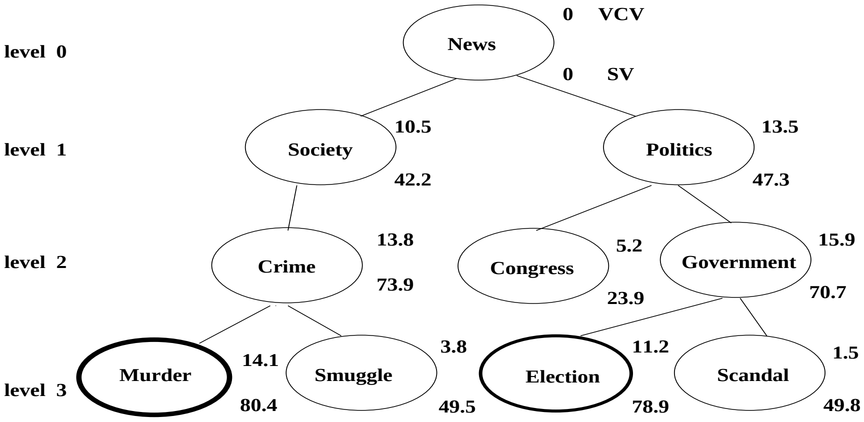

FIGURE 7. Sample 1—class hierarchy.

TABLE 7. Sample 2—VCV and SV.

Class name	Level no.	VCV	EVCV	SV
News	0	0	80.6	0
Finance	1	5.6	80.6	80.6
Stock market	2	14.4	32.3	70.2
Currency	2	13.2	13.2	32
Future	2	10.8	10.8	27.2
Policy	2	18.7	18.7	43
Option	3	3.8	3.8	45.8
Stock price	3	3.2	3.2	44
Warrant	3	6.4	6.4	53.6
Fund	3	4.5	4.5	47.9

two classes most appropriately represent the subjects of the given video document. However, if the significance value is considered as the classification criterion, the ‘Murder’ class and the ‘Election’ class are selected as the classification results. The level number of the ‘Election’ class is larger than the level number of the ‘Government’ class. In the calculation of the significance value, the specificity contributing to the SV of ‘Election’ class is larger than the exhaustivity contributing to the SV of ‘Government’ class. Thus the SV (78.9) of ‘Election’ class exceeds the SV (70.7) of ‘Government’ class, although the VCV of ‘Government’ class is larger than the VCV of ‘Election’ class.

4.2. Sample 2

Sample 2 is a video document which records a news report of ‘Comments of the president of central bank: stock market, financial policy’. In this example, we only present the final calculation results of VCVs, EVCVs and SVs. The detailed calculations of each value are omitted. Table 7 lists the class names, level numbers, VCVs, EVCVs and SVs for some classes in the class hierarchy.

The ‘Finance’ class contains several subclasses with video classification values greater than zero, such as ‘Stock Market’, ‘Currency’, ‘Future’, and ‘Policy’. This observation indicates that the video document entails several different topics. Consequently, the EVCV of ‘Finance’ class is the highest ($EVCV = 80.6$), which is also significantly larger than the VCV itself. Besides, the ‘Finance’ class has the highest significance value and, thus, is selected as the classification result. The classification result reflects that the video document entails a wide range of different topics. Therefore, in addition to considering the video classification value, the classification algorithm should also consider the distributions of VCVs between classes. In doing so, appropriate balance can be achieved between specificity and exhaustivity.

5. EXPERIMENTAL RESULTS

The performance (accuracy) of our classification algorithm is compared with manual classification by human experts. Hereinafter, this study uses the term classifier to denote the classification algorithm. Different classification criteria are implemented to evaluate the effectiveness on the performance of the classifier. Furthermore, experiments are designed to evaluate the performance of the algorithm in terms of precision, recall and F-measure. The F-measure, initially introduced by van Rijsbergen [29], is defined as

$$F = \frac{2PR}{P + R}$$

where P is the precision and R is the recall. The F-measure combines recall and precision with an equal weight. The precision and recall for each class are calculated according to the following definition. *Precision* denotes the proportion of video documents the classifier classifies under a class C that are classified under C by human experts. Meanwhile, *Recall* represents the proportion of all video documents classified under a class C by human experts that the classifier correctly classifies under C. The experiments first compute the precision and recall scores for each individual category and then average them over categories.

Four classification criteria are used to implement the classifiers. SV-classifier denotes the implementation of the classification algorithm that uses the significance value (see

TABLE 8. Simulation results for data set DS1.

Classification criterion	Precision	Recall	F-measure
SV-classifier	0.8682	0.7907	0.8276
SV–NVW-classifier	0.8019	0.7688	0.7850
VCV-classifier	0.7691	0.7380	0.7532
VCV–NVW-classifier	0.7238	0.7204	0.7221

TABLE 9. Simulation results for data set DS2.

Classification criterion	Precision	Recall	F-measure
SV-classifier	0.8298	0.7542	0.7902
SV–NVW-classifier	0.7631	0.7163	0.7389
VCV-classifier	0.8153	0.7718	0.7930
VCV–NVW-classifier	0.7751	0.7529	0.7639

Equation (6)) as the classification criterion. Meanwhile, SV-classifier classifies a video document into the class with the highest significance value. SV–NVW-classifier resembles SV-classifier, except that SV–NVW-classifier does not consider the weight of video level when computing keyword video weight. Note that keyword video weight is defined in Equation (1). VCV-classifier uses the video classification value (see Equation (4)) as the classification criterion. VCV–NVW-classifier is similar to VCV-classifier, except that the weight of video level is not considered when calculating keyword video weight.

The data set, DS1, contains 200 news items collected from the news reports. Each news item is treated as a video document. Each video document is then manually analyzed to obtain the scenes, sequences and content-bearing keywords, and experiments are conducted to classify a video document into a single class. All the video documents are manually classified and justified by four human experts, and the average of users' assigned weights is taken as the $KCW(K_p, C_i)$. A news item, covering several topics (categories), may be classified under an ancestor (higher-level) class by human experts. Such news items are called broad-coverage news items. Some of the broad-coverage news items are removed from DS1 to create another data set, DS2. DS2 contains 170 video documents. Experiments are conducted to measure the effectiveness of the classifiers on different data sets.

The simulation results are reported based on Precision/Recall and F-measure. Experimental results for the data set DS1 and DS2 are shown in Table 8 and Table 9, respectively. Both results demonstrate that all the measures (precision, recall and F-measure) of SV-classifier are higher than those of SV–NVW-classifier. Additionally, the measures of VCV-classifier are higher than those of VCV–NVW-classifier. Without considering the weight of the video-level, SV–NVW-classifier performs worse (less accurately) than SV-classifier. The same observation also applies to VCV-classifier in comparison with VCV–NVW-classifier. Such results suggest that a higher-level (e.g. sequence-level) keyword should be assigned a higher weight than a lower-level (e.g. scene-level) keyword when measuring the classification value. This phenomenon is attributed to the fact that the sequence-level keywords describe the main subjects of an entire sequence, while the scene-level keywords only describe the subjects of a scene, or just a portion of a sequence.

Table 8 also reveals that SV-classifier performs better than VCV-classifier. Notably, SV-classifier conducts the classification by considering the specificity and exhaustivity, while VCV-classifier does not consider these aspects. The data set DS1 contains some broad-coverage video documents. A broad-coverage video document is contained in several subjects (categories) and thus is classified under an ancestor class by human experts. In such cases of video documents, VCV-classifier does not perform well because it does not consider exhaustivity. As for the experiment involving data set DS2, in comparison with the experiment on data set DS1, the measures (precision, recall and F-measure) of VCV-classifier are increased, while those of SV-classifier are decreased. Importantly, data set DS1 contains more broad-coverage video documents than data set DS2. This finding implies that VCV-classifier performs better when the data set contains less broad-coverage video documents. In fact, as for the case of the experiment on data set DS2, the precision of VCV-classifier is slightly lower than that of SV-classifier, but the recall and F-measure of VCV-classifier are slightly higher than those of SV-classifier.

To conduct the experiments, the content-bearing keywords and the hierarchical structure of a video document are established manually by humans. Owing to the limitations of the manual process, the experiments only involve a set of 200 video documents. The scalability of the proposed approach for large sets of video documents requires further study, and this is proposed for a future work.

6. CONCLUSION AND FUTURE WORK

Increasing use of the World Wide Web as well as advances in video database and digital library technologies necessitate the storage of large amounts of video documents. Therefore, video documents must be categorized, to help users browse and search video documents according to their categories of interest. The proposed classification approach classifies video documents based on contents while considering the hierarchical structure of video data. The proposed approach recognizes that keywords describing the contents of video units at different levels of the hierarchical structure are of differing importance in representing the subjects of the video document. The approach also strives for an appropriate balance between specificity and exhaustivity in video classification. One interesting problem not considered herein is the effect of the addition or deletion of a class in the class hierarchy. Further work is needed to design efficient algorithms for recalculating the significance value and reclassify the video documents by considering the addition or deletion of a class.

The proposed approach can be applied to applications with video documents containing semantic descriptions and hierarchical contents structures. This work does not focus on video parsing and processing techniques for analyzing video contents [8, 10, 12, 13, 15, 16, 21, 26]. Rather, this study assumes that video documents have already been preprocessed, either by humans [22, 23, 24] or by video parsing and processing tools [8, 15, 16, 21, 26], to obtain video structures and keywords (semantic annotations) representing the contents of a video document. The assumption is a limitation of this work. However, as video technologies advance, more content processing and analysis tools will be developed to extract the contents automatically or semi-automatically. Moreover, with the increasing need to utilize video contents and the support of the multimedia content description standard, MPEG-7, more video contents will become available in MPEG-7. Further work is required to investigate automatic or semi-automatic techniques for identifying video structures and parsing video contents. In the future, we plan to enhance our approach with the access of video contents from content descriptions represented in MPEG-7. In future studies, we also plan to investigate the integration of video parsing and indexing techniques with our classification method to create an automatic or semi-automatic procedure for classifying video documents. We will integrate our method with automatic techniques to extract keywords from movie transcripts or closed-caption transcripts of news videos.

ACKNOWLEDGMENTS

The authors would like to thank the National Science Council, Taiwan, Republic of China, for partially supporting this research under Contract No. NSC 87-2213-E-009-099.

REFERENCES

[1] Chang, S. F., Smith, J. R., Beigi, M. and Benitez, A. (1997) Visual information retrieval from large distributed online repositories. *Commun. ACM*, **40**, 63–67.

[2] Lin, S. H., Shih, C. S., *et. al.* (1998) Extracting classification knowledge of internet documents with mining term associations: a semantic approach. In *Proc. 21th ACM SIGIR Conf. on Research and Development in Information Retrieval*, Melbourne, Australia, pp. 241–249.

[3] Ng, H. T., Goh, W. B. and Low, K. L. (1997) Feature selection, perceptron learning, and a usability case study for text categorization. In *Proc. 20th ACM SIGIR Conf. on Research and Development in Information Retrieval*, Philadelphia, PA, pp. 67–73.

[4] Oranger, S. (1995) The newspaper image database: empirical supported analysis of user's typology and word association clusters. In *Proc. 18th ACM SIGIR Conf. on Research and Development in Information Retrieval*, Seattle, pp. 212–218.

[5] Wong, J. W. T., Kan, W. K. and Young, G. (1996) Action: automatic classification for full-text documents. *SIGIR FORUM, ACM Special Interest Group on Information Retrieval*, **30**, 26–41.

[6] Yang, Y. and Liu, X. (1999) A re-examination of text categorization methods. In *Proc. 22th ACM SIGIR Conf. on Research and Development in Information Retrieval*, CA, pp. 42–49.

[7] Huang, J., Kumar, S. R. and Zabih, R. (1998) An automatic hierarchical image classification scheme. In *Proc. ACM Int. Conf. on Multimedia*, Bristol, UK, pp. 219–228.

[8] Smith, M. A. and Christel, M. G. (1995) Automating the creation of a digital video library. In *Proc. ACM Int. Conf. on Multimedia*, San Francisco, CA, pp. 357–358.

[9] Zhang, H., Kankanhalli, A., Smoliar, S. W. and Tan, S. Y. (1993) Automatic partitioning of full-motion video. *Multimedia Sys.*, **1**, 10–28.

[10] Mai, K., Miller, J. and Zabih, R. (1995) A feature-based algorithm for detecting and classifying scene breaks. In *Proc. ACM Int. Conf. on Multimedia*, San Francisco, CA, pp. 189–200.

[11] Nakamura, Y. and Kanade, T. (1997) Semantic analysis for video contents extraction—spotting by association in news video. In *Proc. ACM Int. Conf. on Multimedia*, Seattle, WA, pp. 393–401.

[12] Fischer, S., Lienhart, R. and Effelsberg, W. (1995) Automatic recognition of film genres. In *Proc. ACM Int. Conf. on Multimedia*, San Francisco, CA, pp. 295–304.

[13] Aoki, H., Shimotsuji, S. and Hori O. (1996) A shot classification method to select effective key-frames for video browsing. In *Proc. ACM Int. Conf. on Multimedia*, Boston, MA, pp. 1–10.

[14] Yeung, M. M., Yeo, B. L., Wolf, W. and Liu, B. (1995) Video browsing using clustering and scene transitions on compressed sequences. In *Proc. SPIE Multimedia Computing and Networking*, San Jose, CA, pp. 399–413.

[15] Zhong, D., Zhang, H. and Chang, S. F. (1996) Clustering methods for video browsing and annotation. In *Proc. SPIE Conf. on Storage and Retrieval for Still Image and Video Databases IV*, Vol. 2670, San Jose, CA, pp. 239–246.

[16] Kobla, V. and Doermann, D. (1997) Video trails: representing and visualizing structure in video sequences. In *Proc. ACM Int. Conf. on Multimedia*, Seattle, WA, pp. 335–346.

[17] Merlino, A., Morey, D. and Maybury, M. (1997) Broadcast news navigation using story segmentation. In *Proc. ACM Int. Conf. on Multimedia*, Seattle, WA, pp. 381–391.

[18] Huang, Q., Liu, Z. and Rosenberg, A. (1999) Automated semantic structure reconstruction and representation generation for broadcast news. In *Proc. SPIE Conf. on Storage and Retrieval for Still Image and Video Databases VII*, Vol. 3656, San Jose, CA, pp. 50–62.

[19] Agosti, M., Crestani, F. and Melucci, M. (1996) Design and implementation of a tool for the automatic construction of hypertexts for information retrieval. *Inf. Proc. Management*, **32**, 459–476.

[20] Brown, M. G., Foote, J. T., Jones, G. J. F., Jones, K. S. and Young, S. J. (1995) Automatic content-based retrieval of broadcast news. In *Proc. ACM Int. Conf. on Multimedia*, San Francisco, CA, pp. 35–43.

[21] Furht, B., Smoliar, S. W. and Zhang, H. (1995) *Video and Image Processing in Multimedia Systems*. Kluwer Academic Publishers, Boston, MA.

[22] Hjelsvold, R. and Midstraum, R. (1994) Modeling and querying video data. In *Proc. 20th VLDB Conf.*, Santiago, Chile, pp. 686–694.

[23] Kelly, P., Gupta, A. and Jain, R. (1996) Visual computing meets data modeling: defining objects in multi-camera video databases. In *Proc. SPIE Conf. on Storage and Retrieval for Still Image and Video Databases IV*, Vol. 2670, San Jose, CA, pp. 120–131.

[24] Rowe, L. A., Boreczky, J. S. and Eads, C. A. (1994) Indexes for user access to large video databases. In *Storage and Retrieval for Image and Video Databases II, IS&T/SPIE Symp. on Elec. Imaging Sci. and Tech.*, San Jose, CA, pp. 150–161.

[25] Wu, J. K., Narasimhalu, A. D., Mehtre, B. M., Lam, C. P. and Gao, Y. J. (1995) CORE: a content-based retrieval engine for multimedia information systems. *Multimedia Sys.*, **3**, 25–41.

[26] Zhang, H. J., Low, C. Y., Smoliar, S. W. and Wu, J. H. (1995) Video parsing, retrieval and browsing: an integrated and content-based solution. In *Proc. ACM Int. Conf. on Multimedia*, San Francisco, CA, pp. 15–24.

[27] MPEG-7 (2000) International standard ISO/IEC JTC1/SC29/WG11 N3349—overview of the MPEG-7 standard. http://www.cselt.it/mpeg/standards/mpeg-7/mpeg-7.htm

[28] Lam, W. and Ho, C. Y. (1998) Using a generalized instance set for automatic text categorization. In *Proc. 21th ACM SIGIR Conf. on Research and Development in Information Retrieval*, Melbourne, Australia, pp. 81–89.

[29] van Rijsbergen, C. J. (1979) *Information Retrieval*. Butterworths, London.