REIHE INFORMATIK
TR-04-008

Automatic Generation of Video Summaries for Historical Films

Stephan Kopf, Thomas Haenselmann, Wolfgang Effelsberg
University of Mannheim
– Fakultät für Mathematik und Informatik –
Praktische Informatik IV
L 15,16
D-68131 Mannheim, Germany

Automatic Generation of Video Summaries for Historical Films

Stephan Kopf*, Thomas Haenselmann, Wolfgang Effelsberg
Praktische Informatik IV
University of Mannheim
Germany
{kopf,haenselmann,effelsberg}@informatik.uni-mannheim.de

ABSTRACT
A video summary is a sequence of video clips extracted from a longer video. Much shorter than the original, the summary preserves its essential messages. In the project *ECHO* (European Chronicles On-line) a system was developed to store and manage large collections of *historical* films for the preservation of cultural heritage. At the University of Mannheim we have developed the video summarization component of the ECHO system. In this paper we discuss the particular challenges the historical film material poses, and how we have designed new video processing algorithms and modified existing ones to cope with noisy black-and-white films. We also report empirical results from the use of our summarization tool at the four major European national video archives.

Keywords
Video summarization, digital video libraries, skimming, video content analysis.

1. INTRODUCTION
The number and volume of digital video libraries is growing rapidly. TV broadcasters and other private and public film archives are digitizing their film collections. Local users of the archives have the possibility to access the material, but it is also often desirable to make the content available to the public via the Web.

A major problem of large film archives is the fact that it is difficult to visually search the content. Therefore additional meta-data information is stored for each film. Relevant films are found by searching the meta-data information. Typically the result of a query is a list of key frames with some textual information. It would be desirable to also have short *video summaries* that contain the essence of a longer film. A video summary is a short video clip that has been extracted from a longer video. Much shorter than the original, the summary preserves its essential messages. A summary does not change the presentation medium; image and audio information is available to the user.

In the project *ECHO* (European Chronicles On-line) a large software system was developed that stores and manages collections of historical films. A major goal of ECHO is to make the historical documentaries available to many users. The ECHO test archive contains more than 200 hours of films from four major national film archives[1]. The collections are precious from a cultural point of view since they document different aspects of life in European countries from the beginning of the last century until today.

ECHO user queries are text-based. The result of a query consists of textual information and key frames. In addition a user can view a video summary of the (much longer) film. In the remainder of this paper we concentrate on the generation of these summaries.

Historical film archives are a great challenge for video analysis tools. Many well-known algorithms fail due to the properties of the old material (e.g., black-and-white films, a much higher noise level in the frames and in the audio). We have developed new algorithms that analyze such material reliably. Others than the traditional features are required to find relevant shots in historical documentaries. A new heuristic approach is presented that selects the most important shots for the summary. Our video summarization tool reads and writes MPEG-I/II videos. User interaction and manual specification of parameters is possible but not required.

The remainder of this paper is organized as follows: Section 2 describes related work in the area of video presentation and summarization. Section 3 gives an overview of our video summarization application. Sections 3.1 and 3.2 describe the automatic computation of features and the selection of relevant shots for the summary. We then present empirical results in Section 4. Section 5 concludes the paper.

2. RELATED WORK
Many tools have been developed to generate a compact representation of a long video. The process is usually called *video summarization, video skimming* or *video abstracting.* Most approaches analyze visual features alone, extract key frames or calculate background mosaic images on per-shot basis. Many applications allow quick navigation based on the key frames; after clicking on a key frame they play the corresponding shots of the video.

The MoCA abstracting was one of the first tools to generate moving summaries from feature films automatically [12]. The system was initially developed to generate trailers of feature films. A major component was the detection of events of particular relevance such as explosions, gun-fire or dialogs.

The Informedia Digital Video Library project [19] at the Carnegie Mellon University has developed two applications to visualize video content. The first one provides an interface to generate and dis-

*The work of Stephan Kopf was financially supported by the European project ECHO *(European Chronicles Online)*.

[1]Instituto Luce (Italy), Memoriav (Switzerland), Netherlands Audiovisual Archive (the Netherlands) and Institut Nationale de l'Audiovisuel (France)

play so-called video skims [5]. Important words are identified in the textual transcript of the audio. Text and face recognition algorithms detect relevant frames. Video skims are generated based on the results of the automatic analysis. Additionally, a collage as an interface for browsing video collections has been introduced where information from multiple video sources is summarized and presented [4, 13].

A simple approach to reduce the length of a video is to increase the frame rate and thus speed up the playback process (time compression) [14]. IBM's CueVideo system uses this approach and modifies the time scale of the audio signal [1].

Lienhart describes an approach for video summarization especially tailored to home videos [11]. Text segmentation and recognition algorithms are used to identify the date and time inserted into the frames by the camcorder. Hierarchical clusters are build with shots based on the recording time. A heuristic selects shots based on these clusters without actually analyzing the content of the home video.

Li et al. have analyzed the user behavior for multimedia presentations in order to understand which browsing capabilities (e.g., time compression, pause removal, navigation using shot boundaries) were considered useful and allowed a quick understanding of the video [9].

In an earlier paper we have proposed to avoid the detection of shot boundaries altogether: histograms of frames are analyzed, and clusters with similar frames are build based on the k-means algorithms [6]. The selection of key frames is based on these clusters.

Many other methods have been proposed, e.g., a comic-book presentation style to arrange the key frames [3, 18] or summaries based on background mosaic images [2]. A method to summarize and present videos by analyzing the underlying story structure has been proposed very early by Yeung et al. in [20].

None of the existing research projects have addressed the specific challenges of historic film material. Our experience shows that new algorithms must be developed and existing algorithms must be modified to cope with old films:

- most material is black-and-white, making color-based features useless,
- there is a lot of noise in the images, thus a comparison of two adjacent frames is often misleading,
- there is considerable jitter in the luminance. As a consequence many histogram-based techniques (e.g., for cut detection) fail.
- Films are often shaky, because hand-held cameras are used, making motion-based analysis much more difficult.
- In addition early camera men often made recording mistakes, e.g., the camera was pointed to the ground, early film editors did not notice them or ignored them.
- Mistreatment in laboratories or early film projectors leads to scratches and stripes in the film.

We address these issues in our ECHO summarization paper.

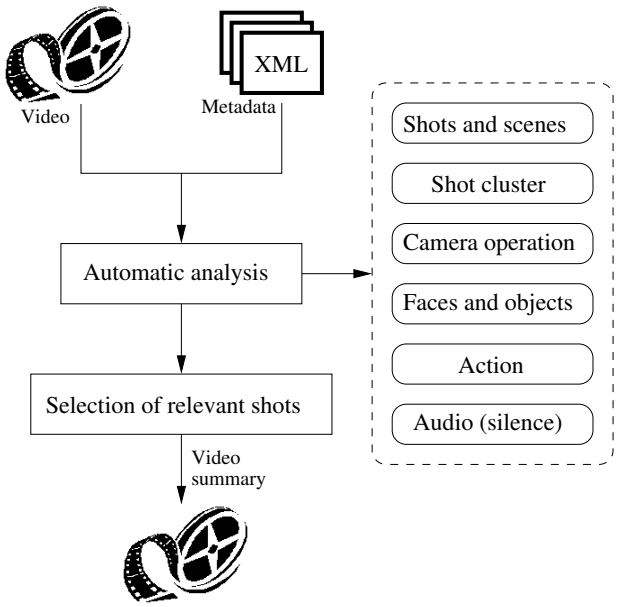

Figure 1: Overview of the ECHO video abstracting process.

3. ECHO SUMMARIZATION

In the ECHO abstracting system the generation of a video summary is done in two steps: first the video is analyzed and relevant features are calculated. In a second step the most relevant scenes and clips are assembled to form the summary. Figure 1 gives an overview of this process.

The automatic analysis algorithms extract syntactic and semantic information from the video. As in most other systems, shots and scenes define the basic structure of a video. A shot is defined as continuous camera recording, where as a scene is an aggregation of consecutive shots that have some common properties, usually the same physical location.

We have developed our own grouping mechanism that identifies shots with visual similarity. Similar shots are grouped to clusters. Consecutive shots are not required for clusters. The size of a cluster – defined as the number of frames of all shots in this cluster – indicates the relevance of this cluster. The size of a cluster is an indication for importance. At least one shot will be selected for the video summary from very large clusters.

Another important feature that we use to determine relevance is camera operation. Motion of the camera indicates more significant shots. Usually the specific object or person in the center of a camera zoom is important.

Although the understanding of the full semantics of an image or shot is unfeasible, it is possible to detect specific features or moving objects that are relevant for the summary. We have developed special modules to detect frontal faces and identify specific moving objects (e.g., cars or walking people). Often, a large face is an indication of an important person, such as a main actor in a feature film or a politician in a documentary.

Another useful criterion for relevance is *action intensity:* The more motion we find in a shot the more of it we need in the summary.

Motion can be object motion or camera motion. In the analysis phase we automatically detect camera operations, moving objects and general action intensity, and we use these features to determine the relevance of shots and scenes in the synthesis phase.

After completing the shot selection the summary videos in MPEG-I or MPEG-II format is created. The digital video is stored in the ECHO archive, to be accessible to all users of the ECHO system.

3.1 Feature Extraction

3.1.1 Shot Boundary Detection

The first step in making a film is the recording of the shots. Then the shots are edited, the order is determined, and transitions between the shots are added. It took a major effort to create advanced transitions such as fades, dissolves or wipes in historical films since every such transition had to be created manually. In comparison to modern TV or movie productions, advanced transitions were rarely used.

Shots define the basic structure of a film and constitute the basis for the detection of most other semantic features like moving objects or faces. Over 92 percent of the transitions in our 200 hour collection of historical videos are hard cuts. The fades amount to about 5 percent, 3 percent are dissolves. Other transitions, such as wipes, do not occur.

Our shot boundary detection algorithm identifies *hard cuts, fades* and *dissolves.* Because the noise and the number of damaged images in historical films are high, an analysis of histograms does not suffice to determine cuts reliably. We combine histograms with edge information and camera motion in order to detect shot boundaries.

We use quantized luminance histograms to compare frames (color does not play a major role in our historic material). The distance $D_{i,j}$ of frames i and j is defined as the sum of the absolute differences of corresponding histogram values.

In a first step, candidates for a hard cut are identified. A hard cut between frames i and $i+1$ is detected if

$$D_i > 2 \cdot \mu \cdot \max\{D_{j,j+1} : j = i-5 \ldots i+5, j \neq i\},$$

where μ is the average histogram difference of all neighboring frames in this film. A hard cut is detected if the histogram difference is significantly larger than the maximum histogram difference in the five-frame neighborhood of the analyzed frame. We use the five-frame neighborhood for the following reasons: Short-term changes in frames, such as those coming from flashlights or single-frame errors, should not be identified as hard cuts. Some reels of films are older than 80 years; the luminance in parts of those films changes significantly between frames. Histograms are very sensitive to these changes.

In order to improve the cut detection reliability we also compute the edge change ratio (ECR, [21]) between adjacent candidate frames. The ECR analyzes the number of edge pixels which appear (incoming edges) or disappear (outgoing edges) in two consecutive frames. The ECR is the normalized sum of outgoing and incoming edge pixels. Many edge pixels change at hard cuts, but luminance changes do not affect the ECR significantly.

Our detection of *fade-ins* and *fade-outs* is based on the standard

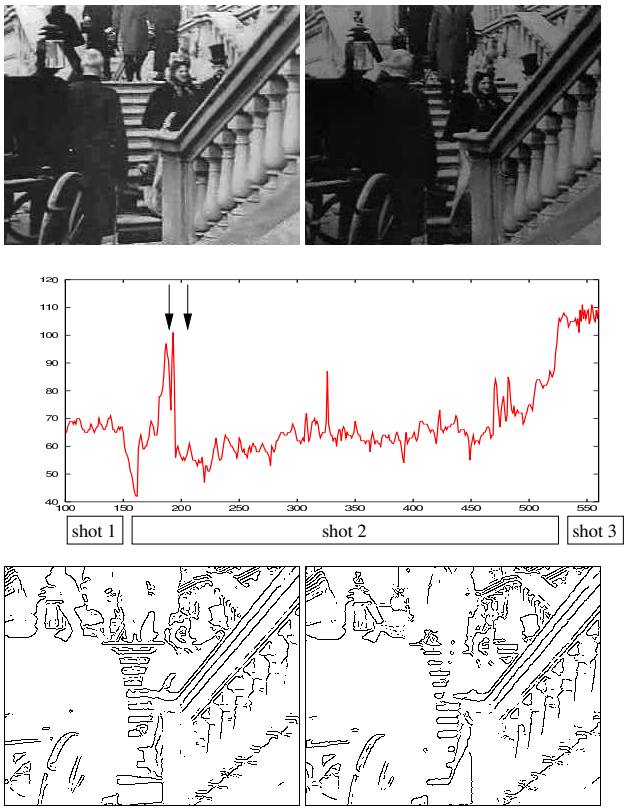

Figure 2: Two frames of a shot with average luminance values of 85 and 60. Whereas the histogram differences are very high the ECR is very low. The diagram in the center shows the average luminance for each frame in this shot.

deviation of luminance values for each frame: If the standard deviation decreases from frame to frame and the final frames are close to monochrome frames, we qualify the sequence as a fade-out. We validate a fade-out by also computing the ECR: The number of edges decreases in a fade-out, with many outgoing and no incoming edges.

A *dissolve* has characteristics similar to those of a fade. The standard deviation of the gray-scale values of the pixels in the middle of a dissolve is significantly lower than that at the beginning or end of it. As the significant edges disappear in the first part of the dissolve, the number of outgoing edges increases. In the second half of a dissolve that of incoming edges is much higher than the number of the outgoing edges.

If a fast horizontal or vertical camera operation occurs (pan or tilt) the images are often blurred. The blurring causes the standard deviation and number of edges to decrease. When the movement stops, the values increase again. To avoid classifying fast camera movements as dissolves, we analyze the camera model and explicitly eliminate fast camera movements. Figure 2 depicts two frames of a shot. The average gray-scale value changes significantly between these frames. On the other hand, the significant edges and ECR values remain very similar.

For each shot a representative key frame is selected. This is typically the center frame. To validate the key frame, its histogram is

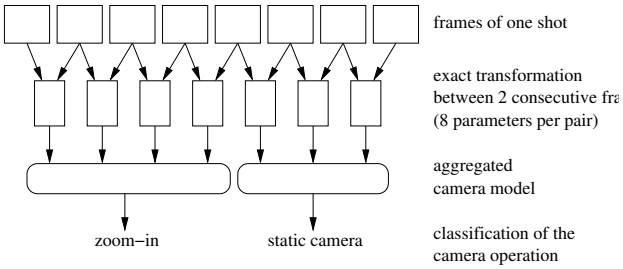

Figure 3: Calculation of the exact camera model and the aggregated camera model parameters. The classification of the camera operation is based on the aggregated camera model.

compared to the average histogram of all frames in the shot. If the difference is very high (e.g., a damaged image), another key frame is selected. The validation is required because damaged image occur frequently in historical films.

3.1.2 Camera Operations

Motion is one of the most important features of a film. We distinguish camera operations and object motion: if video is recorded with a moving camera, not only the objects in the foreground move, but those in the background as well.

We use a perspective camera motion model to describe the camera operations in a unified manner. The perspective model allows us to formulate different transformations between two frames: translation (pan or tilt of the camera), scaling (zoom-in or zoom-out), rotation, or perspective transformation. Eight parameters are sufficient for an exact description of the parameters of the perspective model:

$$x' = \frac{a_{11}x + a_{12}y + t_x}{p_x x + p_y y + 1} \qquad y' = \frac{a_{21}x + a_{22}y + t_y}{p_x x + p_y y + 1}$$

The parameters a_{ij} define the affine transformation, t_x and t_y the translation, and px and p_y the perspective transformation.

The camera model is calculated at two different levels; an exact model, and an approximation, that aggregates the camera parameters of several frames within one shot. The approximation ignores the jitter of the hand-held cameras. The *exact camera model* calculates the eight transformation parameters of two consecutive frames. These parameters are used to detect and segment the moving objects within a shot (see Section 3.1.6 below).

Frames with similar camera parameters are grouped, and an *aggregated camera model* is determined. If a shot contains more than one significant camera operation the shot is split, and a new aggregated camera model is calculated for each part. This approach removes errors in single frames and the jitter effect of hand-held cameras. The classification of a camera operation (e.g., zoom-in, pan, tilt) is based on the aggregated camera model. Figure 3 depicts the calculation of the camera operation. The eight parameters of the perspective camera model of pairs of consecutive frames are calculated first (exact camera model).

To calculate the camera parameters we use a feature-based parameter estimate [17]. The idea is to identify a set of positions (features) in a frame that can be tracked throughout the shot. An example would be the corners of a building. If such features can be well localized, image motion can be estimated with high confidence. On the other hand, for pixels inside a uniformly colored region, camera motion cannot be determined; to be able to track a feature reliably, a corner is required.

The Harris corner detector [8] is employed to select appropriate feature points. Once the corners are identified, we establish correspondences between corners in successive frames. In order to estimate camera parameters reliably from a mixture of background and object motion, we apply a robust regression method to estimate the eight parameters of the perspective camera model. The details of this approach are described in [7].

3.1.3 Shot Clustering

We define cluster as a syntactic grouping of frames based on a similarity measure, in contrast to a scene that is a semantic grouping. The shot-clustering module detects shots with similar visual content and groups them into clusters. The size of each cluster is an indication of its importance. At least one shot from each very large cluster should be included in the video summary.

For each key frame a feature vector is extracted. The distance measure of the selected feature must correspond to visual similarity. We use quantized luminance histograms for 9 equal-sized regions of the key frame as feature vectors. The distance measure is the sum of absolute differences.

In the process of clustering we create a certain number of cluster centers. Each cluster center and key frame is represented as histogram, that describes a position in a multi-dimensional space. The idea is to add new cluster centers till the distance of all key frames to the nearest center is very low. If the distance of a key frame is above a threshold value, a new cluster center is added.

The clustering algorithm is an iterative process that can be summarized as follows:

1. The first cluster center is initialized as the average histogram of all key frames. The summarized distances of the cluster center and all key frames is minimal at this position in the multi-dimensional space.

2. The nearest cluster center for each key frame is located (in the first iteration, we have just one cluster center). The key frame is attached to the nearest cluster, and the distance between cluster and key frame is calculated and stored.

3. The position of each cluster center is updated. The new position is the average histogram value for all key frames now attached to this cluster.

4. The key frame with the maximum stored distance is selected. If the distance is significant (i.e., above a given threshold), a new cluster center is added at the position of this key frame, and the algorithm continues with step 2.

The algorithm terminates if all key frames within one cluster are similar. It is possible that the number of shots and clusters in very

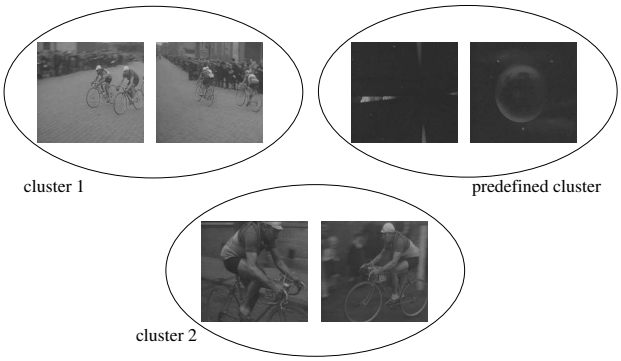

Figure 4: Three shot clusters, each with two key frames, are depicted. The right cluster was initialized with a predefined cluster center (near-black). The shots in the right cluster will be discarded from the summary.

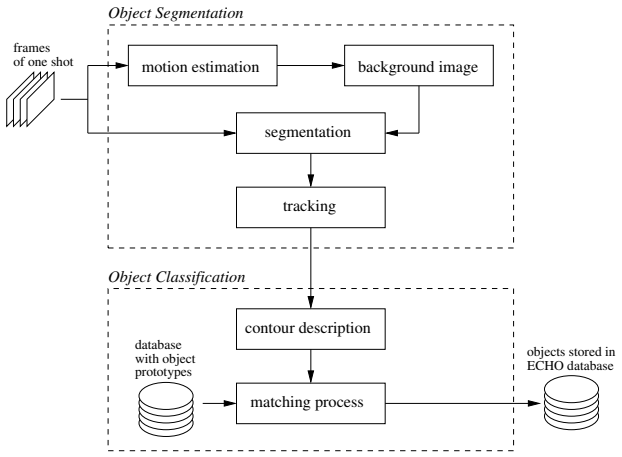

Figure 5: Overview of the object recognition process.

short films are the same. This will be the case if the shots differ significantly.

Many frames and shots in historical films are damaged. It is very important that these shots not be selected for the video summary. The clustering algorithm can be modified to identify them: Cluster centers are initialized with predefined shots that should *not* be part of the video summary; we call them *delete clusters.* Typical delete cluster centers are black, gray, or white frames. If a shot of a video is attached to a delete clusters, the selection process will discard them.

Let us consider an example. Figure 4 depicts three clusters, each containing two shots. The right cluster is a delete cluster; it was initialized with a predefined near-black cluster center.

3.1.4 Scene Boundary Detection

A *scene* or *act* is a longer story-telling unit. In contrast to a cluster a scene is a semantic unit of the video. Typically, the background in all shots of a scene is similar. A *dialog* is a special scene where the camera switches back and forth between two or more persons.

It is not necessary to select an entire scene for a video summary; usually, it is sufficient to select two or three representative shots. Early user experience shows that we have to be careful with the audio when determining the boundaries of a sequence to be included in the summary: If speech is cut within a sentence, the effect is very annoying. If silence cannot be detected near the borders of the selected shots, the next shot is added or a fade-in or fade-out of the audio is automatically generated.

Identification of the scene boundaries is based on the results of shot clustering. Usually the visual similarity of frames within a single scene is high because the background in the frames is similar. Even a change of the camera angle has no significant influence on the main background color. The scene detection algorithm searches for successive shots from up to two different shot clusters.

3.1.5 Face Detection

Persons are very important in most types of video, and very much so in documentaries of historical value. Close-up views of the faces of main actors are important in feature films, whereas historical documentaries often feature sports persons, politicians, etc. Shots where faces take up much of the screen are prioritized in our selection process.

Rowley, Baluja and Kanade [16] have developed a famous, very reliable face recognition algorithm based on a neural network. The algorithm detects about 90 percent of the frontal faces in a video. Non-face areas (i.e., false hits) are rare. We have re-programmed the face detector and trained our own network with more than $7,500$ faces [10]. We were able to reproduce the good detection results. We have extended the algorithm from [16] to detect slightly tilted faces (+/-30 degrees).

A second processing step tracks the faces within a shot. The tracking allows us to find single skipped faces and removes most of the false hits (mis-classified face regions). The tracking analyzes all detected faces in a shot. If one face could be detected, the position and size of the face is estimated for the next frame by the global camera motion. The tracking increases the reliability of the face detection algorithm with only a very small increase of computation time.

3.1.6 Recognition of Moving Objects

Moving objects deliver additional semantic information. If the same moving object is visible in many shots, it should also be visible in the summary. The number of moving objects in a video is also an indicator for motion intensity. A film of a car race or a tennis match repeatedly shows moving cars or tennis players. The selection algorithm will assign a high priority to shots containing these identified moving objects.

Our object recognition algorithm consists of two components, a segmentation module and a classification module. Figure 5 depicts the main recognition steps. The motion of the camera is estimated in a first step (see Section 3.1.2 above). The parameters of motion estimation are used to construct a background image for the entire sequence. During construction of the background, foreground objects are removed by means of temporal filtering. Object segmentation is then performed by evaluating differences between the current frame and the constructed background mosaic.

Many frames in historical videos are noisy. On the other hand, the object segmentation algorithm is very sensitive to this noise since it is based on image differences. To reduce the effect of incorrectly

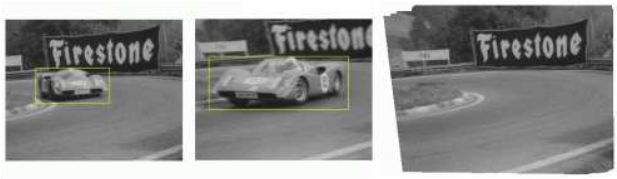

Figure 6: The images on the left side show two shots of a scene. The automatically segmented and classified objects are marked in these frames. The constructed background image is on the right side.

detected object areas, a tracking algorithm is applied to the object masks. Only objects that could be tracked through several frames of the shot are kept for further processing.

The classification module analyzes the segmented object masks. For each mask, an efficient shape-based representation is calculated (*contour description*) [15]. A curvature scale space (CSS) image is used to describe a contour. The CSS technique is based on the idea of curve evolution and provides a multi-scale representation of the curvature zero crossings of a closed planar contour. The CSS has been selected to describe shapes in MPEG-7. The matching process compares these contour descriptions to pre-calculated object descriptions stored in a database. The matching results for a number of consecutive frames are aggregated. This adds reliability to the approach since unrecognizable single object views occurring in the video are insignificant with respect to the entire sequence. The position, size and name of the detected and classified objects are stored in the ECHO database and made available for search queries.

Figure 6 depicts two sample frames from a shot of a car race and the automatically constructed background image. The segmented and classified object (*car*) is marked with a rectangle. A detailed description of the segmentation and classification algorithm can be found in [7, 15].

3.1.7 Action Intensity

We consider action intensity to be another relevant parameter for selecting shots for the summary. Three factors are responsible for significant changes between consecutive frames: A fast camera operation changes the visible area significantly, especially if the motion of the camera is fast. A fast-moving camera is often combined with moving objects or persons. Also, special events like fire or explosions are often the cause of significant and fast changes between frames.

But experience shows that naive inter-frame differences lead to a large number of false hits. For example when analyzing a shot where the light is switched on, the difference between two frames is high. In contrast, fire or explosions produce high differences between *all* frames in a shot. The changes based on camera motion, objects or special events are classified as *action*. Shots with a high action intensity are handled in a special way in the summarization process.

We apply two measurements to detect the action intensity of a shot. The first is the summarized absolute pixel difference in two consecutive frames. The second estimates the average motion vector length based on the calculated camera motion model (see 3.1.2 above). Both values are normalized, and the values of all frames in a shot are aggregated.

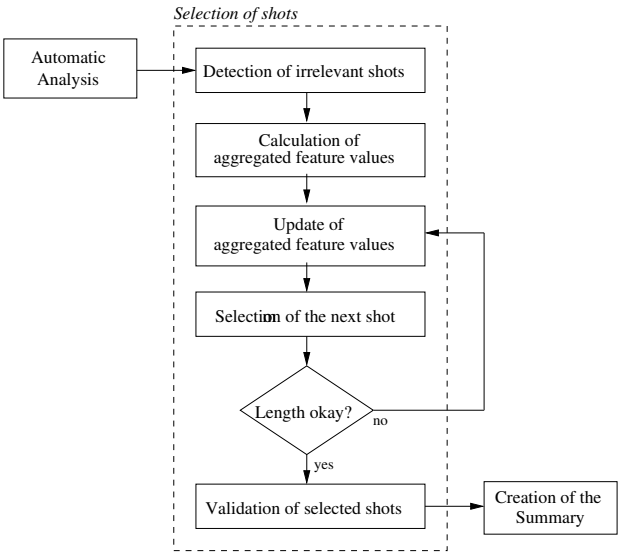

Figure 7: Overview of the selection of shots.

The correlation between both action measures is very high, although changes in the luminance (fire, explosion) have a larger impact on the first measure and camera motion affects the second measure more.

3.1.8 Analysis of the Audio Track

If audio is cut at random positions, the effect is very disturbing. Many sentences are split even though the audio is cut at shot boundaries. Cutting a sentence reduces the understanding of the spoken text significantly.

We have limited the analysis of the audio track to the detection of silent areas. It should be possible to cut the audio in these areas without generating many broken sentences.

The quality of the audio signal differs significantly in historical films. Part of the ECHO collection includes silent movies and films with music, text or background noise. The noise level is always very high.

We detect the silent areas by analyzing the energy of the audio track. The energy values are smoothed by a median filter to eliminate noise peaks. We define a time interval as silent if low energy values occur for more than 1 second. Possible intervals for audio cuts are provided to the selection process, which in described in the following Section.

3.2 Selection of Shots for the Summary

For each video in the ECHO collection additional meta-data information is stored in a database. If a feature value is available in the ECHO database it will be used. Otherwise the feature values are calculated and stored in the database.

Figure 7 depicts the main steps of the selection process. In a first step irrelevant shots are identified. Shots that have been attached to predefined cluster centers or very short shots (less than three seconds) are deleted from the list. A user cannot remember the content of very short shots.

3.2.1 Aggregated Feature Values

We calculate aggregated feature values to make the different features comparable, e.g., the face detection process recognizes the position, size and rotation angle of faces and the camera operation detects the type of camera operation and the motion speed. The aggregated feature value characterizes a feature on the level of shots. Each aggregated feature value is normalized to the interval $[0, 1]$.

Table 1 lists the features that are relevant for the selection process of the summary. The stored meta-data information and the length of the video segment is listed for each feature. Most aggregated feature values are initialized once and a modification is not required during the selection process. Other feature values depend on previously selected shots. They are updated when a new shot is selected. For each feature and shot an aggregated value is calculated in a first step.

Faces
The aggregated face value is the normalized quotient of face pixels to all pixels in a frame. With our definition the relevance of two medium sized faces is similar to the relevance of a large face. The average value of all frames in a shot is stored as aggregated face value.

Moving objects
Our moving object classification algorithm detects planes, boats, cars and people. At the beginning to the middle of the last century cars, boats or planes were much more special than they are today. Very few people could afford to buy a car or take a trip with a plane. Moving objects may indicate a special event.

A high aggregated value for moving objects is used, if planes, boats or people could be detected in a shot. On the other hand, it is not possible to characterize a moving car in a single shot of a film as special event. Only if many cars can be recognized they are of great importance to the film (e.g., car racing or politicians arriving at a meeting). If cars can be recognized in more than 10 percent of the shots in a film, the value of these shots is set to its maximum.

The aggregated value for moving objects is determined by the number of recognized objects in a shot, the size of the objects and the reliability of the recognition.

If the recognition of an object is possible, we known that the quality of a shot is high. A background image cannot be constructed with blurred frames and noise prevents an exact segmentation. Due to the good quality of the shots, the aggregated value for moving objects is increased. If an object could be detected, the normalized aggregated value for shots is set to the interval $[0.5, 1]$,

Camera Operations
A zoom-out, pan or tilt introduces a location, where the following action takes place. Typically the countryside, a building or a room is recorded. During a zoom-in an important person or relevant object is in the center of the view.

The aggregated value for camera operations is a function based on the type of operation (a zoom-in is most significant), the length of the motion vectors, and the duration of the operation. If the camera is static at the end of the shot for several seconds, the aggregated value is increased.

Action
The action value is the normalized sum of the two action values based on frame differences and motion (see Section 3.1.7). The aggregated action value is the average of these values of all frames in a shot.

Contrast
It is very hard to recognize the content of shots with a very low contrast. We analyze the contrast to avoid the selection of these shots. The aggregated contrast value is the average contrast of all frames in a shot.

Shot cluster
The aggregated values, that have been described so far, are initialized once and the values do not change. It is necessary to update the values of shot clusters, scenes or position values if a new shot has been selected.

The relevance of a cluster C_i depends on the length of all shots that have been attached to cluster i:

$$C_i = \frac{L_i}{max\{L_j\}} \cdot \frac{1}{1+S_i}, \qquad j = 1 \ldots N,$$

where L_i is the summarized length of all shots of cluster i, S_i is the number of already selected shots from this cluster and N is the total number of clusters.

Scenes
For the better understanding of the content of a scene at least two consecutive shots should be selected. The summary will show more redundant information if many shots from one scene are selected. Consecutive shots should be chosen to avoid broken sentences.

The calculation of the aggregated scene values is done in several steps: the values are initialized with an average value of 0.5 first. This value is reduced, if two or more shots have been selected in a scene. If one shot has been selected the values of neighboring shots are increased and all other values are decreased. More consecutive shots are selected with this approach.

Distance
A major goal of a video summary is to give an overview of the full video. The shots should be selected from all parts of the video. A summary of a feature film may have a different goal, because the thrilling end of a film should not be revealed.

The $distance$ value tries to distribute the selected shots among the full length of the video. The value calculates the distance from a shot to the next selected one. This distance is normalized to the interval $[0, 1]$. Figure 8 depicts a video with three selected shots and the aggregated distance values for the shots.

3.2.2 Selection of Shots

The selection process uses the aggregated feature values. The total relevance R_i for each shot i is defined as

$$R_i = \alpha_i \cdot F_i, \qquad \sum \alpha_i = 1.$$

We have used fixed weights ($\alpha_i = \frac{1}{8}$) in our implementation, although a user can define individual weights.

The selection algorithm is an iterative process as depicted in Fig-

feature	stored meta-data information	video level	update aggr. feature value
faces	size, position, rotation angle	frame	
moving objects	size, contour description, object name, reliability	frame	
camera operations	type (pan, tilt, zoom, ...), motion speed	part of a shot	
scenes	list with shot numbers	shot	yes
action	difference and motion-based action value	frame	
contrast	contrast value	frame	
distance	distance to the next selected shot	shot	yes
shot cluster	list with shot numbers	shot	yes
audio	time codes of silent parts	part of the video	

Table 1: Stored meta-data information for each feature. An aggregated feature value is calculated for each feature and shot.

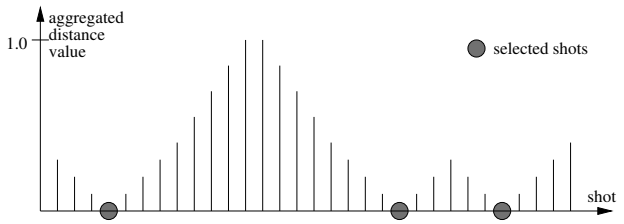

Figure 8: Video with three selected shots. The aggregated distance value is depicted for each frame.

ure 7. When the feature values have been calculated, the shot with the maximum total relevance R_i is selected for the summary. The algorithm stops, if the summary has the desired length. Otherwise the feature values are updated and the next shot is selected.

3.2.3 Validation of Selected Shots

The presentation of the selected shots is very important for the acceptance of the video summary. Some constraints must be regarded to avoid disturbing effects. The most important constraints are:

- Too many subsequent camera operations (e.g., a zoom-in followed by a pan followed by a zooms-out) within one shot are visually unpleasant: professionally created films avoid such camera operations. Two shots with significant camera operations should be separated by at least one shot with a static camera.
- At least two shots should be selected from a scene. These shots should be consecutive.
- Only silent areas should be used to cut the audio.
- The average action of the summary should not be significantly higher than the action of the full video. Especially in films with much action (e.g., films from the World Wars) a validation of the action intensity is required. Otherwise the probability would be very high, that nearly all selected shots of the summary have a very high action intensity.
- The length of the summary should be similar to the length specified by the user.

The length of the summary can be defined by the users of the ECHO system as an absolute or relative value. If no length is specified, it

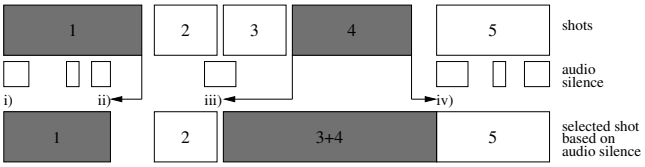

Figure 9: Selection of the exact cut position based on silence. Selected shots are marked in gray color. i) shot boundary and audio match, ii) next silent area is within shot 1 (frames are deleted), iii) next silent area is at the beginning of shot 3 (shot 3 is added), iv) next silent area is near the end of shot 4 (frames are added).

is set to a predefined value depending on the length of the original film. In our collection of historical films many short clips with a length of 2-5 minutes are stored. Using a fixed percentage to set the length of the summary does not make much sense. A default length of 10 percent would create a summary of 12 seconds from a 2-minutes clip. Only videos with a length of at least three minutes are summarized. The predefined length of the summary is based on a linear function. The default length of summaries varies between 1 and 10 minutes.

If constraints are violated one or more shots are removed, added or replaced. This depends on the current length of the summary. All constraints are checked iteratively until all violations have been solved.

The audio is very important for the acceptance of video summaries. Speech and music should not be cut at random positions. The audio is not changed between two consecutive shots. If we cut a film, the next position of silence is located. To keep the audio and video synchronized frames of the video are removed or added. If several frames have to be added a fade or dissolve is used.

If the next shot is very long (> 30 seconds) we search a silent area within this shot. We add or remove this shot otherwise. If no silent area at the position of the new shot boundary is detected, we add a fade-in or fade-out of the audio track. Figure 9 depicts the selection of the audio cuts. It is necessary to move the other shot boundaries.

3.3 Creation of the summary

The last step selects the transitions between the shots and creates the summary. The transitions of the summary and the film should be similar. E.g., if the film uses many dissolves, they should be

chosen as transition for the summary, too. The user can modify the frame-rate, resolution or bit-rate. E.g., if a user wants to create MPEG-I summaries with a lower resolution from high-resolution MPEG-II videos, he can specify the required parameters and the summary will be generated.

4. EXPERIMENTAL RESULTS

Within the scope of the project *ECHO* a system has been developed to store and manage large collections of historical films. Four major film archives have selected very precious historical films. More than 100.000 hours of historical films are stored in these archives. Several other research institutes (e.g., the Carnegie Mellon University in the United States or Consiglio Nazionale delle Ricerche (CNR) in Italy) have developed tools, that have been included in the ECHO system.

The ECHO system has stored more than 4500 films from 1920 to 1965 so far. The length of a film varies between 1 and 60 minutes. Summaries have been created for films with at least three minutes length. For each new film, meta-data information is calculated and a summary is generated automatically.

The calculation of meta-data requires a lot of computational effort. If the calculated meta-data information is available a new video summary with other parameters can be created very fast (real-time) on a personal computer that is up to date. A user may want to specify a different length or other weights for the features. E.g., if a user wants to see a summary with all faces or many moving cars he can create a special summary.

The shot boundary detection algorithm is very reliable due to the combination of different approaches (histograms, edges and camera operation). We have analyzed the reliability of the shot boundary detection algorithm with random selected films with a total length of more than one hour. More than 91 percent of all cut boundaries can be detected with 2 percent false hits. A simple histogram-based approach detects 70 percent with an error rate of 10 percent. Our approach is very reliable for noisy or damaged films.

The estimation of the camera operation is very precise. Otherwise it would not be possible to calculate background images of shots for the object recognition. Errors occur if large foreground objects are present or in case of blurred background images.

Very few scenes can be found in historical documentaries. It is more common, that shots with the same common location are distributed over the entire length of the video. Our algorithm detects about 60 percent of the scenes. The failure rate is high due to missing color in historical films.

Our face detection system is very reliable. About 90 percent of the faces with a width and height of at least 25 pixels can be detected.

The detection of moving objects is much more complicated. The recognition rate in shots with one car or one person is acceptable (about 40 percent). It is much lower for planes or boats due to changing background (water) or very few edges in the background (e.g., sky with some clouds). Many objects could not be detected, but nearly no wrong classification occurs. The probability to detect an object is very low, if:

faces	0.38	moving objects	0.00
scenes	0.50	camera operations	0.00
action	0.20	distance	0.55
contrast	0.91	shot cluster	0.84
total	3.38		

faces	0.00	moving objects	0.00
scenes	0.50	camera operations	0.00
action	0.91	distance	1.00
contrast	0.94	shot cluster	0.69
total	4.04		

faces	0.00	moving objects	0.00
scenes	0.50	camera operations	0.00
action	0.09	distance	0.48
contrast	0.32	shot cluster	0.53
total	1.92		

Figure 10: Three key frames of a circus film from 1942. The first two shots have been selected for the summary.

- more than one object moves in the shot,
- the object is very large,
- the background is blurred,
- noise or jitter in the luminance occurs, or
- the object is partially occluded.

Figure 10 depicts key frames of three shots of one film. The first two shots were selected for the summary. The aggregated values for the features are displayed for each shot.

sky Within the last year we have received feedback from our partners of the ECHO project and made some local tests. Two major problems were reported: shots were selected for the summary, that did not show anything meaningful. The major similarity of all these shots is a very low contrast. That was the reason why we added the contrast measure.

The second problem is the audio track of the summaries. It is very disturbing, if a sentence or music is interrupted. Due to the noise in the audio track a reliable recognition of words is nearly impossible and the end of sentences cannot be detected. A possible solution might be to fade-in and fade-out the audio. Another aggregated feature could be added to prioritize the selection of consecutive shots. Additional research is required in this area.

The selection of the shots was mostly good. In some cases important parts of the film were missing and the understanding of the content of the summaries was very difficult. This is a typical problem of very short summaries.

A large evaluation with expert users of the archives is planned. 20 professional users, that work at archives, will test the ECHO system

at different installations. We expect very important results of this study, because the people who work in the archives can judge much better, whether the video summaries may be useful for their daily work. The tests will be finished at the end of April 2003 [the results of this evaluation will be included in the final version of this paper].

5. CONCLUSIONS

It is difficult to solve the problem about the quality of a video summary. Objective criterions have to be regarded (e.g., the audio should not be cut within a sentence) but the selection of shots is very subjective and an optimal summary is not possible. Two persons will select different shots from a long documentary, because they rate their importance differently. An automatic generated summary makes a third – not necessarily optimal – selection.

Acknowledgments

The work of Stephan Kopf was financially supported by the European project ECHO. We thank the colleagues in the ECHO project – especially from Instituto Luce, Memoriav, NAA and INA – for the film material.

6. REFERENCES

[1] A. Amir, D. Ponceleon, B. Blanchard, D. Petkovic, S. Srinivasan, and G. Cohen. Using audio time scale modification for video browsing. In *IEEE 33rd Hawaii International Conference on System Sciences*, volume 3, 2000.

[2] A. Aner and J. R. Kender. Video summaries through mosaic-based shot and scene clustering. In *Proc. European Conference on Computer Vision*, 2002.

[3] J. Boreczky, A. Girgensohn, G. Golovchinsky, and S. Uchihashi. An interactive comic book presentation for exploring video. In *CHI 2000 Conference Proceedings*, pages 185–192. ACM Press, 2000.

[4] M. G. Christel, A. G. Hauptmann, H. D. Wactlar, and T. D. Ng. Collages as dynamic summaries for news video. In *Proceedings of the 2002 ACM workshops on Multimedia*, pages 561–569. ACM Press, 2002.

[5] M. G. Christel, A. G. Hauptmann, A. S. Warmack, and S. A. Crosby. Adjustable filmstrips and skims as abstractions for a digital video library. In *Proceedings of the IEEE Advances in Digital Libraries Conference*, pages 98–104, 1999.

[6] D. Farin, W. Effelsberg, and P. H. N. de With. Robust clustering-based video-summarization with integration of domain-knowledge. In *IEEE International Conference on Multimedia and Expo (ICME)*, volume 1, pages 89–92, 2002.

[7] D. Farin, T. Haenselmann, S. Kopf, G. Kühne, and W. Effelsberg. Segmentation and classification of moving video objects. *Handbook of Video Databases*, pages 561–591, 2003.

[8] C. Harris and M. Stephens. A combined corner and edge detector. In *Proc. Alvey Vision Conference*, pages 147–151, 1988.

[9] F. C. Li, A. Gupta, E. Sanocki, L. wei He, and Y. Rui. Browsing digital video. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 169–176. ACM Press, 2000.

[10] R. Lienhart. Verfahren zur Inhaltsanalyse, zur Indizierung und zum Vergleich von digitalen Videosequenzen, Dissertation, Universität Mannheim (in German),1998 .

[11] R. Lienhart. Dynamic video summarization of home video. In *Proceedings of the SPIE, Storage and Retrieval for Media Databases 2000*, volume 3972. SPIE, 2000.

[12] R. Lienhart, S. Pfeiffer, and W. Effelsberg. Video abstracting. In *Communications of the ACM*, volume 40, pages 55–62, 1997.

[13] T. D. Ng, H. D. Wactlar, A. G. Hauptmann, and M. G. Christel. Collages as dynamic summaries of mined video content for intelligent multimedia knowledge management. In *AAAI Spring Symposium Series on Intelligent Multimedia Knowledge Management*, 2003.

[14] N. Omoigui, L. He, A. Gupta, J. Grudin, and E. Sanocki. Time-compression: systems concerns, usage, and benefits. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 136–143. ACM Press, 1999.

[15] S. Richter, G. Kühne, and O. Schuster. Contour-based classification of video objects. In *Proceedings of SPIE, Storage and Retrieval for Media Databases*, volume 4315, pages 608–618, January 2001.

[16] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 20, pages 23–38, 1998.

[17] P. Torr and A. Zisserman. Feature based methods for structure and motion estimation. In B. Triggs, A. Zisserman, and R. Szeliski, editors, *Vision Algorithms: Theory and Practice, International Workshop on Vision Algorithms*, volume 1883 of *Lecture Notes in Computer Science*, pages 278–294, Berlin, Heidelberg, 1999. Springer.

[18] S. Uchihashi, J. Foote, A. Girgensohn, and J. Boreczky. Video manga: Generating semantically meaningful video summaries. In *Proceedings ACM Multimedia*, pages 383–392. ACM Press, 1999.

[19] H. D. Wactlar. Informedia – search and summarization in the video medium. In *Proceedings of Imagina*, 2000.

[20] M. M. Yeung, B.-L. Yeo, and B. Liu. Extracting story units from long programs for video browsing and navigation. In *Proc. IEEE International Conference on Multimedia Computing and Systems*, pages 296–305, 1996.

[21] R. Zabih, J. Miller, and K. Mai. A feature-based algorithm for detecting and classifying scene breaks. In *Proceedings ACM International Conference on Multimedia*, pages 189–200. ACM Press, 1995.