14th European Signal Processing Conference (EUSIPCO 2006), Florence, Italy, September 4-8, 2006, copyright by EURASIP

HIERARCHICAL VIDEO SUMMARIES BY DENDROGRAM CLUSTER ANALYSIS

Sergio Benini, Aldo Bianchetti, Riccardo Leonardi, Pierangelo Migliorati

DEA-SCL, University of Brescia, Via Branze 38, I-25123, Brescia, Italy
phone: + (39) 030 3715450, fax: + (39) 030 380014
email: {firstname.lastname}@ing.unibs.it, web: www.ing.unibs.it/tlc

ABSTRACT

In the current video analysis scenario, effective summarization of video sequences through shot clustering facilitates the access to the content and helps in understanding the associated semantics. This paper introduces a generic scheme to produce hierarchical summaries of the video document starting from a dendrogram representation of clusters of shots. The evaluation of the cluster distortions, and the exploitation of the dependency relationships between clusters on the dendrograms, allow to obtain only a few semantically significant summaries of the whole video. Finally the user can navigate through summaries and decide which one best suites his/her needs for eventual post-processing. The effectiveness of the proposed method is demonstrated by testing it on a collection of video-data from different kinds of programmes, using and comparing different visual features on color information. Results are evaluated in terms of metrics that measure the content representational value of the summarization technique.

1. INTRODUCTION

As long as we are entering the multimedia era, tremendous amounts of video data have been made available to the normal users. Meanwhile, the needs for efficient retrieval of desired information has led to the development of algorithms that enable automated analysis of large video databases.

If the field of video analysis, the segmentation into shots and the key-frame extraction are now commonly considered as the prior steps for performing effective content-based indexing, summarization and retrieval. However, a shot separation often leads to a far too fine segmentation. So building upon this, efforts are invested towards grouping shots into more compact structures sharing common semantic threads. Providing a compact representation of a video sequence, clusters of shots results to be useful for generating static video summaries. Clustering methods based on a time-constrained approach have been presented in [11] and [8]. Visual similarity between shots has been measured between key-frames by means of color pixel correlation in [11], or by block matching in [5]. Other algorithms adopt a short term memory-based model of shot-to-shot *coherence* as in [6] and [10]. Lately, spectral methods [7] resulted to be effective in capturing perceptual organization features. Video summarization techniques using clusters of shots can be found in [4], and [12] while other recent summarization methods use graph theory [1] and curve splitting [2].

The principal aim of the paper is to propose a general strategy to obtain a hierarchical summary of a given video. The proposed tracking of cluster distortion and the use of dendrogram representation allow to stop the clustering process only on few significant levels. The goal of such analysis is to generate hierarchical summaries of the video document at increasing level of granularity, providing the user with a fast non-linear access to the desired visual material. The obtained results can be useful for further post-processing, such as semantic annotation and story unit detection [5].

The proposed procedure can employ any visual low-level feature provided with a method to estimate similarity between shots. In the performed tests a tree-structured vector-quantization on *LUV* color space and color histogram on *HSV* space have been employed and compared in representing the video shot content.

The paper is organized as follows: sections 2 and 3 present an effective shot-clustering algorithm which allows the generation of the hierarchical summaries through a dendrogram analysis; section 4 proposes two different low-level features that can be employed in the presented generic scheme; finally, in sections 5 and 6 experimental results and conclusions are discussed.

2. VIDEO SHOT CLUSTERING

In the next we assume that a method of shot detection has already been adopted, and that the video has been already decomposed into shots.

2.1 Shot-to-Shot Similarity

Let us also assume that a measure $\phi(S_i, S_j)$ that compares the similarity between shots S_i and S_j is provided. Usually shot-to-shot similarity is computed on the most representative key-frames extracted from each shot. In general we suggest that the procedure has to be functionally scalable to the case when more than one key-frame per shot is needed to represent its visual content.

We did dissertate about a shot-to-shot "similarity" and not of shot-to-shot "distance" since, although a measure can be symmetric, it is no longer a metric, for two reasons as exposed in [6]. First, since two different shots can have a dissimilarity of 0, this measure is not positive definite. Second, this measure does not obey the triangle inequality. Nevertheless, as we will show in section 4, efficient similarity measures able to capture differences between shot visual contents can be proposed.

2.2 Cluster-to-Cluster Similarity

Once it is possible to evaluate similarity between shots, the next step is to identify clusters of shots. Suppose we have a sequence with N_s shots. At the beginning of the iterative process each shot belongs to a different cluster (level-N_s). At each new iteration, the algorithm sets to merge the two most similar clusters, where similarity between clusters C_i and C_j, $\Phi(C_i, C_j)$, can be defined as the average of the similarities

between shots belonging to C_i and C_j, *i.e.*:

$$\Phi(C_i, C_j) = \frac{1}{N_i N_j} \sum_{S_i \in C_i} \sum_{S_j \in C_j} \phi(S_i, S_j) \qquad (1)$$

where N_i (N_j) is the number of shots of cluster C_i (C_j).

2.3 Dendrograms

The ouput resulting from a clustering process can be graphically rendered by a dendrogram plot. A dendrogram consists of many ⊓-shaped lines connecting objects in a binary tree. For our scope, a dendrogram represents the whole clustering process of N_s shots, from the level-N_s (each cluster containing one single shot) up to level-1, where a single cluster contains all the shots of the sequence (as in Figure 1). Moreover the height of each ⊓-branch represents the similarity between the two clusters being connected, so that low (high) connections correspond to similar (dissimilar) merged clusters. Through a dendrogram, it is therefore possible to follow the clustering process at each iteration step, every level providing a different representation of the video sequence.

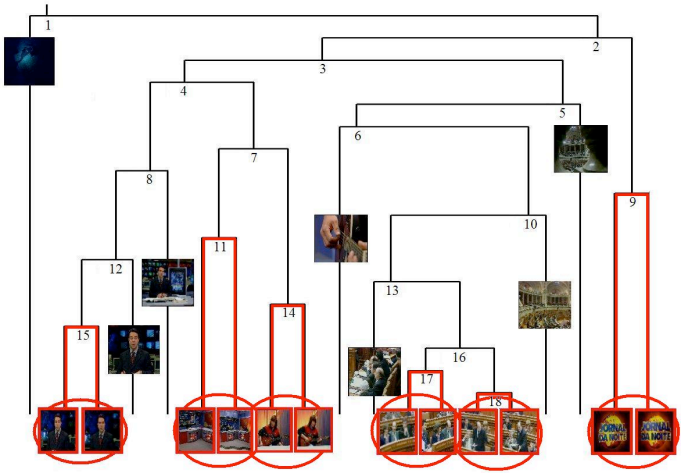

Figure 1: Dendrogram with the *leading* clusters highlighted.

3. HIERARCHICAL SUMMARIZATION

Due to the large number of merging steps, observing the clustering process at each level is almost of no use for a multimedia content consumer. Our principal aim is to automatically determine few significant levels (among all generated ones) able to offer the user semantically significant *summaries* of the observed video sequence.

A movie, for example, can be summarized at various levels of granularity, but only very few of them are *semantically* significant. For example, the top level *summary* can be the whole programme; on a lower level, it may be helpful to discriminate between the "outdoor" and the "indoor" shots; then, inside the "indoor", to distinguish among the different settings, and so on. With such a hierarchic scheme, the video content can be expressed progressively, from top to bottom in increasing levels of granularity.

3.1 *Leading* Clusters in Dendrogram

Observing the bottom level of a dendrogram, it is easy to single out the *leading* clusters as the ones originally formed by the fusion of two single shots (see Figure 1). In the upper branches of the dendrogram, each time a *leading* cluster merges with another one, it propagates its property of being a *leading* cluster to the new formed one. Since at each merging step at least one of the two merged clusters is a *leading* cluster, only by tracking the evolution of the *leading* clusters it is possible to perform a complete analysis of the dendrogram.

Let C_k^* be a *leading* cluster, and let us call $C_k^*(i)$ the cluster at level-i, where $i \in I = \{N_s, N_s - 1, \ldots, 1\}$. Tracking the evolution of C_k^* from level-N_s to level-1 it is possible to evaluate the cluster's internal distortion introduced as the cluster grows bigger. The cluster distortion is simply the reconstruction error that would be produced in using the cluster visual content to represent the actual content of its video shots. In particular, let $I_k^* = \{i_1, i_2, \ldots, i_n\} \subseteq I$ be the sub-set of levels of I in which $C_k^*(i)$ actually takes part in a merging operation, the internal distortion of cluster C_k^* at level i_j can be expressed as:

$$\Psi(C_k^*(i_j)) = \Phi(C_k^*(i_{j-1}), C_h) \qquad (2)$$

where C_h is the cluster (which can be *leading* or not) merged with C_k^* at level-i_j (*i.e.* the internal distortion is given by the cluster similarity between the two clusters being merged).

3.2 *How to Extract the Significant Summaries*

Tracking the evolution of the internal distortion of each *leading* cluster C_k^* on each level belonging to I_k^*, it is possible to automatically determine which are the few semantically significant levels to be considered for building *summaries*. Observing the internal distortion of each *leading* cluster, $\Psi(C_k^*)$, and setting a threshold on its discrete derivative

$$\Psi'(C_k^*(i_j)) = \Psi(C_k^*(i_j)) - \Psi(C_k^*(i_{j-1})) \qquad (3)$$

the user is able to stop the *leading* cluster C_k^* growth at levels $D_k^* = \{i_{d_1}, i_{d_2}, \ldots, \ldots, i_{d_n}\} \subseteq I_k^*$, that we can call *arrest levels*. These levels usually indicate meaningful moments in the growing process of C_k^*. In fact, when the height of the ⊓-branch of the dendrogram varies significantly with respect to the previous steps, this represents a substantial change in the cluster's visual content (*i.e.* the shots belonging to the two clusters being merged are visually different). In general low threshold values (allowing the merge of clusters showing strong visual similarity) determine a large number of *arrest levels*, while higher threshold values (allowing the fusion of also visually different clusters) reduces the levels of D_k^*.

Once computed all the sets D_k^* for each C_k^* for the given threshold, all the significant *summaries* for the investigated sequence can be obtained. The number of the available *summaries* is given by

$$w = \max_k |D_k^*| \qquad (4)$$

where w is the maximum cardinality among sets D_k^*. By modifying the threshold value, the user influences the number of the extracted summaries, obtaining more refined hierarchical summarizations as the threshold value decreases.

In order to build the m^{th} *summary* of the video ($m = 1, 2, \ldots, w$), the algorithm lets each *leading* cluster C_k^* grow until the m^{th} *arrest level* $i_{d_m} \in D_k^*$. Obviously, depending on the internal distortion derivative of each *leading* cluster C_k^*, the m^{th} *arrest levels* for different C_k^* can be placed at different heights in the dendrogram. At each step i_j^k, starting

from the bottom of the dendrogram up to the m^{th} *arrest level* ($i_j^k \in \{i_1^k, \ldots, i_{d_m}^k\} \subseteq I_k^*$), the cluster C_k^* merges with another cluster C_h. If C_h is a *leading* cluster, the dependency condition between the merging clusters must be verified, *i.e.* that the m^{th} *arrest level* of C_h is higher in the dendrogram than the level i_j^k. This condition (which is formally given by $i_{d_m}^h \leq i_j^k$) verifies that the cluster C_h has not been already arrested on a previous level with regard to that of the merging with C_k^*. If the dependency condition is not fulfilled, the growth of C_k^* must be stopped iteratively at lower levels on the dendrogram (*i.e.* going back to level $i_{(j-1)}^k$) until the dependency condition with the corresponding merging cluster is verified.

The resulting set of all the obtained clusters determines the m^{th} *summary* of the video. Notice that the output clusters for the m^{th} *summary* are placed at different heights in the dendrogram, depending on the single *leading cluster* internal distortions. For example, referring to the 3^{rd} summary of the *Portuguese News* sequence in Figure 2 obtained from the dendrogram of Figure 1, the first cluster is arrested at level 5 of the dendrogram, the second at level 4, the third at level 9 and the last one (being not a *leading cluster*) remains isolated until level 1.

4. LOW-LEVEL COLOR FEATURES

The proposed procedure for the creation of hierarchical summaries can employ any visual low-level feature provided with a method to estimate a shot-to-shot similarity. In the next we briefly present two different low-level color features and their related shot similarity measures.

4.1 Histogram on *HSV* color space

As a first low-level feature for visual content representation, the color histogram in the *HSV* space has been computed for all extracted key-frames. More specifically, adopting the method used by the correspondent *MPEG7* descriptor, the three color components *HSV* have been quantized into 12, 4 and 4 bins respectively. Similarity between two shots, S_i and S_j, has been measured by means of a symmetric form of the *Kullback-Leibler* distance computed on the histograms of the representative key-frames.

4.2 Tree-Structured Vector Quantization

Starting from the given shot decomposition, each shot can be analyzed in terms of its *vector quantization* codebook computed on *LUV* color space. For each extracted key-frame, a *tree-structured vector quantization (TSVQ)* codebook is designed so as to reconstruct each frame with a certain distortion with respect to the original one. In the specific, after having been sub-sampled in both directions at *QCIF* resolution, and filtered with a denoising gaussian filter, every frame is divided into non overlapping blocks of $N \times N$ pixels, scanning the image from left to right and top to bottom. All blocks are then represented using the *LUV* color space and used as the training vectors to a *TSVQ* algorithm [3] by using the *Generalized Lloyd Algorithm (GLA)* for codebooks of size $2^n (n = 0, 1, 2, \ldots)$ without exceeding a pre-determined distortion limit. The objective of this approach is to produce codebooks for each key-frame with close distortion values, so as to allow for a further comparison between different codebooks. The similarity between two shots can then be measured by using the codebooks computed on respective shots by the similarity measure proposed in [9] based on the cross-effect of the two codebooks on the two shots.

5. EXPERIMENTAL RESULTS

Applying this scheme, for example to a short *Pulp Fiction* sequence, *summaries* can be parsed into a hierarchical structure, each level containing a compact overview of the video at different granularity. Looking at Figure 2, the top (4^{th}) *summary* is a unique cluster containing all the shots; the 3^{th} *summary* distinguishes among three different settings. Then, the hierarchical decomposition continues on lower *summaries* at increasing levels of granularity, allowing the user to evaluate the quality of the decomposition with respects to his/her own desires. After that, he/she can recursively descend the hierarchy until a satisfactory result is achieved.

In order to objectively evaluate the cluster decomposition accuracy, we carried out some experiments using video segments from one news programme, three feature movies, two soap operas, one miscellaneous programme and one cartoon for a total time of about 4 hours of video.

To judge the quality of the detected results, the following criterion is applied:
*"A cluster belonging to the m^{th} summary is judged to be correctly detected if and only if **all** shots in the current cluster share a common semantic meaning. Otherwise the current cluster is judged to be falsely detected"*.

Clustering *Precision P* is used for performance evaluation, where *P* is defined as:

$$P = \frac{\text{\# rightly detected clusters}}{\text{\# detected clusters}} \tag{5}$$

For example in the 1^{st} summary of *Pulp Fiction* (see Figure 2) we have a cluster containing only shots sharing the semantics "J. Travolta in a car". In this case the cluster is considered as correctly detected. It has to be pointed out that, if we look to a specific shot in the final hierarchy, the semantics of the clusters containing the shot changes depending on the summarization level. In our example, if we climb on a higher level of abstraction to the 2^{nd} summary, the shots with J. Travolta in the car are clustered together with those showing S.L. Jackson in the same car, so that the shared semantics among all shots would be a more general "Man in a car".

Clearly, at the top level summary (all shots belonging to one cluster), the cluster detection precision would be 100%. And the same happens if we treat each shot as a cluster. Hence, in order to discriminate the representative power of a given summary, another measure is needed to express the *Compression* factor of the summary, i.e.:

$$C = 1 - \frac{\text{\# detected cluster}}{\text{\# shot in the video}} \tag{6}$$

The experimental results of cluster detection at different summarization levels for all our video data set are given in Table 1 in terms of *Precision P* and *Compression C* for both the employed features (*HSV* and *TSVQ*). As can be noted, the use of tree-structured vector quantization and its related shot-to-shot similarity measure, clearly outperforms the use of *HSV* histograms in representing each shot content and evaluating visual similarity between shot and clusters of shots.

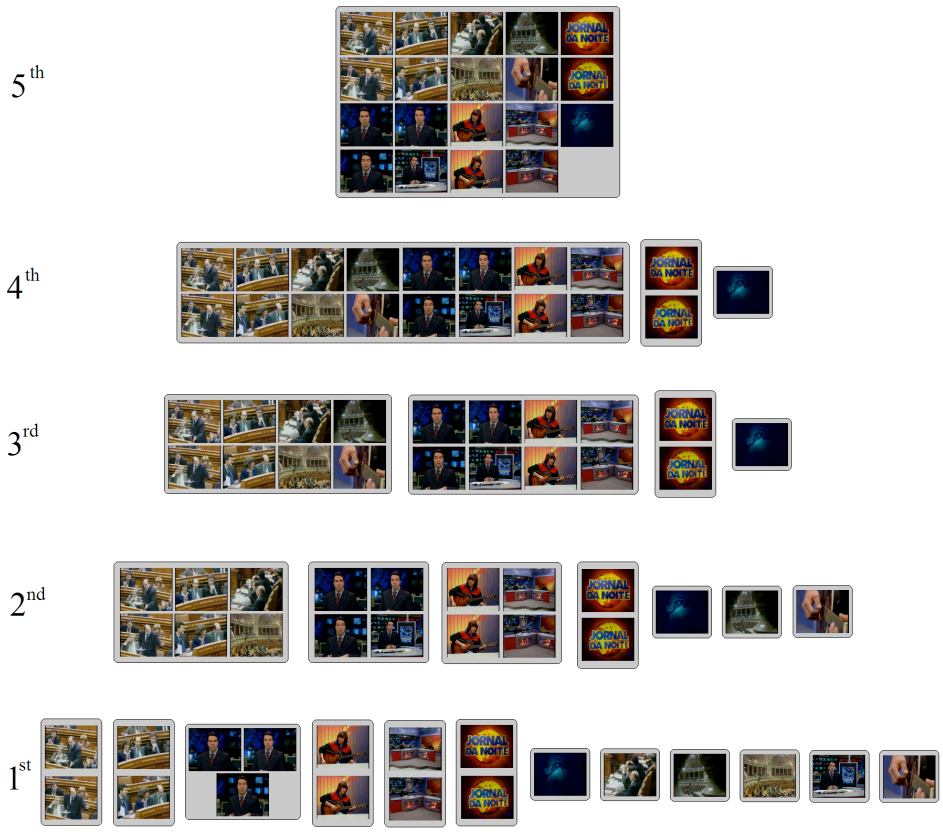

Figure 2: Hierarchical summaries for *Portuguese News* programme obtained with the *TSVQ* color feature.

Figure 3: Hierarchical summaries for *Pulp Fiction* movie obtained with the *TSVQ* color feature.

Video	Summaries *(by TSVQ)*	*C (%)*	*P (%)*	Summaries *(by HSV)*	*C (%)*	*P (%)*
Portuguese News (news)	1^{st} *(217 clusters)*	54.4	86.1	1^{st} *(194 clusters)*	59.2	82.6
476 shots	2^{nd} *(117 clusters)*	75.4	68.3	2^{nd} *(125 clusters)*	73.7	63.0
47:21	3^{rd} *(81 clusters)*	82.9	49.3	3^{rd} *(78 clusters)*	83.6	46.1
Notting Hill (movie)	1^{st} *(201 clusters)*	53.1	88.1	1^{st} *(183 clusters)*	57.3	86.9
429 shots	2^{nd} *(111 clusters)*	74.1	81.1	2^{nd} *(114 clusters)*	73.4	78.9
30:00	3^{rd} *(69 clusters)*	83.9	65.2	3^{rd} *(72 clusters)*	83.2	62.5
A Beautiful Mind (movie)	1^{st} *(96 clusters)*	52.5	97.9	1^{st} *(85 clusters)*	57.9	87.0
202 shots	2^{nd} *(56 clusters)*	72.3	92.8	2^{nd} *(52 clusters)*	74.2	78.8
17:42	3^{rd} *(26 clusters)*	87.1	65.4	3^{rd} *(34 clusters)*	83.2	64.7
Pulp Fiction (movie)	1^{st} *(91 clusters)*	48.3	94.5	1^{st} *(73 clusters)*	58.5	90.4
176 shots	2^{nd} *(54 clusters)*	69.3	87.0	2^{nd} *(50 clusters)*	71.6	84.0
20:30	3^{rd} *(35 clusters)*	80.1	71.4	3^{rd} *(36 clusters)*	79.5	69.4
Camilo & Filho (soap)	1^{st} *(68 clusters)*	51.4	95.6	1^{st} *(60 clusters)*	57.1	91.2
140 shots	2^{nd} *(35 clusters)*	75.0	80.0	2^{nd} *(37 clusters)*	73.6	72.9
38:12	3^{rd} *(23 clusters)*	83.6	73.9	3^{rd} *(24 clusters)*	82.9	62.5
Riscos (soap)	1^{st} *(192 clusters)*	54.6	88.0	1^{st} *(182 clusters)*	56.9	86.3
423 shots	2^{nd} *(107 clusters)*	74.7	75.7	2^{nd} *(109 clusters)*	74.2	77.0
27:37	3^{rd} *(65 clusters)*	84.6	69.2	3^{rd} *(69 clusters)*	83.7	68.1
Misc. (basket/soap/quiz)	1^{st} *(94 clusters)*	51.8	93.6	1^{st} *(77 clusters)*	60.5	88.3
195 shots	2^{nd} *(47 clusters)*	75.9	82.9	2^{nd} *(45 clusters)*	76.9	80.0
38:30	3^{rd} *(30 clusters)*	84.6	80.0	3^{rd} *(32 clusters)*	83.6	78.1
Don Quixote (cartoon)	1^{st} *(96 clusters)*	48.9	83.3	1^{st} *(60 clusters)*	68.1	58.3
188 shots	2^{nd} *(42 clusters)*	77.7	54.8	2^{nd} *(38 clusters)*	79.8	57.9
15:26	3^{rd} *(19 clusters)*	89.9	31.6	3^{rd} *(26 clusters)*	86.2	34.6

Table 1: For each video the first three *summaries* obtained using the *TSVQ* feature and the *HSV* color histogram are presented in terms of *Compression C* and *Precision P*.

6. CONCLUSIONS

This work describes the issue of obtaining a reduced set of hierarchical video summaries by clustering shots and by using a dendrogram representation for clusters. The proposed scheme is suitable for employing different visual low-level features and shot-to-shot similarity measures. Extensive test demonstrated that obtained summaries express video content progressively at increasing levels of granularity and provide the user with a compact representation of video content for a fast access to the desired video material.

REFERENCES

[1] H. S. Chang, S. S. Sull and S. U. Lee, "Efficient video indexing scheme for content based retrieval," IEEE Trans. on CSVT, Vol. 9, No. 8 , Dec 1999.

[2] D. DeMenthon, V. Kobla and D. Doermann, "Video Summarization by curve simplification," CVPR'98, Santa Barbara, USA, 1998.

[3] A. Gersho and R. M. Gray, "Vector Quantization and Signal Compression", Kluwer Academic Publishers, 1992.

[4] Y. Gong and X. Liu, "Video summarization and retrieval using Singular Value Decomposition," ACM MM Systems Journal, Vol. 9, No. 2, pp. 157-168, Aug 2003.

[5] A. Hanjalic and R. L. Lagendijk, "Automated high-level movie segmentation for advanced video retrieval systems," IEEE Trans. on CSVT, Vol. 9, No. 4, June 1999.

[6] J. R. Kender and B.-L. Yeo, "Video scene segmentation via continuous video coherence", CVPR'98, pp. 367-373, Santa Barbara, USA, 1998.

[7] J-M. Odobez, D. Gatica-Perez and M. Guillemot, "Video shot clustering using spectral methods", CBMI'03, Rennes, France, Sept 2003.

[8] E. Sahouria and A. Zakhor, "Content analysis of video using principal components," IEEE Trans. CSVT, Vol. 9, No. 8, pp. 1290-1298, 1999.

[9] C. Saraceno and R. Leonardi, "Indexing audio-visual databases through a joint audio and video processing", Int. Journal of Imaging Systems and Technology, Vol. 9, No. 5, pp. 320-331, Oct 1998.

[10] H. Sundaram and S. F. Chang, "Determining computable scenes in films and their structures using audio-visual memory models," ACM, pp.95-104, Los Angeles, USA, 2000.

[11] M. M. Yeung and B.-L. Yeo, "Time-constrained clustering for segmentation of video into story units," ICPR'96, Vol.III-Vol.7276, p.375, Vienna, Austria, Aug 1996.

[12] D. Q. Zhang, C. Y. Lin, S. F. Chang and J. R. Smith, "Semantic video clustering across sources using bipartite spectral clustering," ICME'04, Taiwan, June 2004.