Estimation of Web Video Multiplicity

Sen-ching Samson Cheung and Avideh Zakhor

Department of Electrical Engineering and Computer Sciences
University of California at Berkeley, Berkeley, CA 94720, U.S.A.

ABSTRACT

With ever more popularity of video web-publishing, many popular contents are being mirrored, reformatted, modified and republished, resulting in excessive content duplication. While such redundancy provides fault tolerance for continuous availability of information, it could potentially create problems for multimedia search engines in that the search results for a given query might become repetitious, and cluttered with a large number of duplicates. As such, developing techniques for detecting similarity and duplication is important to multimedia search engines. In addition, content providers might be interested in identifying duplicates of their content for legal, contractual or other business related reasons. In this paper, we propose an efficient algorithm called video signature to detect similar video sequences for large databases such as the web. The idea is to first form a "signature" for each video sequence by selecting a small number of its frames that are most similar to a number of randomly chosen seed images. Then the similarity between any two video sequences can be reliably estimated by comparing their respective signatures. Using this method, we achieve 85% recall and precision ratios on a test database of 377 video sequences. As a proof of concept, we have applied our proposed algorithm to a collection of 1800 hours of video corresponding to around 45000 clips from the web. Our results indicate that, on average, every video in our collection from the web has around five similar copies.

Keywords: Multiplicity Detection, Internet Video, Visual Similarity, Random Sampling

1. INTRODUCTION

The amount of information on the world wide web has grown enormously since its creation in 1990. A study published in the July 1999 issue of *Nature* estimates that there were 800 million indexable pages on the web as of February 1999.[1] As there is no central management on the web, duplication of content is inevitable. As such, researchers have been interested in detecting highly similar text documents on the web for a number of years.[2,3] Overly-duplicated contents waste resources in storage and increase the effort in information mining for both human and artificial agents. This problem is in fact quite severe: as reported by Shivakumar and Garcia-Molina[3] in 1998, about 46% of all the text documents on the web have at least one "near-duplicate" – document which is identical except for low level details such as formatting, and 5% of them have between 10 and 100 replicas.

For multimedia contents, the problem of duplication is likely to be more severe than text documents. This can be attributed to the fact that multimedia content is often mirrored in multiple locations in order to facilitate downloading, or bandwidth intensive video streaming applications. Identifying all the similar contents on the web can be beneficial to many web retrieval applications. Specifically:

1. Search results can be clustered to allow easy browsing.
2. During network outages or in cases of expired links, an alternative copy in a different location can provide fault tolerance.
3. Without using costly transcoding procedures, the search engine can present the best version to users based on resource/location estimation and users' specifications. The simplest example is to choose the copy which is physically closest to the user.

Further author information: (Send correspondence to A. Zakhor)
A. Zakhor: E-mail: avz@eecs.berkeley.edu, WWW: http://www-video.eecs.berkeley.edu/~avz
S. Cheung: E-mail: cheungsc@eecs.berkeley.edu, WWW: http://www.eecs.berkeley.edu/~cheungsc

4. Clustering of information provides useful cues for statistical analysis of web data mining. For example, the inclusion of the similar content in two different users' homepages is a strong indication of the two users belonging to the same community.[4]

In this paper, we are concerned with developing efficient algorithms for detecting similar video sequences on the web. We define similar video sequences to be those with roughly the same content but possibly compressed at different qualities, reformatted to different sizes and frame-rates, or undergone minor editing in either spatial or temporal domain. In recent years, there has been a significant amount of research on similarity search in video databases. An excellent review can be found in a recent book by Perry et al.[5] Most existing work in this area focuses on developing video processing techniques to match our intuitive notion of similarity, with experimental results based on either high-quality, domain-specific video databases such as movies, television news,[6–8] or a small set of video clips from the web.[9,10] In contrast, our work focuses on detecting similarity on a large set of web video clips, which are extremely diverse in both content and quality,[11] and thus are not amenable to domain specific techniques. In addition, we pay special attention to developing low complexity algorithms that scale well to large databases such as the web.

In general, comparing two video sequences involves a computationally intensive process of measuring their edit distance.[6,7] If the primary concern is to identify approximately equivalent video sequences, it is conceivable to reduce the complexity by focusing on a few distinctive features, or using random sampling. Indyk et al.[10] use the time series of shot change durations as signature for an individual video. However, since the majority of the video sequences on the web are recordings of lectures, conferences, or single scenes from movies, televisions or demonstrations, they are generally short and contain no or very few shot changes. Furthermore, the visual quality of web video sequences is generally poor, and many of them are in streaming formats with severe frame drops during network congestions, making shot change detection an unreliable technique for similarity detection.

In this paper, we propose a new video comparison scheme called video signature. The idea behind video signature is to use a small number of frames from each video as the signature, in order to facilitate rapid comparisons. To ensure that similar video sequences produce similar signatures, a number of random images are first selected and then for each video, the frame which is the most similar to each of the random images is used to form the signature. If two video sequences are similar, it is easy to see that their signatures must also be similar, provided that the frame similarity function used is robust against low-level differences such as compression noise or minor changes in frame sizes or frame rates. While it might be possible for two completely different sequences to share a few similar signature frames, by choosing a diverse set of random seed images, it becomes increasing more unlikely for them to have overwhelmingly similar signatures. As we will see later, using the video signature algorithm, we are able to detect web video similarity with high precision and recall ratios with only a small number of frames per video.

This paper is organized as follows: the video signature model is first described in Section 2. We describe a simple statistical pruning technique for complexity reduction in Section 3. Performance and parameter selection using a ground-truth set are discussed in Section 4. To estimate the actual web video multiplicity, we have collected a large set of web video sequences. The collection process is briefly described in Section 5, along with the results of our algorithm on this database. We conclude this paper by discussing future work.

2. VIDEO SIGNATURE

We model a video V as a collection of its individual frames $\{v\}$. The similarity between video sequences is based solely on the similarity between individual frames. As such, any temporal relationship between frames is ignored. We assume that if a large percentage of the frames of two web video sequences are similar, the probability that they differ significantly in temporal dimension is small. We now proceed to define our notion of video similarity.

Let $V = \{v\}$ and $W = \{w\}$ denote two video sequences. Assume that we have a distance function $d(v, w)$ between frames v and w. We define video similarity $S(V, W)$ in terms of the degree of overlap between the two video sequences:

$$S(V,W) \triangleq \frac{1}{2}\left[\frac{\sum_{v\in V} 1_{\{d(v,w)\le\epsilon,\ w\in W\}}}{|V|} + \frac{\sum_{w\in W} 1_{\{d(v,w)\le\epsilon,\ v\in V\}}}{|W|}\right], \quad (1)$$

where ϵ is a small positive number denoting the noise tolerance between similar frames, $|V|$ denotes the size of set V, and 1_A equals to 1 if the set A is non-empty and zero otherwise. The first sum in (1) indicates the fraction of

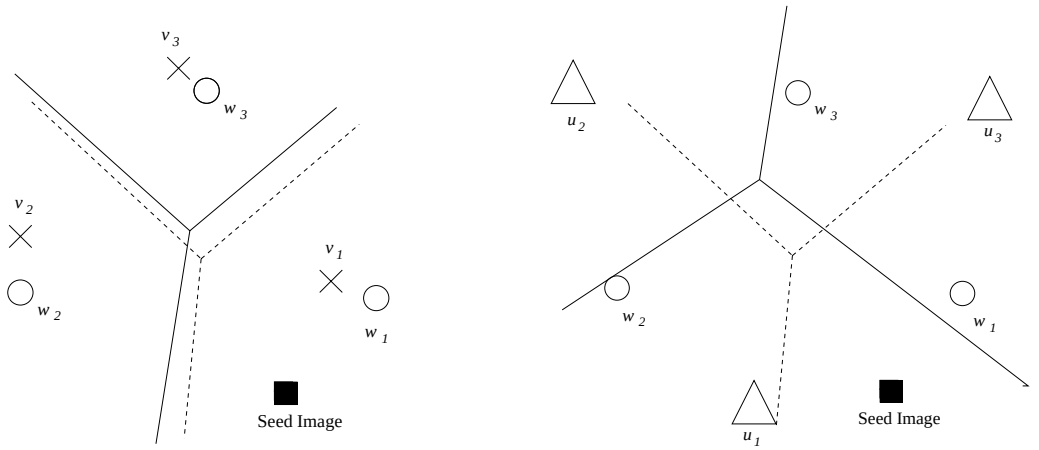


Figure 1. *Voronoi diagrams for (a) similar video sequences V and W, and (b) dissimilar video sequences U and W. The broken lines denote the voronoi regions of video W in both (a) and (b) while the solid lines denote the voronoi regions of video V in (a) and those of video U in (b). Using a single seed image as shown in the diagram, u_1, v_1 and w_1 become the signatures for U, V and W respectively.*

the frames in video V that has at least one similar corresponding frame in video W, and the second sum reflects the same measurement of W with respect to V. $S(V, W)$ is close to one when V and W share roughly the same set of similar frames, and zero when V and W have no frames in common. It is difficult to use S in practice because its complexity is proportional to the product of the sizes of V and W. To reduce the complexity, we introduce a particular form of random sampling.

Let $R = \{s_1, s_2, \ldots, s_M\}$ be a set of M random images which we call *seed images*. Define an M-tuple of frames V_R called the *signature* of V with respect to R as follows:

$$V_R \triangleq (v_{s_1}, v_{s_2}, \ldots, v_{s_M}), \quad \text{where} \quad v_{s_i} \triangleq \arg\min_{v \in V} d(v, s_i). \tag{2}$$

Intuitively the signature consists of the frames in V which are closest to the seed images in R. If two video sequences are similar, it is likely that the corresponding signature frames of the two sequences are also similar. Given two signatures V_R and W_R, we can now define the signature similarity $T(V_R, W_R)$ as follows:

$$T(V_R, W_R) \triangleq \frac{\sum_{i=1}^{M} 1_{\{d(v_{s_i}, w_{s_i}) \leq \epsilon\}}}{M}. \tag{3}$$

For similar signatures, T is close to one, while for dissimilar ones, it is close to zero. The above signature similarity function is analogous to the video similarity function defined in (1), except that it is applied to the signatures instead of raw video frames. The advantage of using the signature similarity is that computing $T(V_R, W_R)$ is much simpler than $S(V, W)$ since it only involves $O(M)$ operations. As we demonstrate experimentally in Section 4.2, using a small value of M is sufficient to produce high recall and precision ratios in identifying similar video sequences.

In Figure 1, we show two comparisons of three 3-frame video sequences U, V, and W. Video V and W are similar to each other while U is completely different. Each frame is represented as a point in the two-dimensional space. As the selection of signature is based on choosing the nearest frame to the seed, the decision region for a given sequence takes the form of a geometric construction called the Voronoi diagram.[12] Figures 1(a) and (b) show the Voronoi diagrams for similar video sequences V and W, and dissimilar video sequences U and W respectively. The broken lines denote the Voronoi regions of video W in both Figures 1(a) and (b), while the solid lines denote the Voronoi regions of video V in Figure 1(a), and those of video U in Figure 1(b). The Voronoi diagram partitions the entire space into Voronoi regions according to the nearest-neighbor rule. Thus, the three frames for a given video sequence divide the plane into three regions whereby all the images in a given region are closer to the representative

frame associated with the region than the other two frames. Specifically, the signature for a video with respect to a seed image is simply the frame for the corresponding region the seed image resides in. For similar video sequences V and W in Figure 1(a), the edges of their Voronoi diagrams are close to each other. Unless the seed image falls between the "cracks" of the two Voronoi diagrams, it is easy to see that with high probability, the signature frames chosen from the two video sequences are close to each other as well. On the other hand, for totally dissimilar video sequences such as U and W in Figure 1(b), signature frames are far apart from each other regardless of the choice of seed images. The reliability of the similarity estimate is improved when (a) more seed images are used, and (b) seed images are far apart from each other so as to avoid bias in signature frames.

We now illustrate the concept of video signature with a particular frame-based feature distance function d used in our experiments. There are a large number of image similarity functions proposed in the literature. Our goal is to choose an appropriate similarity function which is robust to compression noise, size, and aspect ratio changes as well as different sampling rates. The last requirement in particular suggests that the similarity function should be insensitive to small spatial changes due to motion. One possible candidate is the color histogram which is commonly used in detecting shot changes in video sequences. We have chosen a region-based HSV (Hue-Saturation-Value) color histogram as our frame feature vector, with sum of absolute differences (l_1-distance) as our metric.[13] Each frame is divided into four quadrants and a 178-bin color histogram is extracted for each quadrant. The color histogram has 18 bins for hue, 3 for saturation, 3 for value, plus 16 pure gray levels. We use a finer quantization in gray levels than the scheme proposed by Smith[13] to provide better classification on black-and-white video clips. Many web video sequences contain lecture notes, slides, or simple computer animations with frames sharing the same or similar monochrome background. The straightforward application of the l_1 distance between the histograms of these video frames results in a small distance value because a single color dominates the distance measurement. To remedy such a situation, when two color histograms share a single dominant color, i.e. more than 30% of the entire picture, this color is first removed, and the rest of the histogram bins are renormalized before the l_1 distance is computed. To determine if two frames are similar, the distance between the color histograms of the two frames is first computed. Then, if their distance is smaller than or equal to a pre-determined threshold ϵ, the two frames are declared to be similar. We refer to the value of ϵ as *feature distance threshold* as it depends on the particular choice of feature vector and its capability in identifying similar video frames.

An example of using the proposed color histogram is illustrated in Figure 2. A random picture selected from the web is used as our seed. We use two four-minute long MPEG7 test sequences* to illustrate the signature generation process. We first subsample the first sequence at one frame per second and call it video U. For the second sequence, we prepare two different versions named V and W to mimic common modifications of video sequences seen on the web. V is obtained by subsampling the second sequence at one frame per second, and recompressing it based on motion-JPEG with quality factor 60, while W is the subsampled version of the second sequence at ten frames per second, followed by motion-JPEG compression with quality factor 90. Despite the difference in frame rate and compression quality, the signature frames of V and W are visually much closer to each other than to the signature frame of U, as shown in Figure 2. The l_1 distance between the two color histograms corresponding to the signature frames of V and W is 1.01, while the distance between those of U and V is 4.90, and between those of U and W is 4.86.

Our approach to detecting video similarity can now be summarized in two steps: In the first step, we compute the video signature for each video sequence in our database with respect to a set of M random seed images; in the second step, we compute the signature similarity function T, as shown in Equation (3), for every pair of video signatures in the database. In the next two sections, we address a number of issues regarding the basic approach outlined above. These issues are: First, what are optimum values for parameters such number of seed images, M, and feature distance threshold, ϵ to be used in Equation (3)? Second, how do we measure the effectiveness of our approach, namely, the random seed sampling and choice of feature vectors? Third, what steps can be taken to reduce the inherent complexity of the above approach, which requires high dimensional feature computation and comparisons?

Our basic approach to addressing these issues is to construct a relatively small ground truth set, in which all video similarities are determined by subjective inspection of every video sequence in the set. Our assumption is that the video clips in the ground-truth set are representative of the ones in the larger web database, so that the parameters derived from the ground-truth set are appropriate for the larger set as well. By comparing results from the video signature algorithm with subjective inspection, and taking computation complexity factors into consideration, we

*Clips from MPEG7 data-set V11, disc 33, courtesy of Service Du Film De Recherche Scientifique.

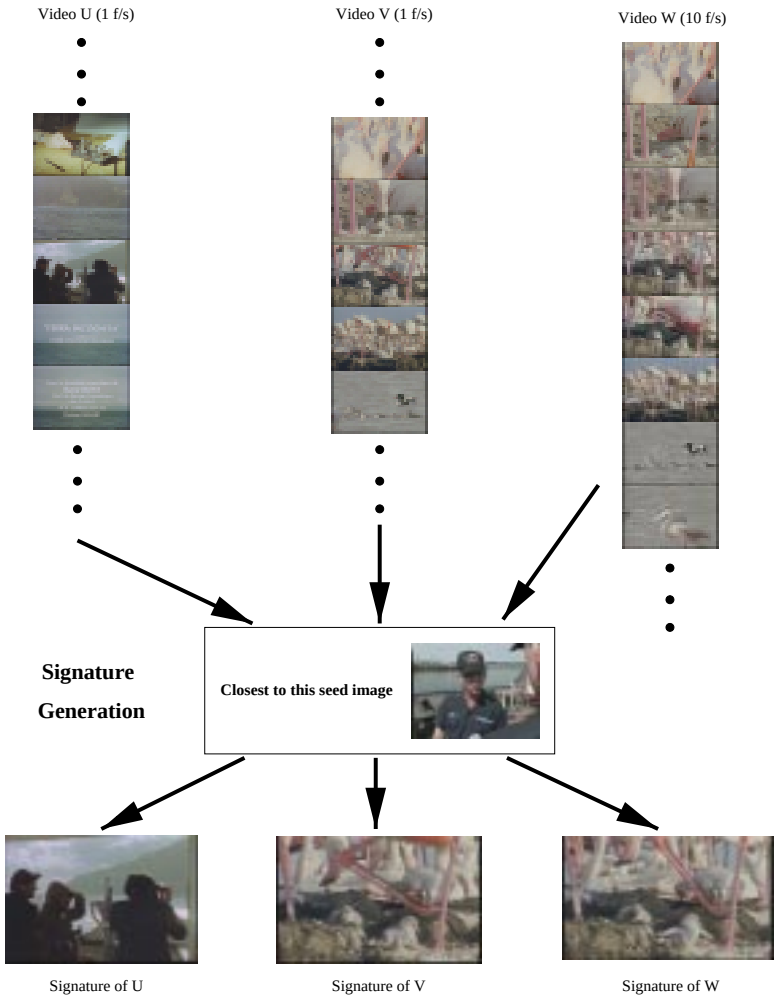


Figure 2. *Single-seed signature generation for three video sequences U, V, and W. V and W are similar to each other, while U is dissimilar to both.*

determine appropriate parameters for the video signature algorithm to be used with the large web database. In addition, the ground-truth set enables us to characterize the performance of the video signature algorithm and the effectiveness of our chosen random sampling and feature selection.

3. COMPLEXITY REDUCTION IN SIMILARITY SEARCH

High dimensional feature vectors used in constructing video signatures can make similarity detection a daunting task from a computational point of view. Classical data indexing techniques such as R-trees or SR-tress have been shown to be no better than sequential search once the dimension exceeds ten.[14] One approach to reduce the search complexity is to first apply data clustering techniques such as principal component analysis[15,16] to reduce the dimension of the data. Another class of techniques attempts to find an approximate answer in order to speed up the search process.[17–19] Many of these algorithms employ different forms of random projections of the data onto lower dimensional spaces where the relative distance between data points is roughly preserved. These techniques are of relevance to our video signature scheme since finding the frame in a video which is closest to a random seed image is a special form of random projections. In this section, we demonstrate how to take advantage of the signature generation process to reduce the complexity of finding similar signatures in a large set.

When computing the signature similarity as defined in Equation (3), a major step is to determine whether two signature frames are close to each other, i.e. $d(v_{s_i}, w_{s_i}) \leq \epsilon$ where v_{s_i} and w_{s_i} are the i^{th} signature frames of video

sequences V and W corresponding to the i^{th} seed image s_i, and ϵ is the feature distance threshold. The complexity of this computation is proportional to the dimension of the feature vector. In particular, for a database made of N clips, $N(N-1)/2$ pairwise distance computations are needed to detect similar videos. If the feature vector is D dimensional and M seeds are used, the resulting computational complexity is $MDN(N-1)/2$ or equivalently $O(MDN^2)$. By triangle inequality, $d(v_{s_i}, w_{s_i}) \geq |d(v_{s_i}, s_i) - d(s_i, w_{s_i})|$, and thus if $|d(v_{s_i}, s_i) - d(s_i, w_{s_i})| > \epsilon$, there is no need to proceed on computing $d(v_{s_i}, w_{s_i})$. Notice that both $d(v_{s_i}, s_i)$ and $d(s_i, w_{s_i})$ are distances between the signature frames and the corresponding seed and as such, are readily available from the video signature computation shown in Equation (2). Furthermore their absolute difference is a scalar operation which is significantly less complex than the distance computation between two feature vectors. We refer to this absolute difference as the *seed distance* and to $d(v_{s_i}, w_{s_i})$ as the *feature distance*.

In order to determine computational savings offered by the triangle inequality as shown above, we generate a scatter plot of seed distances versus feature distances for a ground-truth set, made of 377 video sequences which has been subjectively inspected to have 88 similar pairs. The scatter plot for a single seed is shown in Figure 3(a). The crosses indicate those pairs which are subjectively found to be similar, while the gray dots represent dissimilar pairs. As seen in Figure 3(a), the upper bound of the feature distance for all similar video pairs is around 6.7. To identify all the similar video sequences in the ground-truth set, the feature distances of all video pairs smaller than 6.7 are declared to be similar. As discussed before, the triangle inequality suggests that we might be able to reduce the number of vector feature distance computations by first pruning all signature pairs with seed distances larger than 6.7. However, as seen in Figure 3(a), this results in absolutely no savings in computation, since there are no pairs of video sequences with seed distance larger than three. Instead, we make the observation that all seed distances between similar video pairs are smaller than 0.6. This suggests that, only video pairs with seed distances below the horizontal line of height 0.6 in Figure 3(a), need to be considered for feature distance calculations. As it turns out for $M = 1$, the number of video pairs with seed distance of 0.6 or smaller is 48% of the total number of video pairs. Vector feature distance calculation only needs to be done for this portion, resulting in about a factor of two in speed up. We refer to 0.6, as the *seed distance threshold* for the case $M = 1$ for this ground-truth set, and denote it by ϵ_s. We also refer to this technique as statistical pruning.

Similar phenomenon holds true when the number of seeds is larger than one. The scatter plots of seed distance versus feature threshold for $M = 5$ and $M = 21$ are shown in Figures 3(b) and 3(c) respectively. For $M > 1$, seed and feature distances are vectors of dimension M, rather than scalars, and as such, we use the median of these vectors for the plots in Figures 3(b) and 3(c). As seen, the resulting seed distance threshold becomes smaller as the dimensionality of seed distance vector increases, resulting in larger amount of prunings, and hence larger savings in computations. This is because as more seed images are used, the number of similar signature frames for two similar video sequences increases, thus lowering both the feature distance and the seed distance. For $M = 5$ in Figure 3(b), the upper-bound on the seed distance value of similar video pairs, obviates the need for computing feature distance for 91% of all $N(N-1)/2$ video pairs, resulting in a factor of $1/0.09 \approx 11$ speed up. Similarly, for $M = 21$ in Figure 3(c), pruning rate is 97.8%, resulting in speed up factor of 45.

In interpreting these savings in computation, one must exercise caution since, in practice, the exact optimum value of ϵ_s is unknown for a set in which the ground-truth is unknown. Hence, at best, we can use estimates for ϵ_s based on the results from the existing ground-truth set, and hope that the resulting statistical pruning does not eliminate any similar video pairs for future feature distance calculations. Thus, in practice, choosing too small of a value for ϵ_s, could result in erroneous pruning of similar pairs, underestimating the amount of similarity in a data-set.

The computational savings resulting from the above statistical pruning is primarily due to the fact that for similar video pairs, the seed distance is usually much smaller than the feature distance, even though the triangle inequality merely upper bounds the seed distance by feature distance. This means that for similar video pairs, the upper bound is hardly ever achieved with equality. This is not surprising since achieving the upper bound with equality corresponds to w_{s_i}, v_{s_i} and s_i becoming co-linear, and for similar video clips, this co-linearity is highly unlikely, as long as s_i is chosen randomly and independent of w_{s_i} and v_{s_i}. Using color histogram as an example, since two similar signature frames, v_{s_i} and w_{s_i}, must share roughly the same set of non-empty bins, for s_i to be co-linear with them, it must have the same set of non-empty color bins; this is unlikely since the seed image is independent of any particular video.

Figures 3(a)–(c) only show a particular choice of seeds. To show the statistical variation for different selections of seeds, we have performed experiments over a wide range of number of seeds M, with ten random sets of seed images

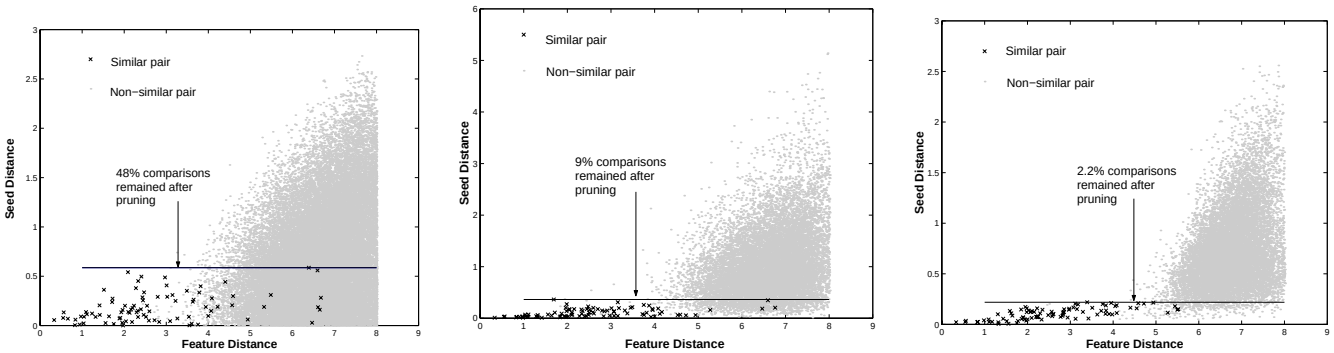


Figure 3. *Scatter plots of seed and feature distance for the ground-truth set for (a) $M = 1$, (b) $M = 5$, and (c) $M = 21$.*

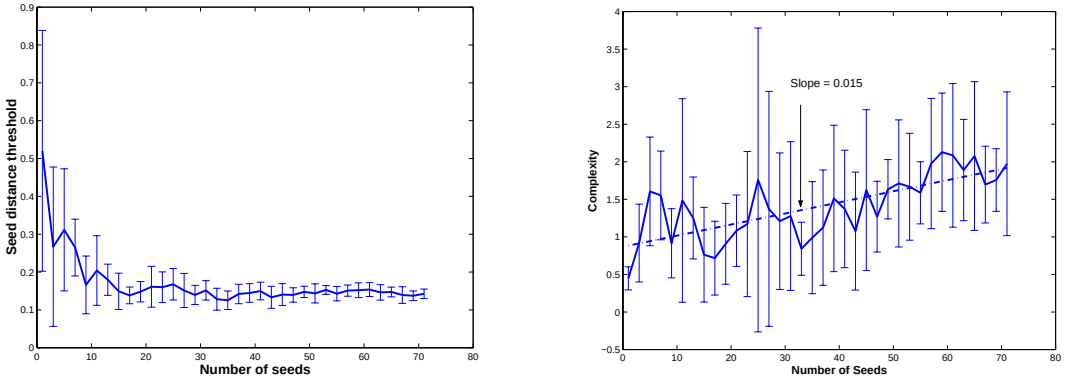
for each M. Random seed images are selected from a large database of web video outside the ground-truth set. For each M, a set of ten seed images are first selected by hand to represent a wide variety of video materials. Their corresponding color histogram feature vectors are also verified to be at least seven distance unit apart from each other[†] to avoid bias in estimation. Additional seed images are obtained by searching the video database and selecting frames that are at least seven units apart from all the selected seeds so far. Figure 4(a) shows the variation of the seed distance threshold ϵ_s as a function of the number of seed images used for the ground-truth set. As expected, the mean seed distance threshold and its variation decrease as more seeds are used but flatten out when the number of seeds exceeds 17.

As more seed images are used, more pruning is possible since the seed distance threshold also decreases. On the other hand, the remaining feature comparisons are more complex to compute since there are more signature frames, and thus the overall computational complexity may not necessarily decrease. Let N denote the number of video sequences in the database, M be the number of seeds used, D be the dimension of the feature vector, and $\rho(M)$ be the fraction of sequences in the database that survive the initial pruning. Notice that $\rho(M)$ is a function of the number of seeds M, since $\rho(M)$ decreases as more seeds are used. The total number of operations needed to find all similar video sequences in a database for a given query can be computed as follows: First, NM operations are needed to compute the seed distance between the given video and all the N video clips in the video database. After statistical pruning, $N\rho(M)$ videos have remained for which feature distance calculations must be computed; this results in $N\rho(M)DM$ operations. Thus, the total number of operations is:

$$NM(1 + D\rho(M)) = NDM(1/D + \rho(M)) \approx NDM\rho(M). \tag{4}$$

The approximation is due to the fact that seed distance is typically much simpler to compute than feature distance. In our experiments, computing feature distance takes 712 operations, while seed distance takes only one. In contrast to Equation (4), if feature distance thresholds for all video clips are computed, without the use of statistical pruning, the number of operations is NDM. Since the number of seeds M only affects $\rho(M)$ in Equation (4), we can plot the complexity factor $M\rho(M)$ and its variation as a function of M, as shown in Figure 4(b). The complexity factor is quite noisy showing a strong dependency on the particular choice of seeds. Overall, the complexity factor increases slowly with the number of seed images. The rate of increase, computed using linear regression on the average complexity factor with respect to the number of seeds, is 0.015. This is about $1/0.015 = 68$ times lower than the case when no statistical pruning is used since the complexity factor in this case grows linearly with M with slope of one, rather than $\rho(M)$. This demonstrates the effectiveness of the statistical pruning in terms of complexity reduction.

[†]Using l_1-distance, the maximum distance between two quadrant-based histograms is eight units.



(a) (b)

Figure 4. *(a) The variation of seed distance threshold for pruning versus the number of seeds; (b) The variation of the complexity factor versus the number of seeds used. Each error bar shows plus/minus one standard deviation.*

4. GROUND-TRUTH SET

4.1. Video preprocessing and Generation of the ground-truth set

To evaluate the performance of video signature algorithm, and to determine the optimum values for various parameters, a ground-truth data-set is established. This is done by first manually examining a random sample of 377 video clips obtained from the web, and then identifying 88 pairs of similar video sequences subjectively. These similar video sequences include different segments from President Clinton's television testimony, trailers of the movie *Titanic* and other arbitrary video sequences. Even though most of these pairs of similar video sequences are identical in content, they differ greatly in many low-level details : they are of different durations, sizes and compression qualities; many of them have black frames in the beginning and at the end while others are slightly modified versions of each other[‡]. Such variations among similar video sequences are typical among many of the web video sequences we have observed.

A number of preprocessing steps are performed on the video sequences before their signatures are computed. First, all black frames are removed because they are commonly used in many video sequences but provide no useful information regarding similarity. We then reduce the complexity of signature generation process by using only key-frames from the video. Using a simple quadrant-based 16-bin luminance histogram, shot boundaries of each video are first detected, and only two key-frames delimiting each shot are retained. The key-frame generation uses a conservative threshold to ensure that eliminated frames are similar to the remaining key-frames when measured with the more complex color histograms. Notice that most of the identified shot boundaries do not coincide with subjective assessments since this step is applied only to reduce the number of frames without affecting the signatures generated. On average 75% of the frames are removed using the key-frame generation process.

4.2. Parameter selection and performance evaluation

To calibrate the retrieval performance of video signature on the ground-truth set, we use concepts of recall and precision ratios, commonly used in evaluating the performance of information retrieval systems. Let α denote the number of pairs of video sequences which are found to be similar using the video signature on the ground-truth set. If among these α pairs, $\beta \leq \alpha$ pairs are in fact truly similar, then the precision ratio of the video signature algorithm is defined as β/α. The precision ratio can generally be made close to one by making the similarity test more strict and thus reducing α. To ensure that a reasonable portion of truly similar video sequences are detected by the algorithm, we measure the recall ratio which is defined as β/γ where γ is the total number of truly similar pairs in the ground-truth set. All design parameters of an algorithm should be set to optimize both recall and precision ratios simultaneously.

[‡]One such example includes two similar introduction clips to RealVideo player and RealVideo player plus.

As shown in Equation (3), three parameters are required to detect similar video sequences : the feature distance threshold ϵ for noise tolerance between similar frames, the number of seed images M, and the minimum similarity level τ above which two video signatures, V_R and W_R, are declared to be similar, i.e. $T(V_R, W_R) \geq \tau$.

The number of seeds M is chosen in such a way as to balance the retrieval performance and the computational complexity for the ground-truth set. To determine a reasonable number of seeds to be used, we first fix the recall ratio at 80% and compute the precision ratios for a range of numbers of seeds. Ten independent sets of seeds are used for each operating point. The mean precision ratios and the standard deviations are computed. As shown in Figure 5(a), the precision ratio improves sharply when more seeds are used but levels off at around 85% when the number of seeds exceeds 15. After examining those video pairs which are misclassified, we conclude that most of the problems are due to the limitations of using color histogram as the feature. For example, a number of the similar video sequences in the ground-truth set are under different illuminations which is known to degrade color histogram retrieval performance.[20]

Based on the results in Figures 4 and 5(a), we choose the number of seeds, M, to be 21 for our future experiments on large data-sets from the web; this value of M provides a reasonable compromise between precision/recall performance and computational complexity on the ground-truth set and as such, we are assuming that this compromise is also reasonable for the larger data set.

To choose the optimum value of the feature distance threshold, ϵ, to be used in detecting video multiplicity on the large data-set from the web, we fix the recall ratio at 80%, and run ten independent experiments, each using a different set of 21 random seed images on the ground-truth set. In each experiment, we compute the feature distance vector for all similar pairs in the ground-truth set, determine the median for each vector, and then obtain the average over the medians. This average is then further averaged over ten independent experiments corresponding to ten sets of 21 random seed images, in order to result in feature distance threshold, ϵ, of 3.9354.

The choice of τ is closely related to that of ϵ, since τ can be set arbitrarily close to one if ϵ is large enough. The median operation on the feature distance vector for specifying ϵ suggests the choice of 50% for τ. This is because, from Equation (3), if ϵ is chosen to be the median of the feature distance vector for two similar video signatures, V_R and W_R, then $T(V_R, W_R)$ is guaranteed to be at least 50%.

To choose the seed distance threshold, ϵ_s, for our experiments on the large data-set, we again run ten independent experiments each using a different set of 21 random seed images. For each experiment, we compute the seed distance vector for all similar video paris in the ground-truth set, compute the median of each vector and determine the threshold above which no similar video pairs exist. We then compute the average and standard deviation of the thresholds over these ten experiments and choose ϵ_s to be mean plus one standard deviation; the extra standard deviation is used to prevent excessive pruning.

We conclude this section, by showing the performance of our proposed approach over the ground-truth set. We do this, by choosing a particular set of 21 seed images, and computing the precision/recall plot for the ground truth set, as shown in Figure 5(b). In this plot, ϵ changes as we move from one extreme to another in recall ratio. Precision and recall values for $\epsilon = 3.9354$ are 84% and 85% respectively.

5. EXPERIMENTAL RESULTS

In order to obtain a reliable estimate of web video multiplicity, it is important to base our results on a representative collection of video sequences on the web. For our experiments, we use a database of about 45,000 video clips from the web with a total duration of around 1800 hours. We briefly describe this database in Section 5.1. The details of the multiplicity measurements and analysis over this database are discussed in Section 5.2.

5.1. Web video collection

A common approach to collect data from the web is to use a web crawler. A web crawler is a program that automatically traverses the web's hyperlink structure and retrieves desired information. As video sequences are sparsely distributed over the web, a web crawler requires a substantial amount of time and resources to collect a representative data-set. Our approach to building a video collection is to send a large set of queries to commercial multimedia search engines to obtain Uniform Resource Locator (URL) addresses of web video sequences. Similar methods have been used in estimating the size of the web.[21] In order not to be biased towards particular types of content, our query word set consists of about 300,000 words obtained from a general search engine,[22] an internet

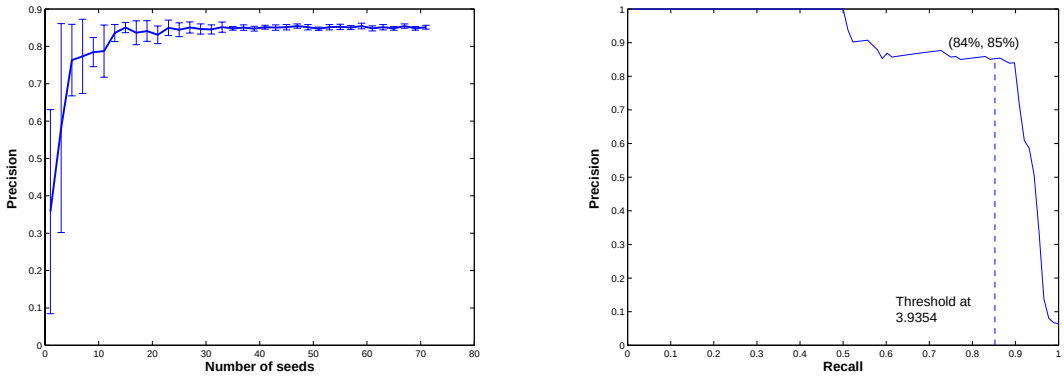


Figure 5. *(a) Precision ratios versus number of seeds for fixed recall ratio at 80%. Each error bar shows plus/minus one standard deviation. (b) Precision ratios versus recall ratios for the set of 21 seeds used in the experiment.*

video-on-demand site[23] and a speech recognizer's vocabulary.[24] All the query requests are carefully controlled so as not to overburden the search engine. Over the entire month of June 1999, about 62,000 URL addresses pointing to video content were obtained. This constitutes roughly 15% of all the non-broadcast video clips on the web, according to the figure estimated by Cambridge Research Laboratory in November 1998.[11]

The second step is to follow the resulting URLs and download the actual video sequences. Among all the video URLs, the most popular formats are RealVideo, Quicktime, MPEG-1, and AVI. Except for MPEG-1 which is an open standard,[25] the remaining formats are proprietary. This has a significant impact on the download time since no fast bitstream level processing can be applied, and the video sequences can only be captured after full decoding. In other words, the capture time is limited by the decoding speed or even real-time display in certain formats. RealVideo streaming format[26] presents an additional level of challenge since its display quality depends on the network conditions during the download. Depending on the settings of the content server, heavy packet losses on the network may cause delay, frame drops or corrupted frames. We have developed a capturing software that takes the delay due to packet losses into account but fails to detect frame drops or corrupted frames. As a result, the captured quality may vary significantly even for the same video downloaded at two different instances. In order to reduce storage requirements, all the video sequences are resampled at three frames per second. For each sequence, almost identical neighboring frames with peak signal to noise ratio (PSNR) larger than 50 dB are removed, and the remaining frames are compressed using motion-JPEG.

After eliminating synonymous[§] and expired URL entries, we capture 44,674 video clips with total duration of around 1800 hours. The total disk space required for the motion-JPEG video sequences exceeds 100 Gigabytes. The total capture time is around 30 days using four Intel Pentinum-based personal computers. In other words, on average, it takes 1.6 hours to capture 1 hour of video. The bottleneck in capturing is primarily due to the buffering delay in recording streaming RealVideo. The statistics of the four most abundant types of collected video sequences are shown in Table 1. Except for video sequences in RealVideo type, most of the other sequences are short and less than one minute long.

5.2. Video Multiplicity Measurements

In this section, we estimate the multiplicity of web video which can be defined as the average number of similar copies for any arbitrary video on the web. As described in Section 4.2, an arbitrary set of 21 seed images are used to generate signatures for our entire collection of web video sequences. All signatures are then compared to each other using the statistical pruning technique described in Section 3 with seed distance threshold set to 0.2151. Based on the results on the ground-truth set, those pairs of video signatures with similarity values, T, larger than 50%

[§]Synonymous URLs are detected using the following heuristics[27] : (i) removing the port 80 designation (the default), (ii) removing the first segment of the domain name for URLs with a directory depth greater than one (to account for machine aliases), and (iii) unescaping any "escaped" characters.

Video Type	% over all clips	Duration (mean ± std-dev in minutes)
MPEG	31	0.26 ± 0.7
QuickTime	30	0.51 ± 0.6
RealVideo	22	9.57 ± 18.5
AVI	16	0.16 ± 0.3

Table 1. *Statistics of collected web video sequences*

using feature distance threshold of 3.9354 are recorded as similar video sequences. Preliminary examination of part of the data-set indicates that our parameters work well for color video sequences but are too lenient for most of the black-and-white video clips. This is understandable because less than 10% of the total number of bins in our feature vector are allocated to gray values. To remedy this situation, we reduce the feature distance threshold ϵ to one when comparing two black-and-white video sequences. We classify a video as black-and-white when more than 70% of the pixels are gray.

All the video preprocessing and signature generation routines are written in C and take less than a day for the entire database on three Pentinum-based personal computers. The routines for cross comparisons among all the signatures are written in Perl and are run on three Pentinum-based personal computers and four Sun Microsystems UltraSparc workstations. The cross comparisons take around five days to complete. The primary bottleneck is the transfer speed between the data server and the computers.

We measure the degree of similarity of each video U, defined as the total number of video sequences similar to U. Figure 6 shows the pie chart of the degrees of similarity of all the video sequences in the data-set. The average degree of similarity can be interpreted as the average number of similar video sequences for any arbitrary video found on the web. This is different from finding the average number of copies of distinct video sequences: it is ambiguous to identify "distinct" video because similarity is not transitive, i.e. if U and V are similar and V and W are similar, U and W are not necessarily similar. If U and W indeed fail to pass the similarity test, no conclusion can be drawn on the number of distinct video sequences. Around 90% of all the video sequences have degree less than seven. The average degree of similarity is 5.3. Of particular interest are those video sequences with exceedingly large degrees. By examining all the video sequences with larger degrees than 200 (about 0.45% or 203 video sequences) and sampling their corresponding similar video sequences, 171 of them are found to be surveillance video sequences on identical physical locations, all located on the same website[¶]. The rest of them, however, are misclassifications. A number of them are astronomical pictures while others are graphs of scientific experiments or simple computer graphics illustrations. They all share a similar color appearance – a black background with sporadic color lines and dots. Since these video sequences represent significant departures from the natural video sequences we use in our training set, it is not surprising that our color histogram feature fails in these cases. More spatially oriented features should be used in such types of video sequences.

6. CONCLUSIONS AND FUTURE WORK

In this paper, we propose a computationally efficient algorithm called the video signature to detect similar video sequences using a fixed-size signature for each video. Starting with a set of randomly chosen seed images, the frames closest to a set of seed images are used as the signature. Video similarity is measured by computing the similarity between individual signature frames. Using a small ground-truth set, we achieve about 85% recall and precision ratios with 21 seeds using color histograms as the feature vector. To reduce the complexity in cross comparing all the video signatures in a database, a statistical pruning step is used which is based on computing the absolute differences between the distances of the signature frames to the corresponding seeds. Using the video signature algorithm, we estimate that on average, any arbitrary video in our collection of 45,000 web video clips has roughly five similar copies. We are currently experimenting with other features in order to improve retrieval performance and simultaneously reduce the feature dimension. In addition, we are planning to focus on theoretical characterization of the uncertainties involved in the signature generation and the statistical pruning.

[¶]Sky of Kochi'96 by Kochi National College of Technology, http://star.ee.kochi-ct.ac.jp/sky/mpeg-cal-e.html

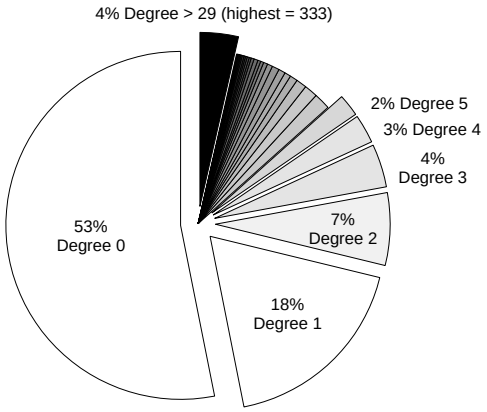


Figure 6. *Pie chart of the degrees of similarity of 44,674 video sequences. Slices with more than 2% are detached.*

ACKNOWLEDGMENTS

The authors would like to thank Dr. Rainer Lienhart of Intel for making his MoCA project source code publicly available as well as Dr. Gary Greenbaum of RealNetworks in answering our questions regarding Realplayer.

REFERENCES

1. S. Lawrence and L. Giles, "Accessibility and distribution of information on the web," *Nature* **400**, pp. 107–9, July 1999.
2. A. Broder, S. Glassman, M. Manasse, and G. Zweig, "Syntactic clustering of the web," in *Sixth International World Wide Web Conference*, vol. 29, no.8-13 of *Computer Networks and ISDN Systems*, pp. 1157–66, September 1997.
3. N. Shivakumar and H. Garcia-Molina, "Finding near-replicas of documents on the web," in *World Wide Web and Databases. International Workshop WebDB'98*, pp. 204–12, (Valencia, Spain), March 1998.
4. R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins, "Trawling the web for emerging cyber-communities," in *Proceedings of the Eight International World Wide Web Conference*, pp. 1481–93, May 1999.
5. B. Perry *et al.*, *Content-based access to multimedia information – from technology trends to state of the art*, ch. 4.3. Kluwer Academic Publishers, Massachusetts, U.S.A., 1999.
6. D. Adjeroh, I. King, and M. Lee, "A distance measure for video sequence similarity matching," in *Proceedings International Workshop on Multi-Media Database Management Systems*, pp. 72–9, (Dayton, OH, USA), August 1998.
7. R. Lienhart, W. Effelsberg, and R. Jain, "Visualgrep: A systematic method to compare and retrieve video sequences," in *Proceedings of storage and retrieval for image and video databases VI*, vol. 3312, pp. 271–82, SPIE, January 1998.
8. M. Smith and T. Kanade, "Video skimming and characterization through the combination of image and language understanding," in *Proceedings 1998 IEEE International Workshop on Content-Based Access of Image and Video Database*, pp. 61–70, (Bombay, India), January 1998.
9. S.-F. Chang, W. Chen, and H. Sundaram, "Videoq: a fully automated video retrieval system using motion sketches," in *Proceedings Fourth IEEE Workshop on Applications of Computer Vision*, pp. 270–1, (Princeton, New Jersey), October 1998.
10. P. Indyk, G. Iyengar, and N. Shivakumar, "Finding pirated video sequences on the internet," tech. rep., Stanford Infolab, February 1999.
11. M. Swain, "Searching for multimedia on the world wide web," Tech. Rep. CRL99/1, Cambridge Research Laboratory Technical Report, March 1999.
12. F. Aurenhammer, "Voronoi diagrams – a survey of a fundamental geometric data structure," *Computing Surveys* **23**, pp. 345–405, September 1991.
13. J. Smith, *Integrated Spatial and Feature Image Systems: Retrieval, Analysis and Compression*. PhD thesis, Columbia University, 1997.

14. R. Weber, H.-J. Schek, and S. Blott, "A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces," in *Proceedings of the 24th International Conference on Very-Large Databases (VLDB'98)*, pp. 194–205, (New York, NY, USA), August 1998.
15. S. Deerwester, S. Dumas, G. Furnas, T. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science* **41**, pp. 391–407, September 1990.
16. E. Sahouria and A. Zakhor, "Content analysis of video using principal components," in *Proceedings 1998 International Conference on Image Processing, volume 3*, pp. 541–5, (Chicago, IL, USA), October 1998.
17. A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via hashing," in *Proceedings of the 25th International Conference on Very-Large Databases (VLDB'99)*, (Edinburgh, Scotland), 1999.
18. J. M. Kleinberg, "Two algorithms for nearest-neighbor search in high dimensions," in *Proceedings of the Twelveth Annual ACM Symposium on Theory of Computing*, pp. 599–608, May 1997.
19. E. Kushilevitz, R. Ostrovsky, and Y. Rabani, "Efficient search for approximate nearest neighbor in high dimensional spaces," in *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, pp. 614–23, May 1998.
20. T. Gevers and A. Smeulders, "Image retrieval by multi-scale illumination invariant indexing," in *Multimedia Information Analysis and Retrieval. IAPR International Workshop, MINAR'98*, pp. 96–108, (Hong Kong, China), August 1998.
21. K. Bharat and A. Broder, "A technique for measuring the relative size and overlap of public web search engines," in *7th International World Wide Web Conference*, vol. 30, no. 1-7 of *Computer Networks and ISDN Systems*, pp. 379–88, April 1998.
22. Yahoo! Inc., http://www.yahoo.com, *Yahoo! Categories*.
23. VideoSeeker, http://www.videoseeker.com, *VideoSeeker*.
24. International Computer Science Institute, http://www.icsi.berkeley.edu/ dpwe/isrintro, *ICSI Speech recognition software*.
25. ISO/IEC, *ISO/IEC 11172-2:1993 : Information technology – Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s – Part 2:Video*, November 1992.
26. RealNetworks, http://www.real.com/devzone/library/whitepapers/overview.html, *RealVideo Technical White Paper*, Feburary 1997.
27. S. Lawrence and C. L. Giles, "Searching the world wide web," *Science* **280**, pp. 98–100, April 1998.