A Novel Tool for Quick Video Summarization using Keyframe Extraction Techniques

Mathias Lux*, Klaus Schöffmann*, Oge Marques[+], Laszlo Böszörmenyi*

*Institute for Information Technology
Klagenfurt University
Universitätsstrasse 65-67
9020 Klagenfurt, Austria
{mlux, ks, laszlo}@itec.uni-klu.ac.at

[+] Department of Computer Science and Engineering
Florida Atlantic University
777 Glades Road
Boca Raton, FL 33431 - USA
omarques@fau.edu

Abstract: The increasing availability of short, unstructured video clips on the Web has generated an unprecedented need to organize, index, annotate and retrieve video contents to make them useful to potential viewers. This paper presents a novel, simple, and easy-to-use tool to benchmark different low level features for video summarization based on keyframe extraction. Moreover, it shows the usefulness of the benchmarking tool by developing hypothesis for a chosen domain through an exploratory study. It discusses the results of exploratory studies involving users and their judgment of what makes the summary generated by the tool a good one.

1 Introduction

The explosion of video information available on the Web has generated an unprecedented need to organize, index, annotate and retrieve video contents to make them useful to potential viewers.

During the past few years, video has been promoted to a first-class data object in web-based applications. From a video consumer's perspective it has become increasingly common to watch streaming video online, or download video programs for future playback on a wide variety of devices, from desktops to cell phones. It has never been easier to create video contents, encode it with a variety of standardized codecs, upload it, embed it into existing web pages or blogs and share it with the world at large.

The popularity of YouTube[1] (with an estimated number of 200,000 videos published every day) and its many competitors and copycats has exacerbated the need for effective ways to present the essence of a video clip without requiring that the user actually watch (part of) the video to know what it is about. After all, despite the fact that the average length of a video clip available on YouTube is only 2 minutes and 46.17 seconds, the time it would take to view all of the material on YouTube (as of March 17th 2008) is 412.3 years!

In summary, video is an inherently unwieldy medium whose contents need to be summarized in order to be truly useful to its potential viewers. Professional video summarization tools (e.g., Virage VideoLogger[2]) have been around for more than a decade and focus on large video footage repositories (e.g., from major TV news networks) and often benefit from the structure found in those programs. In this paper we present a novel, simple, and easy-to-use tool for the benchmarking of low level features for video summarization based on keyframe extraction. We focus on the process for the generation of a number of still images from video frames, which describe the video in an optimal way, while leaving out frames with low relevance for a summary.

The remainder of the paper describes the details of the algorithms and features used in the current version of the tool. Moreover, we discuss the results of exploratory studies involving users and their judgment of what makes a summary a good one to show the applicability of our tool for hypothesis development and benchmarking.

The tool described in this paper is publicly available to video processing researchers willing to experiment with several parameters and features used in the keyframe extraction algorithm. The open source nature of the tool makes it possible to modify and expand it to one's needs or desires. The simple and intuitive summaries provided by the tool make it possible to obtain a quick assessment of video summarization algorithms and parameters, in a way that is comparable to a basic content-based image retrieval (CBIR) system with query-by-example (QBE) functionality for testing image features and dissimilarity metrics (among other things) while fine-tuning an image retrieval solution.

---

[1] Statistics from http://ksudigg.wetpaint.com/page/YouTube+Statistics

[2] URI: http://publications.autonomy.com/pdfs/Virage/Datasheets/Virage%20VideoLogger.pdf

2 Related work

A *video summary* – or *video abstract* – is generally described as a series of still or moving images that represent the content of the video in such a way as to provide concise information about the video to the viewer [Pf96]. Many studies, surveys, and research papers on video summarization have been published during the last decade (e.g. [YL97], [KM98], [HZ99], [PC00], [ZC02], [XMX+03], [CS06], [MP08]). A recent comprehensive survey and review [TV07] alone contains more than 160 references! In spite of the significant amount of work in this field, the consensus is that "video abstraction is still largely in the research phase" [TV07].

Two basic forms of video summaries have been identified by Truong and Venkatesh [TV07]: *keyframes* and *video skims*. A *keyframe* is a *representative frame* for a video, also known as *R-frame*, *still-image abstract*, or *static storyboard*. A *video skim* is a *dynamic summary*, consisting of representative segments of the video, also known as *moving-image abstract* or *moving storyboard*. A very popular example of a video skim is a video trailer. The main advantage of video skims over keyframes is that the former can also communicate audio/speech and motion information while the latter are limited to static visual contents only. However, keyframe-based summarizations have the advantage that a user can immediately see the content of the summary, which can be a significant time-saving advantage in some situations, e.g., when browsing through a large video archive (e.g., YouTube). In such cases, a static preview showing keyframes of the video content can help the user to quickly identify a video of interest, while video skims would require the user to sequentially watch them. Although early studies [KM98] have suggested that users prefer static keyframes to dynamic summaries, the issue is far from settled, and more recent studies [TV07] have concluded that the optimal visualization of the summarized content remains an open question and that research must put more emphasis on "viewer-pleasant" summaries. This need for simpler and more effective summaries has also been corroborated by Money and Agius [MA08] who complained about the lack of personalized video summaries, such as the ones presented in [MP08], and proposed that future research should concentrate on user-based sources of information in such a way that the effort for the user is kept minimal.

Given the current situation with this broad range of methods, tools for assessment of techniques for different domains and scenarios are needed. To the best of our knowledge a tool for benchmarking the effects of different low level features in keyframe selection has not been developed or discussed.

3 Keyframe selection benchmarking tool

The selection of keyframes for a video summary relies on information about the individual frames. In most cases, low-level features such as color histograms or texture features are used to compare keyframes to one another. Our benchmarking tool was motivated by the idea that one could assess the appropriateness and quality of video summaries with different combinations of low-level features and dissimilarity measures (used for pair wise comparison of frames based on the selected low-level features). To try and assess different combinations of low-level features and dissimilarities, an algorithm that works satisfactorily in many different feature spaces (spanned by the different features) is needed.

In our benchmarking tool keyframes are selected by using a clustering algorithm. All available frames are clustered with a fixed number of clusters (*n)*, whereas the number of clusters is equal to the number of selected keyframes. Since clustering is performed based on low-level features extracted from each frame, for appropriately chosen values of *n*, all frames within a cluster tend to be visually similar (and one of them can be selected as a representative keyframe for that cluster). For the sake of video summarization, we assume that one image per cluster describes a whole set of visually similar images. The actual size of a cluster can further be used to assess how much of a videos duration is actually covered by a cluster. Finally, the selected keyframes are visualized as a video summary. Just as with the low-level features, the actual composition of the video summary is defined in a modular way and can be easily changed and adjusted to the specific requirements of a domain, user group or evaluation strategy.

The video summarization approach implemented in the tool described in this paper consists of the following steps:

1. Extraction of global features and calculation of appropriate dissimilarity metrics (Section 2.1);
2. Clustering of frames (Section 2.2);
3. Composition of the summary image (Section 2.3).

The benchmarking tool is written in Java, has been tested on Windows and Linux and is available online[3]. It features a graphical user interface as well as a command line interface for batch processing.

3.1 Low-level feature extraction and dissimilarity calculations

For the sake of keyframe selection, an uncompressed input video is interpreted as a sequence of still images. For each of the images (frames) within a video stream, we extract selected low-level features. The algorithms for low-level feature extraction included in this study were originally made available in an open source Java-based CBIR framework, LIRe [LC08]. Additional feature extraction methods can be easily integrated by implementing a simple Java interface. In the current implementation of the video summarization tool, we employed five different combinations of features and dissimilarity functions, namely those already existent in the underlying framework:

1. 64-bin RGB color histograms with L1 distance.
2. Tamura global texture features [TMY78].
3. Color and edge directivity descriptor [CB08a] with the Tanimoto coefficient.
4. Fuzzy color and texture histogram [CB08b] with the Tanimoto coefficient.
5. Auto color correlograms [Hu97] with L1 distance.

[3] URI: http://www.semanticmetadata.net

3.2 Clustering and keyframe selection

After indexing all frames we employ a clustering algorithm to assign each frame to one of *n* clusters (where *n* is a fixed number). The choice of a clustering algorithm is limited to those that rely on a distance (dissimilarity) measure without imposing (additional) requirements on the feature space. For the current implementation, the *k*-medoid clustering algorithm has been chosen, which is a very common partitioning clustering algorithm similar to *k*-means [JMF99]. The *k*-medoid approach is applicable to keyframe selection as it has been shown for instance in [HET06]. This approach has two main advantages for our application: (i) The cluster centre is always represented by a real data point and not an "artificial cluster centre" (which is the case with the *k*-means algorithm); and (ii) the clustering only depends on the dissimilarity function applied to the image feature vectors, and not the feature itself or the feature space.

The resulting *n* clusters group frames that are visually similar according to the chosen image feature. The clusters' medoids M_1, M_2, ... M_n minimize the distance to all elements of a clusters and are therefore interpreted as most descriptive elements for the respective groups. Furthermore, to allow a ranking of chosen keyframes relative to their ability to describe the content of the video, we introduce a relevance function for medoids M_k. The relevance $r(M_k)$ of the medoids M_k depends on the number of frames in cluster C_k. Consequently, the bigger a cluster, the more keyframes are in it and more of the video's duration is covered by the cluster. Therefore the medoid of the biggest cluster summarizes the largest part of the video. Also the medoid of the smallest clusters summarizes the smallest part of the video.

$$r(M_k) = \|C_k\|$$

3.3 Summary image composition

The final step of the video summarization method implemented in our tool is the visualization of the medoids, which are actual frames of the video. Our tool presents the video frames selected for the summarization as single still images. Additional data generated in the process, such as the size of the cluster and the actual frame number, are encoded in the file name. In addition to this basic output option we also added two different exemplary visualizations. A simple storyboard summary presents all found keyframes from left to right in sequence as shown in Fig. 1. The order of the sequence depends on the size of the medoid's respective clusters. The leftmost frame in the summary represents the biggest cluster.

Figure 1 - Simple visualization where keyframes are shown in cluster-size order (medoid of the biggest clusters is the first frame to the left).

A second exemplary visualization shows the medoid frame of the largest cluster in full size and the other keyframes in smaller size (see Fig. 2). This visualization reduces the overall width of the summary. Since this visualization was used in the evaluation of the tool, it is explained in more detail in Section 3.

4 Exploratory study

In this section we describe an exploratory evaluation performed on a small user group, whose goal was to gain insight on the impact of different low-level descriptors and dissimilarity metrics on the keyframe selection algorithm. We surveyed seven users on three different videos. To underline the applicability of the benchmarking tool for new domains the videos were taken from YouTube and selected from the overall most viewed animations (Table 1).

Title	Length	Views[4] (~)
Hippo bathing	30 s	360,000
The Room - Vancouver Film School (VFS)	194 s	350,000
Dinosaurs vault	49 s	493,000

Table 1 - Videos employed for exploratory study

For this study we presented summaries based on different descriptors and using different numbers of clusters. For visualization of the selected frames we chose the following image composition (Fig. 2): the medoid frame of the biggest cluster is visualized in full size (on the left of the figure), while the remaining frames (two in our example) are resampled to a quarter of their size (half in width and height) and displayed on the right-hand side of the screen.

Figure 2 – Sample visualization of a summarization of the "Hippo bathing" video with the frame of the biggest cluster in full size to the left and the other two to the right.

[4] As of December 1, 2008

Another (subtle) feature of the proposed summary image composition scheme is the ability to visualize the distribution of cluster members over the video timeline. The last row in each keyframe represents the occurrence of frames in the respective cluster – marked with green pixels – along the time axis. A sample cluster distribution can be seen in a zoomed view on the part marked with the dotted line in Fig. 3. Assuming that the whole width of the larger frame in Fig. 3 represents the timeline of the entire video, we can see that the majority of cluster members in this case concentrate in the first half of the video.

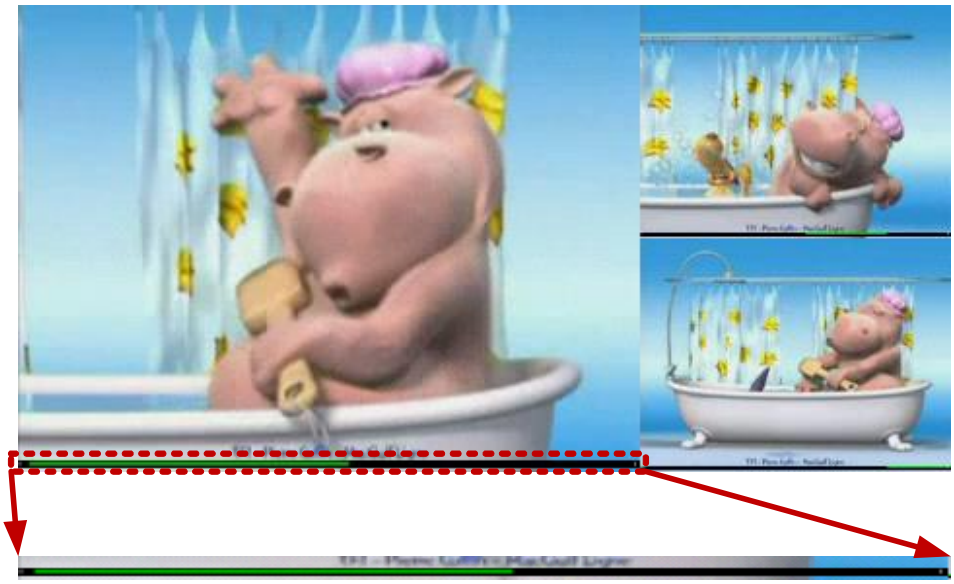

Figure 3 – Distribution of cluster members within the visualization of the selected frame. Green dots (lines) of the black bar (zoomed in from the area indicated with the dotted rectangle) show where cluster members are located in the video timeline.

Two main parameters have been varied for the study: number of clusters (n) and feature/dissimilarity metrics combinations. The case where n=1 has been omitted due to its triviality and the case where n=2 has been omitted due to disappointing results in a first exploratory investigation. Based on the selected visualization metaphor we wanted to study if users preferred 3 still images (one big and 2 small) or 5 still images (one big 4 small). Also we wanted to find out whether a visualization with 3 still images should be generated based on 3 or 4 clusters. We investigated:

- n=3 with a visualization displaying all three medoids,

- *n*=4 displaying only the three most relevant medoids and
- *n*=5 displaying all five medoids.

Note that the selected visualization metaphor features an odd number of images, so we did not test with 4 clusters showing all 4 keyframes. Furthermore for each *n* and video under consideration we created five different video summaries with different feature and dissimilarity combinations as mentioned in Section 2.2. This results in a set of 15 video summaries to assess per video.

The participants were experienced computer users, who use YouTube on a regular basis (at least once a week) and the computer on a daily basis. The survey group consisted of three female and four male participants, with ages ranging from 15 to 30 years old. For each participant the survey took place in a single session, where only the participant and the moderator (the same for each test) were present. For each video the moderator showed the actual video first. Then three groups of summaries were presented: (i) the group of summaries generated with *n*=3, (ii) the group of summaries generated with *n*=4 and (iii) the group of summaries with *n*=5. Each of the groups consisted of five different summaries generated based on the five before mentioned low-level features. The participant had to choose the best summary out of each group and had to rank the three chosen summaries according to their descriptiveness for the video. In addition to selection and ranking the moderator further asked the participant *why* the specific summary was chosen and *which criteria* were used to assess the ranking.

4.1 Results

Out of the 63 chosen images (three images per video with three videos per participant) there is no clear winner in terms of low-level features although one of the features (namely, color histogram) has been chosen the most times in absolute terms, as it can be seen in Figure 4. The visualization based on the color histogram feature has been chosen 19 times as most appropriate video summary followed by the auto color correlogram (ACC, 12 times), the fuzzy color and texture histogram (FCTH, 12 times), the color and edge directivity descriptor (CEDD, 11 times) and the Tamura global texture descriptor (9 times). Table 2 shows how often participants have picked a specific feature for different values of *n*.

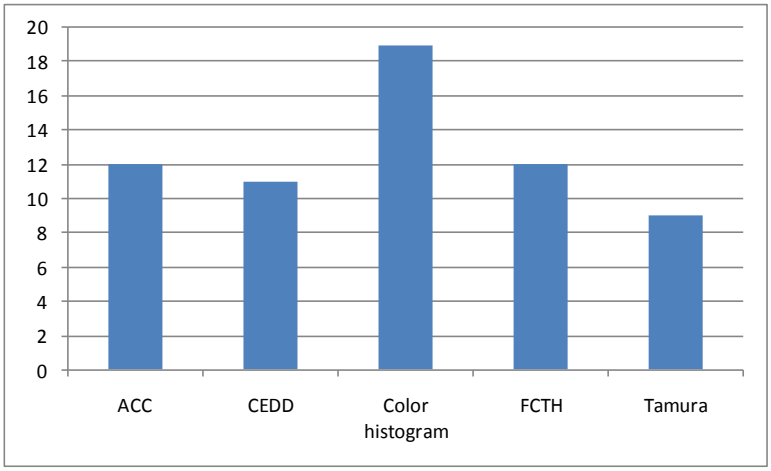

Figure 4 – Low-level features used for keyframe selection and a visualization of how often they have been selected.

	n=3	***n=4***	***n=5***
ACC	5	4	3
CEDD	8	0	3
Color Histogram	5	7	7
FCTH	0	8	4
Tamura	3	2	4

Table 2 - Selected features for different values of *n*.

From Table 2 one can see that the type of chosen features heavily depends on the chosen *n*. An example is the CEDD feature, which performs well on *n*=3 but has not been chosen at all for *n*=4. Table 3 however also indicates that the preference for low-level features changes with different videos. CEDD was mostly selected for the *Dinosaurs* video while FCTH was mainly used for the other two.

	Hippo	**Room**	**Dino**
ACC	7	2	3
CEDD	2	2	7
Color Histogram	3	8	8
FCTH	6	5	1
Tamura	3	4	2

Table 3 - Selected features for specific videos.

When asked to rank the three selected video summaries, the users ranked first the *n*=5 video summary (13 times), followed by the *n*=4 video summary (6 times) and the *n*=3 video summary (2 times). Most users voted for the 5-cluster-based summary because more of the video was captured in the more extensive summary (5 frames compared to 3 in the other two approaches).

4.2 Identified hypotheses

Based on the results of the exploratory evaluation we state three different hypotheses. Note that these hypothesis are based on the observations of the exploratory study and are intended for a detailed future study. Note also that these hypotheses are highly domain-dependent and may not hold for more general use cases. This shouldn't come as a surprise, though, since it is widely acknowledged in the multimedia research community that state-of-the-art solutions for common problems (among them, summarization and content-based retrieval) are limited to narrow domains. Moreover, by allowing users to quickly and effectively experiment with different algorithms and fine-tune their parameters, our tool makes it easier to pursue further work within a domain of choice.

The first and main hypothesis H1 is: *There is a combination of low-level feature and dissimilarity metric that performs best for the sake of keyframe selection.* Once such combination is found for a certain dataset, it may lead to subsequent improvements and optimizations. Note that the selected features and dissimilarity metrics can be quite different from the ones listed in Section 2.1 and might include specialized features and metrics that are more suitable to the chosen domain.

Moreover, due to the users' preference for the n=4 approach (where three images are shown and the smallest cluster is discarded in the visualization) over the n=3 approach, an interesting hypothesis H2 is: *Users prefer summaries where the medoid of the smallest cluster is not shown.* This hypothesis would support the idea of a "junk cluster", where unimportant or low quality frames are grouped together.

Finally, based on the qualitative feedback of the participants we can also postulate hypothesis H3: *There is an optimal number X of frames to be displayed within a video summary which is enough to cover the content of the video but still not too many to be investigated by the user in a short time.* The proposed tool allows the experimental determination of the best value of X for a certain domain in an easy way.

5 Conclusions

We have presented a tool for benchmarking different combinations of low-level features and dissimilarity metrics for video summaries based on keyframe selection. In an exploratory study we have shown the applicability of our tool and we further found that varying the number of clusters and the choice of low-level features and dissimilarity metrics used for analysis provides frame selection results that are different enough to be used as input to user satisfaction studies. The feedback received from this exploratory evaluation led us to identify three promising hypotheses to be investigated in future domain-specific evaluations. These hypotheses suggest additional research on the issues of low-level feature and dissimilarity combinations, the optimal number of images displayed within the video summary, and the relationship between the number of clusters and the number of images displayed in the video summary.

References

[CB08a] Chatzichristofis, S. A. & Boutalis, Y. S.: (CEDD: Color and Edge Directivity Descriptor. A Compact Descriptor for Image Indexing and Retrieval, in A. Gasteratos; M. Vincze & J.K. Tsotsos, ed.,'Proceedings of the 6th International Conference on Computer Vision Systems, ICVS 2008', Springer, Santorini, Greece, pp. 312-322, 2008

[CB08b] Chatzichristofis, S. A. & Boutalis, Y. S.: FCTH: Fuzzy Color And Texture Histogram A Low Level Feature For Accurate Image Retrieval, in 'Proceedings of the 9th International Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS 2008', IEEE, Klagenfurt, Austria, pp. 191-196.

[CS06] Ciocca, G. & Schettini, S.: An innovative algorithm for key frame extraction in video summarization. Journal of Real-Time Image Processing, 1(1) pp. 69-88, 2006

[HET06] Hadi, Y.; Essannouni, F. & Thami, R. O. H.: Video summarization by k-medoid clustering, in 'SAC '06: Proceedings of the 2006 ACM symposium on Applied computing', ACM, New York, NY, USA, pp. 1400-1401.

[HZ99] Hanjalic, A. & Zhang, H.: An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis, IEEE Transactions on Circuits and Systems for Video Technology, 1999 9(8), 1280--1289.

[Hu97] Huang, J.; Kumar, S. R.; Mitra, M.; Zhu, W.-J. & Zabih, R.:: Image Indexing Using Color Correlograms, in Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition, CVPR 1997, IEEE, San Juan, Puerto Rico, pp. 762-768.

[JMF99] Jain, A. K.; Murty, M. N. & Flynn, P. J. (1999), 'Data clustering: a review', ACM Comput. Surv. 31(3), 264--323.

[KM98] Komlodi, A. & Marchionini, G. (1998), Key frame preview techniques for video browsing. In DL: Proceeding of the 3rd ACM Conference on Digital Libraries. ACM Press, New York. 118-125

[LC08] Lux, M. & Chatzichristofis, S. A. (2008), Lire: lucene image retrieval: an extensible java CBIR library, in 'MM '08: Proceeding of the 16th ACM international conference on Multimedia', ACM, New York, NY, USA, pp. 1085--1088.

[MA08] Money, A.G. & Agius, H.: Video summarisation: A conceptual framework and survey of the state of the art, 2008

[MP08] Matos, N. & Pereira, F.:. Using MPEG-7 for Generic Audiovisual Content Automatic Summarization. In Image Analysis for Multimedia Interactive Services, 2008. WIAMIS'08. Ninth International Workshop on, pp. 41-45

[PC00] Parshin & Chen, L.: Implementation and analysis of several keyframe-based browsing interfaces to digital video. Lecture Notes on Computer Science, 2000, vol. 1923, pp. 206

[Pf96] Pfeiffer, S.; Lienhart, R.; Fischer, S. & Effelsberg, W.: Abstracting Digital Movies Automatically, University of Mannheim, 1996

[TMY78] Tamura, H.; Mori, S. & Yamawaki, T.: Textural Features Corresponding to Visual Perception', IEEE Transactions on Systems, Man, and Cybernetics 8(6), 1978, pp. 460-472.

[TV07] Truong, B. T. & Venkatesh, S.: Video Abstraction: A Systematic Review and Classification, in ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP) , 2007, Vol. 3, No. 1, Article 3

[XMX+03] Xu, M. & Maddage, N.C. & Xu, C. & Kankanhalli, M. & Tian, Q.:. Creating audio keywords for event detection in soccer video. In Multimedia and Expo, 2003. ICME'03. Proceedings. 2003 International Conference on, vol. 2

[YL97] Yeung, M. M. & Leo, B. L.: Video visualization for compact representation and fast browsing of pictorial content. IEEE Trans. Circ. Syst. Video Technol.1997, 7, 5

[ZC02] Zhang, D. & Chang, S. F.: Event detection in baseball video using superimposed caption recognition. In Proceedings of the tenth ACM international conference on Multimedia 2002. ACM New York, NY, USA, pp. 315-318

[Zh98] Zhuang, Y.; Rui, Y.; Huang, T. & Mehrotra, S.: Adaptive key frame extraction using unsupervised clustering, in 'Proceedings of the 1998 International Conference on Image Processing. ICIP 98.', pp. 866--870.