Semi-Automatic Video Content Annotation

Xingquan Zhu[1], Jianping Fan[2], Xiangyang Xue[3], Lide Wu[3], Ahmed K. Elmagarmid[1]

[1]Dept. of Computer Science, Purdue University, IN 47907, USA
[2]Dept. of Computer Science, University of North Carolina at Charlotte, NC 28223, USA
[3]Dept. of Computer Science, Fudan University, Shanghai, 200433, China
{zhuxq, ake}@cs.purdue.edu; jfan@uncc.edu; {xyxue, ldwu}@fudan.edu.cn

Abstract. Video modeling and annotating are indispensable operations necessary for creating and populating a video database. To annotate video data effectively and accurately, a video content description ontology is first proposed in this paper, we then introduce a semi-automatic annotation strategy which utilize various video processing techniques to help the annotator explore video context or scenarios for annotation. Moreover, a video scene detection algorithm which joints visual and semantics is proposed to visualize and refine the annotation results. With the proposed strategy, a more reliable and efficient video content description could be achieved. It is better than manual manner in terms of efficiency, and better than automatic scheme in terms of accuracy.

1. Introduction

In recent, advances in computer hardware and networks have made significant progress in the developments of application systems supporting video data. Large scale of video archive is now available to users as various forms. However, without an efficient and reasonable mechanism for retrieving video data, large archive of video data remains as merely unmanageable resources of data. Accordingly, various video index strategies are proposed to describe video content by: (1) *High level indexing*; (2) *Low level indexing*; and (3) *Domain specific indexing*.

Due to the inadequacy of textual terms in describing video content, many low-level indexing strategies have emerged [1][7] to parse video content. Unfortunately, all these strategies alone do not enable a sufficiently detailed representation of video content. Hence, manual annotation is still widely used.

The simplest way to model video content is using free text to manually annotate each shot separately. However, since a single shot is separated from its context, the video scenario information is lost. Accordingly, *Aguierre Smith et. al* [2] implements a video annotation system using the concept of stratification to assign description to video footage. Based on this scheme, the *video algebra* [3] is developed to provide operations for the composition, search, navigation and playback of digital video presentation. A similar strategy for evolving documentary presentation is found in [4]. Instead of using textual terms for annotation, *Davis et. al* [5] presents an iconic visual language based system, however, this user-friendly approach is limited by a fixed vocabulary. Obviously, no matter how efficient a content description structure is,

annotating videos frame by frame is still a time consuming operation. Hereby, a shot based semi-automatic annotation engine is proposed [6], unfortunately, annotators also have to explore scenarios by browsing shots sequentially. And the problems still remain: (1) no efficient scheme has been developed to explore video scenarios for annotation; (2) keywords at various levels should be organized differently; (3) to minimize annotators' subjectivity and influences of synonymy and polysemy in unstructured keywords, ontologies have been proved to be an efficient way; However, methods above either fail to define their ontology explicitly or do not separate ontology with annotation data to enhance the reusability of annotation data.

To address these problems, a semi-automatic video annotation scheme is proposed in this paper. We first define the content description ontology. Then, video group detection strategy is introduced to help annotators explore video context and scenarios. Based on acquired video group information, the annotator could execute extensive operations to improve annotation efficiency.

2. Video Content Description Architecture

2.1. Video Content Description Ontology

As we know, most videos can be represented by using a hierarchy consisting of five layers (video, scene, group, shots and frames), from top to bottom in increasing granularity for content expression. A flexible and comprehensive content annotation strategy should also describe video content at different layers and with different granularities. Hence, video content description ontology is predefined, as shown in Fig. 1, where four content descriptions, *Video Description* (*VD*), *Group Description* (*GD*), *Shot Description* (*SD*) and *Frame Description* (*FD*), are used to describe video content. They are defined as below:

1. The *VD* addresses the category and specialty taxonomy information of the entire video. There are two descriptors (Video category and Speciality category) contained in *VD*. The description at this level should answer questions like "What does the video talk about?"
2. The *GD* describes the event information in a group of adjacent shots that convey the same semantic information. There are two descriptors (Event and Actor) specified in *GD*. The description at this level should answer the query like " Give me all surgery units among the medical videos?"
3. The *SD* describes the action in a single shot. This action could be a part of an event. e.g., a video shot could show the action "doctor shake hands with patient" in a diagnosis event. There are three descriptors (Object, Action and Location) specified in *SD*. Hence, the *SD* should answer the query like "Give me all units where a doctor touches the head of the patient on the bed".
4. At the lowest level, the frame, the description should address the details of objects in frame(s). There are two descriptors (Object and Status) specified in *FD*. The description should answer query like "What is in the frame(s)?"

The keyword tables of various descriptors are predefined and are still extensible for annotators by adding more instances.

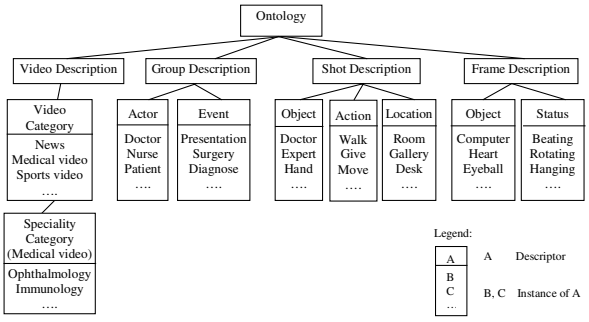

Fig. 1. Video content description ontology

2.2. Shot Based Video Temporal Description Data Organization

To separate the ontology from the description data and integrate video semantics with low-level features, a shot based video description data structure is constructed for each video. Given any video shot S_i, assuming *KA* indicates the *Keyword Aggregation* (*KA*) of all descriptors in the ontology, then $KA=\{VD_l,l=1,\ldots,NV_i;\ GD_l,l=1,..NG_i;\ SD_l\ l=1,..NS_i;\ FD_l\ l=1,..NF_i\ \}$, where VD_l, GD_l, SD_l and FD_l represent the keywords of *VD*, *GD*, *SD* and *FD* respectively, and NV_i , NG_i, NS_i and NF_i indicate the number of keywords for each description. To indicate the region where each keyword takes effect, the symbol v_{a-b}^{ID} is used to denote the region from frame *a* to *b* in the video with a certain identification (*ID*). The *Temporal Description Data* (*TDD*) for shot S_i is then defined as the aggregation of mappings between annotation ontology and temporal frames: $TDD=\{ S_i^{ID},\ S_i^{ST},\ S_i^{ED}, Map(KA, V)\}$, where S_i^{ST} and S_i^{ED} denote the start and end frame of S_i respectively. *KA* indicates the keyword aggregation of all descriptors, *V* indicates a set of video streams, $v_{a-b}^{ID} \in V, ID = 1,..,n$, and *Map* defines the correspondence between annotations and the video temporal information. E.g., $Map(KA_i; v_{a-b}^{ID})$ denotes the mapping between keyword KA_i to region from frame *a* to *b* in video with certain identification *ID*. The advantage of above mapping is that ontology is separated from annotation data. The same video data could be shared and annotated by different annotators for different purposes, and can be easily reused for different applications.

The assembling of *TDD* from all shots forms the *Temporal Description Stream* (*TDS*) of the video. It indicates that all annotation keywords are associated with each shot. The reason we utilize such a data structure is clarified below:

1. A frame based data description structure will inevitably incur large redundancy.
2. Since video shots are usually taken as the basic unit of video processing techniques [1][7][9] the shot based structure will help us integrate low-level features with semantics seamlessly.
3. More keywords can be employed in the *FD* to characterize the changing of shot content. Hence, the details of the video will not be lost.

3. Video Content Annotation

3.1. Video Group Detection

The video shot is a physical unit, it is incapable of conveying independent semantic information. Hence, various approaches have been proposed to determine video units that convey relatively higher level scenario information [9]. In our system, a temporally constrained strategy is employed to merge temporally or spatially correlated shots into groups, as shown in Fig. 2, the details could be found in [9].

Fig. 2. Group detection results with each row denoting one group

3.2. Unified Similarity Evaluation

In Section 2, we specified that the mapping of each keyword has recorded the frame region where this keyword takes effect. To evaluate the semantic similarity between video shots, this region should be considered since it determines the importance of the keyword in describing the shot content. For *VD*, *GD*, and *SD*, keywords at these levels will have longer (or equal) duration than the current shot. Hence, they will be in effect over the entire shot. However, descriptors in the *FD* may last only one or several frames, to calculate the semantic similarity between shots, the *Effect Factor* of each *FD* descriptor's keyword is calculated first.

Assuming FD_k denotes the k^{th} keyword of *FD,* we suppose there are *N* mappings associated with FD_k in shot S_i, and the mapping regions are $v^2_{a_1-b_1},\ldots,v^2_{a_N-b_N}$. Given any two regions $v^2_{a_i-b_i}$, $v^2_{a_j-b_j}$ $(i\neq j,i,j\in N)$ among these mappings, assume operator $\Theta(X,Y)$ denotes the number of overlapped frames between region *X* and *Y*. Then, the *Effect Factor* of keyword FD_k corresponding to shot S_i is defined by Eq. (1).

$$EF(FD_k,S_i)=\frac{\sum_{l=1}^{N}(b_l-a_l)-\sum_{m=1}^{N}\sum_{n=m}^{N}\Theta(v^{ID}_{a_m-b_m},v^{ID}_{a_n-b_n})}{S_i^{ED}-S_i^{ST}},\quad m,n\in N \tag{1}$$

To evaluate the cross intersection between keywords at various levels, we define $\overline{VDS}_k$, $\overline{GDS}_k$, $\overline{SDS}_k$, $\overline{FDS}_k$ as the aggregation of keywords which have been used to annotate shot S_k in *VD*, *GD*, *SD* and *FD* respectively. To describe the relationship among series of keywords (X_1, X_2, .., X_N), three operators $\{\Omega(X_1,X_2,..,X_N),\ \vartheta(X_1,X_2,..,X_N),\ \Psi(X)\}$ are defined:

1. $\Omega(X_1,X_2,..,X_N)=\{X_1\cup X_2\cup..\cup X_N\}$ indicates the union of X_1, X_2, .., X_N.
2. $\vartheta(X_1,X_2,..,X_N)=\{X_1\cap X_2\cap..\cap X_N\}$ is the intersection of X_1, X_2,.., X_N.

3. $\Psi(X)$ represents the number of keywords in X.

Given any two shots S_i and S_j, assume their *TDD* are TDD_i={ S_i^{ID}, S_i^{ST}, S_i^{ED}, *Map(KA, V)*} and TDD_j={ S_j^{ID}, S_j^{ST}, S_j^{ED}, *Map(KA, V)*} respectively. Assume also that KAS_i denotes the union of keywords which have been shown in annotating shot S_i. The semantic similarity between S_i and S_j is then defined by Eq. (2):

$$SemStSim(S_i,S_j)=W_V\frac{\Psi(\vartheta(\overline{VDS}_i,\overline{VDS}_j))}{\Psi(\Omega(\overline{VDS}_i,\overline{VDS}_j))}+W_G\frac{\Psi(\vartheta(\overline{GDS}_i,\overline{GDS}_j))}{\Psi(\Omega(\overline{GDS}_i,\overline{GDS}_j))}+W_S\frac{\Psi(\vartheta(\overline{SDS}_i,\overline{SDS}_j))}{\Psi(\Omega(\overline{SDS}_i,\overline{SDS}_j))}+W_F\frac{\sum_k\{EF(FD_k,S_i)\cdot EF(FD_k,S_j)\}}{FD_k\in\vartheta(\overline{FDS}_i,\overline{FDS}_j)}}{\Psi(\Omega(\overline{FDS}_i,\overline{FDS}_j))} \quad (\mathbf{2})$$

Eq. (2) indicates that the semantic similarity between S_i and S_j is the weighted sum of the cross intersection of keywords at various video content levels.

Based on the semantic similarity in Eq. (2) the overall similarity between S_i and S_j which joint visual features and semantics is given by Eq. (3).

$$StSim(S_i,S_j)=(1-\alpha)\cdot VisStSim(S_i,S_j)+\alpha\cdot SemStSim(S_i,S_j) \quad (\mathbf{3})$$

where $VisStSim(S_i,S_j)$ indicates the visual similarity between shots which is specified in [13]. $\alpha\in[0,1]$ is the weight of the semantic information in similarity measurement, which can be specified by users. Based on Eq. (3), given shot S_i and video group G_j, their similarity can be calculated using Eq. (4).

$$StGpSim(S_i,G_j)=Max_{S_j\in G_j}\{StSim(S_i,S_j)\} \quad (\mathbf{4})$$

Given group G_i and G_j, assume $\hat{G}_{i,j}$ is the group containing less shot, and $\tilde{G}_{i,j}$ is the other group. $M(X)$ denotes the number of shot in X, then, the similarity between G_i and G_j is given in Eq.(5), with more techniques described in [9].

$$GroupSim(G_i,G_j)=\frac{1}{M(\hat{G}_{i,j})}\sum_{i=1;S_i\in\hat{G}_{i,j}}^{M(\hat{G}_{i,j})}StGpSim(S_i,\tilde{G}_{i,j}) \quad (\mathbf{5})$$

3.3. Video Scene Detection

After most video groups haven been annotated, we can integrate semantics and visual features to merge similar groups into semantically related units (scenes). And use them to help the annotator visualize and refine annotation results. To attain this goal, the scene detection strategy takes steps below:

1. Given any group G_i, assume GDE_i denotes the aggregation of the event descriptor's keyword which has been used in *GD* of all shots in G_i.
2. For any neighboring groups G_i and G_j, if $\vartheta(GDE_i,GDE_j)=\varnothing$, these two groups are not merged. Otherwise, go to step 3. I.e., if the event descriptor in two groups is totally different, they cannot be merged into one group.
3. Using Eq. (5) to calculate overall similarity between these two groups; go to step 2 to find all other neighboring groups' similarity. Then go to step 4.
4. Adjacent groups with similarity larger than T_G are merged into a new group. Those reserved and newly generated groups are formed as video scenes.

3.4. Semi-Automatic Video Annotation

Some semi-automatic annotation schemes have been implemented in image database [8] by using semantics, visual features and relevance feedback to assist the annotator for annotation. Derived from the same intuition, in this section, a semi-automatic video annotation scheme is presented.

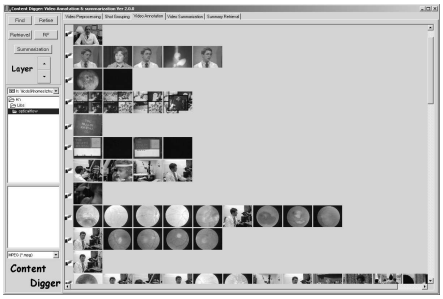

Fig. 3. Video content annotation interface

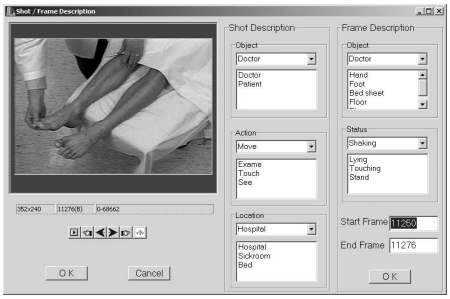

Fig. 4. Shot and frame annotation interface

As the first step, the group detection method is applied to segment temporally or spatially related shots into groups. Then, the groups are shown sequentially for annotation, as shown in Fig.3. Given any group, the annotator has three operations:

1. Annotate a certain shot by double clicking the key-frame of the shot (the result is illustrated in Fig.4.). A series of function buttons such as play, pause, etc. are available to help the annotator determine semantics among the shot and frames.

2. If the annotator thinks that the current group belongs to the same event category, he (she) could specify *GD* and *VD* keywords to the group by clicking the hand-like icon at the left of the group, and select keywords to annotate the group.

3. If the annotator thinks current group contains more than one event category, he (she) can manually separate it into different groups (with each group belonging to only one event category) by dragging the mouse to mask shots in the same event category and click the hand-like icon to assign keywords.

At any annotation state, the annotator can select one or a group of shots as the query to find similar groups for annotation. To do this, the relevance feedback (*RF*) strategy is activated:

1. All selected shot(s) are treated as a video group. The annotator should input keywords to describe them before the retrieval.
2. After the annotator clicks the "Find" button, the similarity evaluation strategy in Eq. (5) is used to find similar groups.
3. At any retrieval stage, the annotator can either annotate retrieved groups separately or mark some of them as feedback examples, and click "RF" button to trigger a *RF* processing. Then, all selected shots are annotated with keywords specified in step 1. The Eq. (6) is used to find other similar groups.

Eq. (6) presents the simplified *RF* model in our system (based on Bayesian formula). Assuming G_i denotes the selected feedback examples in current iteration, for any group G_j in the database, its global similarity $Sim(j)^k$ in the current iteration (k) is determined by its global similarity in the previous iteration $Sim(j)^{k-1}$ and its similarity

to current selected feedback examples $GroupSim(G_i,G_j)$. η indicates the influence of the history to the current evaluation, in our system we set $\eta=0.3$.

$$Sim(j)^k = \eta Sim(j)^{k-1} + (1-\eta)GroupSim(G_i, G_j) \quad (6)$$

By integrating the annotated semantics and visual features related to groups, we can merge the semantically related adjacent groups into scenes to help annotators evaluate and refine annotations results:

1. At any annotation stage, the annotator can click the "Refine" button, the scene detection strategy is invoked to merge adjacent similar groups into scenes.
2. The annotator can specify different values for α to evaluate annotation quality in different situations.

That is, a series of annotation, refinement, annotation, can be recursively executed until a satisfactory result is achieved.

4. Experimental Results

Obviously, the performance of two techniques, video group detection and group similarity assessment, should be evaluated to confirm the efficiency of the proposed semi-automatic annotation strategy. Due to lack of space, we supply only group similarity assessment result; the group detection results could be found in [9]. About 8 hours of medical videos and 4 hours of News programs are used as our test bed. They are first parsed with the shot segmentation algorithm to detect the gradual and break changes [7]. After group detection has been executed on each video, we manually select out groups which have distinct semantic meaning as the test bed, then randomly select one group as the query, all retrieved top-*N* groups are utilized to evaluate the performance of our group similarity assessment. The results are shown in table 1, with *PR* and *PE* define by Eq.(7).

$$PR = SG / N; \qquad PE = SG / AG \quad (7)$$

Where *AG* denotes the number of groups in current video which are similar with the query group; *SG* indicates the number of groups in top-*N* retrieved results (we use top-5 return results, thus *N*=5 in our experiment) which are similar with the query.

Table 1. Group similarity evaluation performance (*Top-5*)

Videos	$\alpha = 0.0$		$\alpha = 0.3$		$\alpha = 0.5$	
type	*PR*	*PE*	*PR*	*PE*	*PR*	*PE*
Medicals	0.68	0.71	0.81	0.92	0.72	0.76
News	0.64	0.76	0.79	0.84	0.70	0.73

Table 1 demonstrates that the proposed video group similarity evaluation strategy could be efficiently utilized to help the annotator find interesting video groups. In average, about 65% similar video groups could be retrieved out with only visual features. By considering semantics, over 80% of similar groups could be retrieved out. However, while α goes higher (e.g. α=0.5), the semantics play a more important role

in similarity evaluation, accordingly, the retrieval results trend to be consisted with semantically related groups (may not be visually similar).

5. Conclusion

Due to the obvious shortcoming of traditional video annotation strategy, we propose a semi-automatic video annotation framework that employs general video processing techniques to improve the annotation efficiency. We first propose an ontology to describe video content at various levels and with different granularities. Then, the video group detection strategy is utilized to help the annotator explore the video scenario information for annotation. Afterward, the relevance feedback technique and unified video group similarity evaluation scheme are employed to help annotators find the interesting video groups for annotation or visualize the video annotation results. The proposed semi-automatic strategy is better than manual manner in terms of efficiency, and better than automatic scheme in terms of accuracy.

Acknowledgement: Jianping Fan was supported by NSF under contract IIS0208539, Xiangyang Xue was supported by NSF of China under contract 60003017, Chinese National 863 project under contract 2001AA114120, Lide Wu was supported by NSF of China under contract 69935010.

References

1. S.W. Smoliar; H.J. Zhang, "Content based video indexing and retrieval", *IEEE Multimedia, 1(2), 1994.*
2. T.G. Aguierre Smith and G. Davenport. "The Stratification System: A Design Environment for Random Access Video". *In 3rd Int'l Workshop on Network and Operating System Support for Digital Audio and Video, 1992*.
3. R. Weiss, A. Duda, and D. Gifford, " Content-based access to algebraic video", *In IEEE Int'l Conf. on Multimedia Computing and Systems, pp. 140-151, Boston, USA, 1994*.
4. G. Davenport, M. Murtaugh, "Context: Towards the evolving documentary", *In Proceeding of ACMM Multimedia conference, San Francisco, Nov., 1995*
5. Marc Davis, "Media streams: An iconic visual language for video annotation*", In IEEE Symposium on Visual Language, pp.196-202, 1993*.
6. M. Carrer, L. Ligresti, G. Ahanger, T. Little, "An annotation engine for supporting video database population", *Multimedia tools and applications, vol. 5, pp.233-258, 1997*.
7. J. Fan, W. Aref, A. Elmagarmid, M. Hacid, M. Marzouk, X. Zhu, "MultiView: Multilevel video content representation and retrieval", *Journal of Electronic imaging, 10 (4), 2001*.
8. X. Zhu, H. Zhang, Liu W., C. Hu, L. Wu, "A new query refinement and semantics integrated image retrieval system with semi-automatic annotation scheme", *Journal of Electronic Imaging, 10 (4). pp.850-860, October 2001*.
9. X. Zhu, J. Fan, W. Aref, A. Elmagarmid, "ClassMiner: Mining medical video content structure and events towards efficient access and scalable skimming", *In Proc. of ACM SIGMOD Workshop on Data Mining and Knowledge Discovery, June, WI, 2002*.