A PEOPLE SIMILARITY BASED APPROACH TO VIDEO INDEXING

Peng Wang[1]*, Yu-Fei Ma*[2]*, Hong-Jiang Zhang*[2] *and Shiqiang Yang*[1]

[1] Department of Computer Science and
Technology, Tsinghua University,
Beijing 100084, China

[2] Microsoft Research Asia
3/F, Beijing Sigma Center, 49 Zhichun Road,
Beijing 100080, China

ABSTRACT

This paper presents a new approach to people-based video indexing. In this approach, we define a people-based similarity measure according to both clothing similarity and speaking voice similarity. Such similarity depicts how perceptually similar two people appearing in different scenes are and if they belong to an identical person. Instead of computing in feature space, the proposed people-based similarity is computed in distance space. The extended Support Vector Machines (SVMs) are employed to map a serial of low-level feature distances to a perceived people similarity. In order to build people-based video indexing, a novel unsupervised clustering algorithm is also proposed, which can more correctly identify individual person according to mutual people similarities between two people. The experiments on large video testing data have demonstrated the effectiveness and efficiency of the proposed people-based similarity, unsupervised clustering and video indexing.

1. INTRODUCTION

Video indexing is one of the important techniques for effectively accessing video data, such as browsing and retrieval. Tremendous efforts have been made to analyze video content, and build all kinds of indexes. In [1], a soccer video analysis approach was proposed to classify video clips into three views and obtain play/break status of the game via heuristic rules. Huang etc. [2] proposed a method to generate hierarchical index table of video for non-linear browsing. Recently, human-machine interaction techniques have been utilized to model the semantic concepts. Naphade & Huang [3] proposed a multiject-multinet framework for semantic video indexing by fusing visual, audio and caption information. The work in [4] introduced the Semantic Visual Template (SVT) which is generated by a two-way interaction between user and computer system. However, current works on semantic video indexing are either confined to specific domain [1, 2] or dependent on user interaction to improve performance [3, 4].

In this paper, we propose a people-based video indexing, in which a novel people-based similarity is also defined. Such video indexing provides users much more high-level information, so it facilitates user to more effectively access video contents. The similarity measure is one of the key issues in video indexing. Fablet & Bouthemy [5] proposed a statistical motion-based similarity measure for video indexing and retrieval. In [6], a two-level hierarchical clustering approach was employed to group shots with similar color and motion features. However, these low-level feature based similarities are lack of semantic information. Because audiences often pay more attention to the human activities, people-based video indexing is definitely useful for video browsing and retrieval. Unfortunately, it had not been deeply investigated in previous works.

The people similarity we defined is based on two basic similarities: clothing similarity and speaking voice similarity. In this work, we do not directly use face features in people similarity measure, though the face detection algorithm is employed to locate clothes area. Experimental results indicate that face features would make people similarity unstable, as current face recognition technologies cannot handle such cases in live videos as the large variance of illumination, all kinds of face poses and expressions. On the other hand, face detection would fail in small faces, non-frontal faces and the faces in dark scene. However, the important faces usually do not appear in these cases. Therefore, the proposed solution is reasonable and practicable. In order to make people-based similarity measure consist with human perception, the extended Support Vector Machines are used to generate perceived similarity from low-level feature distances. Additionally, we also proposed a novel unsupervised clustering algorithm that is used to identify individual person based on the mutual people similarities. There are two major reasons why the other conventional clustering methods, such as k-means, are not used. One is the clustering is carried out in distance space, but not in feature space. The other is the people similarity we defined doesn't fulfill triangle relationship in distance space. The proposed people-based video indexing has been validated by the experiments on large-scale living video programs.

The rest of this paper is organized as follows. In Section 2, we describe the definition of people-based similarity. In Section 3, how to build people-based video indexing is discussed in details. The experimental results are shown in Section 4. Finally, we conclude the whole paper in Section 5.

2. PEOPLE-BASED SIMILARITY

People-based similarity computation consists of two steps. Firstly, the low-level visual and auditory distances are computed respectively. There are total five visual distances based on the clothes area and an auditory distance based on speaker voice extracted. Secondly, we map low-level distances to Perceived People Similarity ($PPS \in [0, 1]$) by employing SVMs [7] tool and its probabilistic outputs theory in [8].

2.1. Low-Level Distance Computation

Clothes area covers the neck-down clothing-part of a person which is located according to the face detected. In our approach, we adopt a multi-view face detector [9] which is able to detect both position and pose of human face. The detected poses include the face views from left side to right side denoted by {−3, −2, −1, 0, 1, 2, 3}. If let the detected face position be ((X_1, Y_1), W_1, H_1), where (X_1, Y_1) is the rectangular center coordinate, W_1 and H_1 are the width and height of the rectangle, we estimate the clothes area ((X_2, Y_2), W_2, H_2) by (1).

$$\begin{cases} X_2 = X_1 + p \cdot 0.2W_1 \quad , \ p \in \{-3,-2,-1,0,1,2,3\} \\ Y_2 = Y_1 + H_1 \\ W_2 = 3W_1 \\ H_2 = H_1 \end{cases} \tag{1}$$

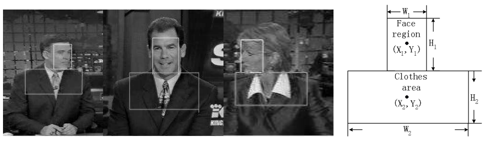


Figure 1 Clothes area estimation based on face detection

Figure 1 visually illustrates how to estimate clothes area based on face detection. From clothes area, 3 visual features are extracted, including Color Histogram, Color Auto-Correlogram [10] and Color Center Position [11]. First, three color histograms are extracted from each channel of YC_bC_r color space. Each color histogram has 64 bins. We employ linear correlation *Corr* (2) to compute the histogram distances which are denoted by D_1, D_2 and D_3.

$$Corr(\mu,\nu) = \frac{\sum_{i=1}^{N}(\nu_i - \bar{\nu})(\mu_i - \bar{\mu})}{\sqrt{\sum_{i=1}^{N}(\nu_i - \bar{\nu})^2}\sqrt{\sum_{i=1}^{N}(\mu_i - \bar{\mu})^2}} \tag{2}$$

In (2), (μ,ν) denote the two compared histograms, ($\bar{\mu},\bar{\nu}$) are the mean value of (μ,ν), and N is dimension of μ,ν.

As color histogram lacks of spatial information, we extract Color Auto-Correlogram (CAC) in RGB color space as well, which describes the spatial self-correlation of color. The RGB color space is quantized into 4×4×4 bins to generate a color histogram. Let k be the L_1 distance between two compared pixels p_1 and p_2. Probability for color C_i (C_i denotes the quantized color in histogram) with distance k is defined as:

$$\gamma_{C_i}^{(k)} \equiv P_r[L_1 \mid p_1 - p_2 \mid = k, p_2 \in C_i \mid p_1 \in C_i] \tag{3}$$

where k = 1, 3, 5, 7. So the dimension of the extracted CAC is 64 ×4. The CAC distance D_4 is also computed by linear correlation (2).

In order to capture more spatial features, we also extract Color Center Position (CCP) in HSV color space. The HSV space is quantized into 36 bins first. Then, the CCP of color C_i is denoted as $(\overline{x_{C_i}}, \overline{y_{C_i}})$, that is, the center of the pixels with color C_i. $\overline{x_{C_i}}$ and $\overline{y_{C_i}}$ are normalized into[0,1] by image width and height respectively. Consequently, the CCP feature is represented by vector $P = [\overline{x_1}, \overline{y_1}, \overline{x_2}, \overline{y_2}, \cdots, \overline{x_{36}}, \overline{y_{36}}]$, whose distance D_5 is computed by L_1 distance.

Speaking voice features are important characters of human. So we extract a serial of features of speaker voice by adopting the approach proposed in [12]. The feature set includes 10-dimensional *LSP*, 8-dimensional *MFCC*, 1-dimensional *CMS* and 1-dimensional pitch information. Then, the speaker voice is modeled by Gaussian Mixture Model (GMM-32), $G_i \sim N(\boldsymbol{\mu}, \boldsymbol{C})$, where $\boldsymbol{\mu}$ is mean vector and $\boldsymbol{C}$ is covariance matrix. Thus, the voice features are represented by 32 diagonal elements of matrix $\boldsymbol{C}$, which is denoted by $V = [c_1^1, c_1^2, \cdots, c_1^{20}, c_2^1, c_2^2, \cdots, c_2^{20}, \cdots, c_{32}^{20}]$. *K-L* distance is employed to calculate auditory distance D_6.

In this way, we obtain a 6-dimensional low-level distance vector:

$$D = [D_1, D_2, D_3, D_4, D_5, D_6] \tag{4}$$

where each distance component D_i is normalized into [0, 1].

2.2. Perceived People Similarity Computation

How to map the distance vector to perceived people-based similarity is a challenging issue. We employ SVMs tool to deal with this problem due to its high performance in classification. The Perceived People Similarity (*PPS*∈[0, 1]) is computed in two steps: 1) A discrete classification result is estimated from the standard outputs of SVMs, which is named Near People Similarity (*NPS*) in this paper. However, there must be errors in this classification process. 2) Therefore, the probabilistic outputs theory of SVMs in [8] is adopted to generate an actual similarity, that is, *PPS*.

First, the 6-dimensional distance vector D is treated as the input vector of SVMs, in which Gaussian Radial Basis function (RBF) is chosen as kernel function. When training a SVM model, two classes of data: $(\boldsymbol{x_1}:y_1),\cdots,(\boldsymbol{x_n}:y_n)$ are employed. $\boldsymbol{x_i} \in R^6$ denotes distance vector, and $y_i \in \{+1, -1\}$ is class label. $y_i = +1$ means two people are identical, while $y_i = -1$ means two people are different. If the standard output of SVMs, $NPS > 0$, +1 is assigned to y_i; if $NPS < 0$, −1 is assigned to y_i.

In the second step, the SVM + Sigmoid method [8] is imposed to fit the relationship between standard SVM outputs (*NPS*) and the probabilistic outputs $P_r(class = +1 \mid NPS)$, that is, *PPS*, by finding proper parameters A and B as follows:

$$PPS = P_r(class = +1 \mid NPS) = \frac{1}{1 + \exp(A \times NPS + B)} \tag{5}$$

This function means that if *PPS* approaches 1, then the reliability of *NPS*>0 is high; while if *PPS* approaches 0, then the reliability of *NPS*<0 is high.

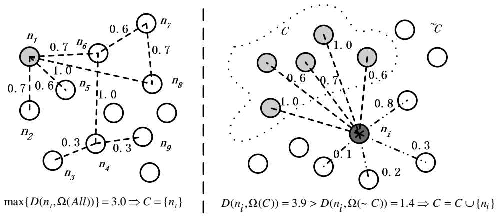
3. PEOPLE-BASED VIDEO INDEXING

To identify people in video and organize the scenes where they appear will dramatically facilitate users to browse video content, especially for the video containing many human activities. So, we may build video index according to people appearance. The people-based similarity provides a good measure to identify each person in video. With such similarity, the video index is generated as follows. First, face detection algorithm is employed to obtain people candidates in each shot. We assume that the size and position of an identical face could not change largely in consecutive frames. Subsequently, the visual features of people candidates are extracted from key-frame, and the auditory features are extracted from audio track around key-frame. If there are N people candidates, total N×(N-1)/2 *PPSs* would be computed. In order to group the scenes where the same people appear and verify the result of human identification from *PPS*, we design an unsupervised clustering algorithm to group all the candidates belonging to an identical person. The clustering process takes advantage of the repeated appearance of the same people to enhance the accuracy of identification based on *PPS*.

3.1. Unsupervised Clustering Algorithm

Assuming there are *N* people, we map them to *N* nodes ($p_i \rightarrow n_i$, $i = 1,2,\cdots, N$). The *PPS* between two people p_i and p_j is used as the weight of edge between two nodes n_i and n_j. The following truncation function (6) is used to reduce the number of initialized edges. In our experiments, α and β are set to 0.4 and 0.7 respectively.

$$F(PPS) = \begin{cases} 0 & PPS \in [0,\alpha) \\ PPS & PPS \in [\alpha,\beta) \\ 1 & PPS \in [\beta,1] \end{cases} \quad (6)$$



n_1
0.7
n_6
0.6
n_7
0.7
1.0
0.7
0.6
n_5
n_8
1.0
n_2
0.3
0.3
n_9
n_3
n_4
$\max\{D(n_1,\Omega(All))\} = 3.0 \Rightarrow C = \{n_1\}$
C
~C
1.0
0.6
0.7
0.6
1.0
0.8
n_i
0.3
0.1
0.2
$D(n_i,\Omega(C)) = 3.9 > D(n_i,\Omega(\sim C)) = 1.4 \Rightarrow C = C \cup \{n_i\}$

Figure 2 Unsupervised clustering in distance space

As shown in Figure 2, we define an operator $D(n_i,\Omega)$, where n_i is a node, Ω is a node set and $D(n_i,\Omega)$ is the sum of the weights of edges between n_i and all the nodes in Ω. There are three kinds of node sets: $\Omega(All)$, $\Omega(C)$ and $\Omega(\sim C)$, where *All* is ensemble set, *C* is subset and *~C* is complementary set of *C*. Define *Node*(*C*) is the number of nodes in subset *C*. The clustering process is:

- *Step-1*: If *All* is empty, stop clustering; else go to *Step*-2.
- *Step-2*: For each node $n_k \in All$, compute $D(n_k,\Omega(All))$.
- *Step-3*: Compute $\hat{D}$, where $\hat{D} = \max\{D(n_k,\Omega(All))\}$. Compute $\hat{n}$, where $\hat{n} = \underset{n_k \in All}{\operatorname{argmax}}\{D(n_k,\Omega(All))\}$.
- *Step-4*: If $\hat{D} = 0$, stop clustering; else set subset $C = \{\hat{n}\}$.
- *Step-5*: For each $n_k \in All - C$, compute $D(n_k,\Omega(C))$ and $D(n_k,\Omega(\sim C))$.
- *Step-6*: If condition $D(n_k,\Omega(C)) > \mu \cdot D(n_k,\sim\Omega((C))$ or $D(n_k, \Omega(C)) > \theta \cdot Node(C)$ is satisfied, $C = C \cup \{n_k\}$; else go to *Step-7*. Where μ and θ are predefined parameters which control the aggregation of subset *C*.
- *Step-7*: Repeat *Step-5* & *Step-6* until there are not any nodes can be included into *C*. Go to *Step-8*.
- *Step-8*: Mark all the nodes in subset *C* as a group, and set $All = All - C$. Go to *Step-1*.

In Figure 2, the left half shows how we choose an initial node in subset *C* as *Step-2* & *Step-4* described. The right half shows the criterion of how to choose a node to add into *C*. In our experiments, parameters μ and θ were set 1.0 and 0.5 respectively. Some errors in the original *NPS* based human classification are corrected by this clustering process as well as computing *PPS*.

3.2. People-based Video Indexing

After people candidates are clustered into groups, each group corresponding to an identical person, we sort people by their appearance times and total duration. The people ranked in front implies that they appear more frequently and more important than others. The shots where these people appear are also listed temporally. So, we define a tree-based index structure in which each node represents an identical person and sub-nodes represent a series of shots where this person appears. With the guidance of such people-based indexing, users may have a quick overview of how many people there are, who are more important than others, what activities these people involve.

Figure 3 shows the proposed tree-based indexing structure. There are total *n* people in this index. Each person node is followed by N_i shots along the timeline, which show the activities he/she involving. At the top of Figure 3, an example of person node is given, which is extracted from movie “Pride and Prejudice”. The lady shown in the person node is the primary actress and the scenes where she appears are displayed temporally in sub-nodes.

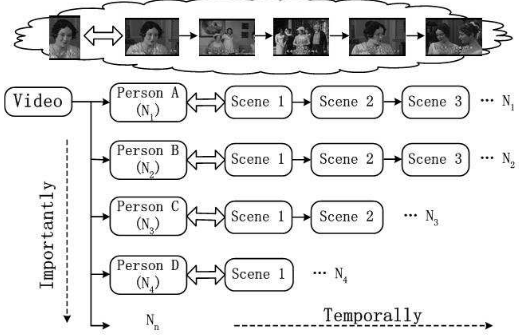


Figure 3 People based video indexing

4. EXPERIMENTS AND DISCUSSION

We chose five video programs to test the proposed approach, including two news videos (N), one home video (H), one talk show (T) video and one movie (M). The detailed information is given in Table 1. We have labeled ground truth for all the five videos, and V_1 was used as the training data of SVMs.

Table 1 Testing Videos

Video	V_1	V_2	V_3	V_4	V_5
Genre	N	N	H	T	M
Length	28:25	30:27	58:33	30:08	14:16
Shot	471	386	472	476	118
Candidate	54	56	29	96	43

First, we compared people pair classification results based on *NPS* (*PC-NPS*) and that based on clustering (*PC-Cluster*). The former is the basic classification results of SVMs, while the latter is the refined results with clustering process. The performance is evaluated by classification error rate (R_{Err}), which is defined as:

$$R_{Err} = [(NoPP - Cor.)/NoPP] \times 100\% \quad (7)$$

where *Cor.* denotes correct judgments of people candidate pairs and *NoPP* is total number of pairs. The average error rate of *PC-NPS* is 3.97%. We find that most of false alarms are caused by noisy clothes area. For instance, the result of *PC-NPS* considers a man who raises a red board in front is identical to a girl who wears a red T-shirt. Sometimes, background is included in the located clothes area or only face is shown in video. These cases will affect the performance of people-based similarity measurement.

The unsupervised clustering process is designed to enhance the performance of *PC-NPS*. The results are also shown in Table 2. We can see that after clustering, average error rate is effectively reduced to 2.74%. Therefore, the average error rate reduction ($R_{Err}Reduced$) is over 36%. It indicates that the performance becomes much more robust, because the clustering process takes advantage of multiple people pairs to make further verification.

Table 2 *PC-NPS* vs. *PC-Cluster*

Video	V_2	V_3	V_4	V_5	*Avg.*
Candidate	56	29	96	43	——
NoPP	1540	406	4560	903	——
Cor. (*NPS*)	1501	388	4314	871	——
$R_{Err}(NPS)$	2.53	4.43	5.39	3.54	3.97
Cor.(*Cluster*)	1531	393	4365	877	——
R_{Err} (*Cluster*)	0.58	3.20	4.28	2.88	2.74
R_{Err} *Reduced*	77.08	27.77	20.59	18.64	36.02

Table 3 gives the results of human identification after people clustering. The ground truth (H_{GT}) is manually labeled based on detected faces. The identification results (H_{ID}) are categorized into three classes: H_{Cor} (the correctly identified people), H_{Err} (containing some error candidates), H_{Mis} (missing some correct candidates). The performance is evaluated by the recall and precision of identification. We notice that the performance in V_3 is relatively low. It is because that the home video quality is poorer than other kinds of videos due to the non-professional shooting and editing. The best performance is achieved in news video, as there are much more salient people in news videos.

Table 3 Evaluation of human identification

Video	V_2	V_3	V_4	V_5
H_{GT}	16	5	19	10
H_{ID}	15	8	22	12
H_{Cor}	13	3	15	8
H_{Err}	1	3	3	2
H_{Mis}	1	2	4	2
Rec.	81.25%	60.00%	78.95%	80.00%
Pre.	86.67%	37.50%	68.18%	66.67%

To those people who have only 2 or 3 candidate samples in video, identification accuracy would be very low, because it is too difficult to correct the errors caused by *PC-NPS* through the mutual relationships between candidate samples.

5. CONCLUSION

In this paper, we have presented a people-based video indexing scheme as well as a people similarity measure and a new unsupervised clustering algorithm in distance space. Meanwhile, the extended SVMs are successfully employed to bridge the gap between low-level features and high-level semantics. The proposed solution is able to provide user another effective method to quickly access video. The experiments on general living video programs have demonstrated the effectiveness and efficiency of the proposed approach. Actually, the people similarity can be further improved by the refinements of low-level feature extraction, such as face recognition, speaker tracking. In addition, the other video analysis technologies also can be employed to enhance people-based video indexing, for example, color or motion based shot clustering can be utilized to optimize the organization of video indexing. These issues will be the focus of our future work.

6. REFERENCES

[1] P. Xu, L. Xie, S. F. Chang, A. Divakaran, A. Vetro, H. Sun, “Algorithms and Systems for Segmentation and Structure Analysis in Soccer Video”, IEEE International Conference on Multimedia and Expo, Tokyo, Japan, August. 22-25, 2001.

[2] Q. Huang, Z. Liu, A. Rosenberg, D. Gibbon, B. Shahraray, “Automated Generation of News Content Hierarchy By Integrating Audio, Video, and Text Information”, International Conference on Acoustics, Speech, and Signal Processing, March, Phoenix, 1999.

[3] M. R. Naphade and T. S. Huang, “Semantic video indexing using a probabilistic framework” In Int. Conf. on Pattern Recognition, vol. 3, pages 79-84, 2000.

[4] S. F. Chang, W. Chen, and H. Sundaram, “Semantic visual templates: Linking visual features to semantics”, Int. Conf. on Image Processing, Chicago, October, 1998.

[5] R. Fablet, P. Bouthemy, P. Perez, “Statistical Motion-Based Video Indexing and Retrieval”, In Proc. 6th Conf. on Content-Based Multimedia Information Access, RIAO’.

[6] C.H. Ngo, T.C. Pong, H. J. Zhang, “On Clustering and Retrieval of Video Shots”. ACM Multimedia 2001, Ottawa, Canada, September 30 - October 5, 2001.

[7] Chih-Chung Chang, Chih-Jen Lin, LIBSVM: a Library for Support Vector Machines (Version 2.31) (2001).

[8] J. C. Platt, “Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods”, in Advances in Large Margin Classifiers, P. B. Alexander J. Smola. Bernhard Scholkopf and Dale Schuurmans, Editor. 1999.

[9] S.Z. Li, L. Zhu, Z. Zhang, A. Blake, H.Zhang, H. Shum, “Statistical Learning of Multi-View Face Detection”, In Proceedings of The 7th European Conference on Computer Vision, May, 2002, Copenhagen, Denmark.

[10] J. Huang, “Color-Spatial Image Indexing and Applications,” PhD thesis, Cornell University, 1998.

[11] L. Zhang, F. Lin, B. Zhang, “A CBIR method based on color-spatial feature”, IEEE Region 10 Annual International Conference 1999 (TENCON'99), Chejum, Korea, pp.166-169, 1999.

[12] L. Lu, H. J. Zhang, "Speaker Change Detection and Tracking in Real-Time Broadcasting Analysis ", ACM Multimedia 2003.