# A Survey on Video-based Graphics and Video Visualization

R. Borgo[1][†], M. Chen[1], B. Daubney[1], E. Grundy[1], H. Jänicke[3], G. Heidemann[2], B. Höferlin[2], M. Höferlin[2], D. Weiskopf[2], X. Xie[1]

[1]Swansea University, UK
[2]Stuttgart University, Germany
[3]Heidelberg University, Germany

**Abstract**

*In recent years, a collection of new techniques, which deal with videos as the input data, emerged in computer graphics and visualization. In this survey, we report the state of the art in video-based graphics and video visualization. We provide a comprehensive review of techniques for making photo-realistic or artistic computer-generated imagery from videos, as well as methods for creating summary and/or abstract visual representations to reveal important features and events in videos. We propose a new taxonomy to categorize the concepts and techniques in this newly-emerged body of knowledge. To support this review, we also give a concise overview of the major advances in automated video analysis, as some techniques in this field (e.g., feature extraction, detection, tracking and so on) have been featured in video-based modeling and rendering pipelines for graphics and visualization.*

## 1. Introduction

Until recently, videos have largely been used only as an *output* medium in computer graphics and visualization. The rapid advance of digital technologies has resulted in an explosion of video data, which stimulates the needs for creating computer graphics and visualization from videos. In this survey, we report the emergence of a new collection of graphics and visualization techniques, which deal with videos as the *input* data.

*Video-based graphics* is concerned with the manipulation and rendering graphical models, which are built from video data, instead of, or in addition to, traditional object representations. Its primary aim is to make creative computer-generated imagery from videos for artistic appreciation and entertainment. There are two main strands in this field, video re-fashioning and video-based scene modeling. The former typically involves manipulation of the geometrical entities (e.g., object shape and distribution) and optical attributes (e.g., lighting, color) in an input video, and produces a new video that captures the essence of the input video but in an expressive art form, such as relighting the video scene with imaginary lights or mimicking hand-drawn cartoon animation. The latter typically involves reconstruction a 3D object or scene model captured by the input video, allowing such a model to be manipulated, combined with other models, and rendered in the same way as conventional graphical models. The main thrust for video-based graphics has been consumer multimedia applications, and the films and games industries.

*Video visualization* is concerned with the creation of a new visual representation from an input video to reveal important features and events in the video. It typically extracts meaningful information from a video and conveys the extracted information to users in abstract or summary visual representations. Video visualization is not intended to provide fully automatic solutions to the problem of making decision about the contents of a video. Instead, it aims at offering a tool that assists users in their intelligent reasoning while removing the burden of viewing videos. This aim justifies deflection from creation of realistic imagery, and allows simplifications and embellishments, if they improve the desired understanding of the input video. In many ways, the subject of video visualization encompasses some aspects of video-based graphics. Nevertheless, the development of the subject has been heavily influenced by many applications in science, medicine, sports and security.

There is a huge collection of literature in the fields of image processing, computer vision, and multimedia technology. Automated video analysis encompasses a variety of techniques, ranging from low-level processing techniques

---

[†] Authors are listed in alphabetical order. Contact author is Dr. Rita Borgo, email: r.borgo@swansea.ac.uk
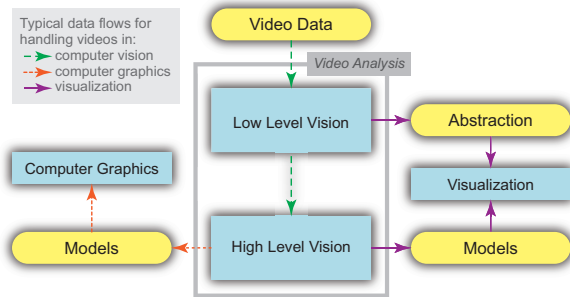
**Figure 1:** *Typical computational pipelines for video-based graphics, video visualization and video analysis. This survey focuses on the first two fields while giving a brief overview of techniques in video analysis.*

for filtering, enhancement, motion flow estimation, image segmentation and feature extraction to high-level analytical techniques for object and event detection and recognition, tracking and 3D reconstruction. Automated video analysis is fundamentally different from video-based graphics and video visualization. The low-level techniques typically result in an output video as a more cost-effective, informative or usable representation than the input. The high-level techniques typically result in a binary or probabilistic decision in relation to a classification, or 3D measurements and models of objects and scenes captured on videos.

Figure 1 illustrates three typical data flows of video-based graphics, video visualization, and video analysis. We can easily observe that these three fields share a substantial amount of functional components, while having dissimilar aims. This survey focuses on video-based graphics and video visualization. In order to provide readers with a brief background about various functional components that are typically published in the literature of image processing, computer vision and multimedia technology, we also provide an overview section on video analysis.

## 2. Taxonomy

Video-based graphics and video visualization are relatively new developments in visual computing. It is thus highly desirable to establish a means for categorizing different technical contributions in the literature. A taxonomy is usually defined upon one or several classification attributes that differentiate entities (e.g., concepts or methods) in a body of knowledge. For video-based graphics and video visualization, such attributes may include (i) the principal goal of a method, (ii) the data type of the output, (iii) the additional information that accompanying the input video, and (iv) the levels of automation.



(a)        (b)
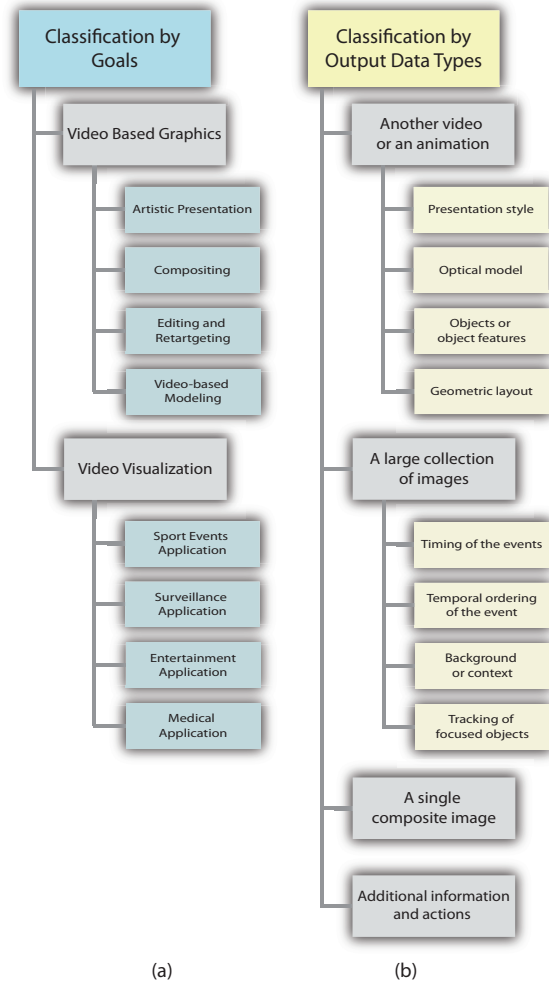
**Figure 2:** *First and second classifications proposed by our Survey: a) by goals; b) by type of output data.*

### 2.1. Classification by Goals

As stated in Section 1, *video-based graphics* differs from *video visualization* primarily by their goals. This enables us to define two distinguishable categories:

A1. **Video-based graphics** — to make use of video content in creating computer-generated imagery for artistic appreciation and entertainment.
A2. **Video visualization** — to provide users with a tool to aid their intelligent reasoning while removing or alleviating the burden of viewing videos.

Video-based graphics can be further categorized by different intents as:

A1.1. *Artistic presentation* — focuses on altering the presentation of a video by rendering it in different styles, typically mimicking a non-photo-realistic
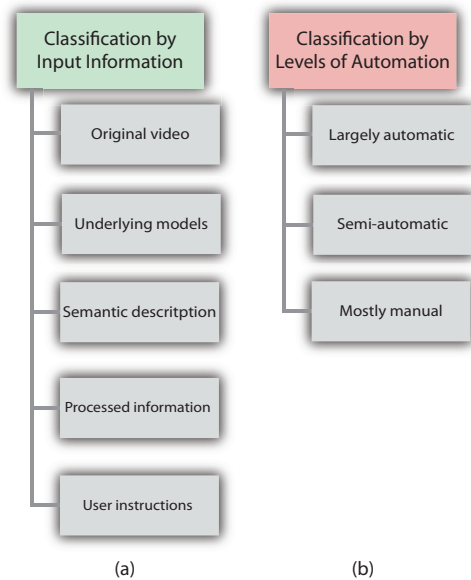
**Figure 3:** *Third and fourth classifications proposed by our Survey: a) by input data; b) by level of automation.*

hand-painting technique and transforming a video to a more "expressive" cartoon representation.

A1.2. *Compositing* — focuses on creating an artistic image by selectively mixing contents from different frames of a video. In some cases, multiple viewing perspectives are mixed in the same composition, akin to cubist artworks. In other cases, objects in different temporal steps are mixed together, mimicking the dynamic characteristics of some futurist artworks.

A1.3. *Editing and retargeting* — focuses on altering video content to accommodate specific display constraints (e.g., empty space removal) or to allow coherent integration into a visual context (e.g., re-lighting). Although editing and re-targeting can be done on a frame-by-frame basis, video-based approach addresses the needs for temporal coherence, while offering a more efficient means of graphical interaction and processing.

A1.4. *Video-based modelling* — focuses on creating graphical models from videos to enhance the perception of spatial and dynamics feature of a scene. This ranges from video-based panorama composition to 3D object reconstruction. The primary use of this class of techniques is the modeling of virtual environments.

The goals of video visualization can be further classified according to those of the applications. For example, for sports applications, the goals may include detecting key events, depicting team formation, and summarizing statistical patterns of a game. For surveillance applications, the goals may include depicting signatures of typical events, detecting anomalies, and tracking important movements. Although many developments in video analysis also aim at

these goals, the computer vision has not yet been able to deliver automated technology to fulfill such goals in most situations. Video visualization, which keeps the user in the loop, is complementary technology to bridge the gap. By removing and alleviating the time-consuming burden of viewing many videos, it enables users to gain an overview of a video, detect important events or identify dynamic features in a video without the need of viewing videos.

### 2.2. Output Data Types

While videos are the principal input to the techniques covered by this survey, the outputs can vary considerably. Typical data types of the output are:

B1. **Another video or an animation**, which is a common form of output in video-based graphics.

B2. **A large collection of images** that cannot be displayed in a single reasonably-sized image. These images may be organized as a linear sequence, or by using a hyperlinked structure.

B3. **A single composite image**, which can be as simple as an annotated keyframe, or as complex as a compositing image comprised of objects extracted from different parts of a video. It may also be a synthesized image showing a 3D model reconstructed from a video.

B4. **Additional information and actions** that accompanying any of the above three data types. One common form of additional information are textual and iconic annotations, which may be used to label objects in an output, depict relationships and connections between objects, or highlight important objects. Here the term "actions" refers to information attached to objects to facilitate interactive activities, such as hyperlinks and hotspots.

Note that we did not include a data type for text-only output. Such output is commonly seen in computer vision (e.g., "a face is detected"). The emphasis on visual forms of the output is one of the main factors that distinguish video-based graphics and video visualization from video analysis and computer vision.

For techniques that generate video output, i.e., data type (B1), we can further categorize them according to what has been added, deleted or modified:

B1.1. Presentation style (e.g., photo-realistic, pen-and-ink, water-color, etc.);

B1.2. Optical model (e.g., lighting, focus, atmospheric effects, etc.);

B1.3. Objects or object features (e.g., object replacement, etc.);

B1.4. Spatial relationship and layout (e.g., empty space removal).

For techniques in classes B2 and B3, we can further categorize them according to what is preserved from the input video:

B2.1. Timing of the events;
B2.2. Temporal ordering of events;
B2.3. Background or context;
B2.4. Tracking of focused objects.

### 2.3. Input Information

As shown in Figure 1, video analysis can provide video-based graphics and video visualization with processed information in addition to the original video. The users can also provided additional information manually. Hence we can also consider a classification based on the input information, which may include:

C1. **Original video**.
C2. **Underlying models**, such as a floor plan, a 3D environmental model.
C3. **Semantic descriptions**, such as a face to be detected.
C4. **Processed information**, such as optical flows.
C5. **User instructions**, such as editing commands and interactive direct manipulation for influencing the output.

### 2.4. Levels of Automation

One can also classify video-based graphics and video visualization techniques based on the levels of automation as:

D1. **Largely automatic**.
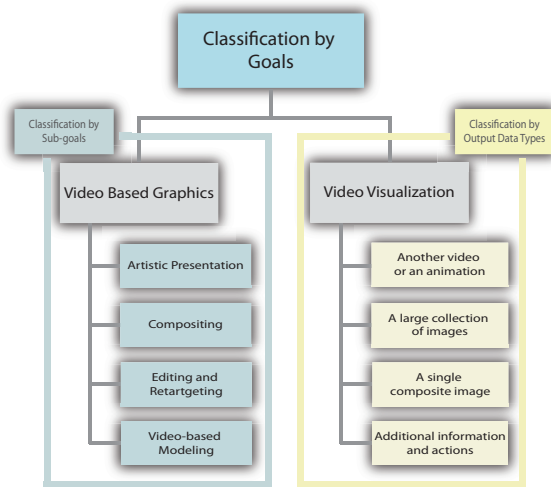D2. **Semi-automatic**.
D3. **Mostly manual**.



**Figure 4:** *Taxonomy used in the Survey*

### 2.5. Taxonomy Used in this Survey

By combining the above four classification schemes, one can define a variety of taxonomic trees. In this paper, we use



**Figure 5:** *Painterly rendering of a flower, from top-left in clockwise order: Watercolor, Van Gogh, Impressionism, Abstract, Pointillism and Flower styles [HE04]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

Scheme A for the top-level classification, separating video-based graphics and video visualization into two categories to be presented in Sections 4 and 5 respectively. For video-based graphics, we use the classification of its sub-goals, that is, categories A1.1 - A1.4, to organize Section 4. For video visualization, we use the classification of output data types, B1 - B4 to organize Section 5. Figure 4 shows a hierarchical representation of the full Taxonomy.

### 3. Video-based Graphics

### 4. Video-based Graphics

Like images, videos can provide computer graphics with spatial information of the scene (e.g., in image-based modeling and rendering), and attributes of objects (e.g., textures, BRDF data). However, videos contain a much richer set of information, such as multiple views and motion of an object. It is thereby not difficult to conclude that video data can in principle help produce more photo-realistic graphics and animation. It also provides computer artists with a richer collection of raw materials, if there are graphics tools to harvest.

### 4.1. Artistic Presentation

The success of techniques for transforming static images of the real world in artistic or technical illustrations (generally termed Non-Photo Realistic Rendering, or NPR) has inspired research into applying similar methods to image sequences or video. The major difficulty is maintaining temporal coherency of the effect throughout the video. Much effort has been made on the artistic front, with relatively little application found for technical illustration methods.

The artistic techniques are widely viewed as a "more expressive" representation of a scene, and particular focus is given to replicating art forms which would require considerable skill and time create animations with; i.e. oil based painting, which preserves brush strokes, and watercolours (which...). Such techniques are occasionally used in cinema

| paper and remark | A: goal | | C: input | | | D: Automation | | |
|---|---|---|---|---|---|---|---|---|
| | cgi | vis | video | model | edit | auto | semi | manual |
| [HE04] | x | | x | | | x | | |
| [FJS96] | x | | | | x | | x | |
| [CAMG06] | | x | x | | | | x | |
| [IA98] | | x | x | | | | x | |

**Table 1:** *Example of Classification A.*

to convey emotional effect (ref. films cited in literature). It is believed that auto-matic, flicker free (i.e. temporally coherent) methods would encourage more frequent use.

From these novel techniques, which attempt to replicate existing art-forms, have come more abstract and sometimes counter-intuitive, methods which we believe are unique to video, and may be termed "Video Art".

Early NPR techniques were applied to video by Litwinowicz [Lit97], highlighting the difficulty of temporal coherence. Minor changes in a video affected the algorithms' placement of brush-strokes, the colour quantization and other features which caused major visual distractions for the viewer.

[HP00] attempt to solve this by only "re-painting" parts of the video which have changed; thereby reliant on the underlying change detection algorithm. Optical Flow is used to direct brush strokes in the directions of movement, to highlight the sense of motion for the viewer. The authors also describe how typical video framerates of 30Hz produce an image which "can look "too real" because "the underlying motion and shape is integrated so well by the human visual system", and suggest framerates of 10-15Hz to accentuate the non-photorealistic feel.

Optical flow and mean-shift segementation are both low-level computer vision techniques which, along with morphological operators are described by Gooch et al. [GCS02] as having some value in this problem domain. Hays and Essa [HE04] extend this relationship by using edge detectors to create a wide variety of painterly styles. The frequency and gradient of the edge is used to define the brush width and stroke direction in the abstract representation. The authors show how parameters of this method can be altered to produce a wide variety of styles 5.

The use of optical flow in the above methods genenerally intends to solve two problems: segementation and direction coherence. Wang [WXSC04] employ a different method for segementing the video data, and do not consider the problem of aligning brush strokes. The authors use a mean-shift segementation of colour information in both spatial and temporal domains, which significantly reduces the effect of flickering. Collomose et al. [CRH05], extend this method to create continuous boundaries around segments idenfitied by the mean-shift operator. These segments then prevent flicker or



**Figure 6:** *Dynamic Still and Clips-Temporal context. The transparency of additional poses is based on their importance. Most informative poses (i.e., motion extreme points) are completely opaque [CAMG06]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

popping from occuring during the segementation stage of the abstraction process.

An alternative method to the previous shape or stroke based renderings involves creating a texture which is advected according to the optical flow field of the video. Bousseau et al. [BNTS07] describe this method as a means to create watercolour representations of images. In this work, the authors use the texture to describe the desposition of pigments during painting.

Real-time methods for video abstraction are uncommon due to the extensive segmentation and refinement processes. However Winnemoller et al. [WOG06]. presented a method whereby an input video is quantized in HSL colour space; the underlying representation of the video when stored in MPEG format. By quantizing only the luminance or saturation channels, similar visual results to mean-shift segementation are achieved.

## 4.2. Compositing

Time and space are intermixed components of a video, entertainment industry plays on re-expressing both components according to different canons. Compositing techniques alter the structural integrity of the contiguous video-flow to attain entertaining and aesthetically pleasing results. Space-time relationships are revisited in favour of feature-events highlighting to enrich the video experience.
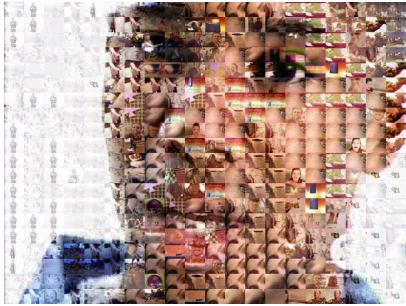
**Figure 7:** *A Frame from a Video Mosaic [KGFC02]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

Pioneer work in the field is represented by the Multiresolution Video Project [FJS96] which first introduced the use of Time Space partitioning trees to enable the organization of video sequences (normally univariate) into different temporal and spatial resolution tiers to allow for varying features and events highlighting within a unique multivariate video. Finklestein et al. enhanced the video experience by enabling the viewer to treat the video sequence as a sort of dynamic panoramic environment where the environment changes in time and carries different amounts of detail in different locations.

Finkelstein et al. paved the way to the employment of videos in a variety of applications ranging from immersive environments with the use of interactive visualisations of high-resolution time-varying video data (panoramas), to video posters with the use of both temporal and spatial multi-resolution images (mosaics).

Compositing techniques must face majour issues related to selection of informative keyframes or poses and minimization of screen space utilisation to avoid cluttering or occlusion while maximizing the conveyed visual information. To face these issues techniques like multi-resolution and clustering are borrowed from the visualisation field to achieve coherence in time and space when visualizing high detailed scenes at interactive frame rate.

### 4.2.1. Mosaicing

Mosaicing is the art of creating patterns or pictures assembling small pieces of colored glass, stones or other materials. The quality of the final outcome relies upon the semantic similarity between each mosaic tile and the respective part of the represented object. Artists have experimented with mosaic images for centuries exploiting the layered image concept and the semantic similarity function beneath the mosaic structure. Screen themselves as collection of color-varying pixels are an emblematic example of digital mosaic. With the advent of digital photography pixels and tile materials could be soon replaced by collections of small images giving birth

to what is now known as Image Mosaic. As a visual media image mosaics correspond to carefully arranged collection of small images that when seen at a proper distance (or resolution) form a recognizable larger image [FR98]. Entertainment industry has exploited the idea behind image mosaics to create large film posters composed by carefully chosen and assembled video keyframes; image tiles often undergo color adjustment to improve the quality of the final result.

Beside being an aesthetically pleasing visual media Video Posters represent a powerful resource for interactive exploration of video sequences. Solutions have been developed that rely on video posters for video browsing as in so far addressing the issues of minimizing user time while maximizing the crux of the conveyed visual information. Caspi et al. [CAMG06] proposed a method based on the tracking and extraction of salient video objects. For each tracked object *key-poses* from different time-frames are selected and eventually fused in the final image to mimic the sensation of the object motion. Key poses, also denoted as *pose slices*, are either composed into a single static image (*dynamic still*) or organized into a short video clip representing the essence of the action (*(clip trailer)*). Dynamic stills differ from standard image synopsis [IA98] as they allow self-occluding pose slices, while image mosaicing techniques usually rely on distribution and translation of object trying to avoid replication or self-intersection (as in [IA98]).

A quite different approach has been proposed by Klein et al in [KGFC02], their technique denoted as *Video Mosaics* uses video frames and not keyframes as tiling units of the mosaic composition. Video mosaics stretch Finkelstein multiresolution video concept, each video tile become a collection of layered images while the mosaic itself becomes a large video clip that can be appreciated both as a static picture or dynamic video-clip. Video tiles are not necessarily related to the master video or to each other.

As time can be stretched along different dimension so does space as in panoramic mosaicing or *panoramas*.

### 4.2.2. Panoramas

The concept of image panoramas dates back to the mid-19$^{th}$ century with majestic examples like the Warsaw panorama [BK75]. Today panoramas reconstructed from digital images are commonly used to provide virtual tours of places of interest like travel destinations and museums, or to add interactivity to simple city maps. With respect to video mosaicing panoramas maintain the temporal information explicit, time is treated as a fixed axis along which the sequence of images develops. Panoramas rely on the assumption that static portions of a scene are not dominant in the process of understanding the information conveyed through the video. This assumption allows for the creation of two distinct layers: a *dynamic layer* corresponding to the moving objects and a *static layer* corresponding to the static background. The panoramic image output is composited merging the two
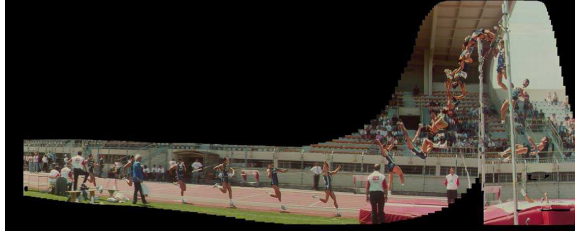
**Figure 8:** *Background and motion panoramas of a jumping athlete [BDH04]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*



**Figure 9:** *Dynamosaic of a waterfall video [RAPLP07]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

layers, static parts remain unchanged while the time-varying path of moving objects is exposed. Exemplar of bridging between the concepts of video mosaics and video panoramas is the work described in [BDH04]. Video sequences are represented as *motion panoramas* i.e. a visual representation of motion. Much effort is put in the segmentation of moving objects with respect to static background, keypose of a moving object are extracted and later stitched and aligned within a final panoramic canvas composed of the static background parts(see Figure 8).

A different approach is taken in [AZP*05] where motion is not conveyed via object tracking and silhouette extraction but maintained explicit as a video sequence. The resulting panorama becomes itself a video mosaic of video parts aligned to a single time interval and consistently stitched together, the technique is referred to as *Panoramic Video Textures* (or PVT). The PVT approach performs extremely well for objects having horizontal motion path, for more chaotic behaviours however the method introduced in [RAPLP07], and referred to as *Dynamosaic*, is better suited. *Dynamosaicing*(see Figure 9) recalls the video cube concept, first an aligned space-time volume is constructed from the input video, second a continuous 2D plane (time front) is swept through that volume generating the sequence of images. Alignment is performed via keyframe interpolation introducing a cost function to minimize artifacts due to chaotic moving objects. The natural step from dynamic panoramas to video textures is short as we can already see with the Panoramic Video Textures, this intriguing aspect of extending video to augment visual appreciation of synthetic scene is deepened in Sections 4.4 and 4.3.1.

### 4.2.3. Cut-outs

Video cut-outs are an hybrid approach between mosaics, panoramas and retargeting techniques (see Section 4.3.3). When the temporal information is lost as in still shots smooth and realistic motion can be synthesized [XWL*08, SCRS09] generating cartoon like animations [WXSC04]. Reverse engineering this process allows for the extraction of moving object from general backgrounds and for the deveeopment of sophisticated interactive systems as

in [LSS05, WBC*05b] for background substitution, object removal and reconstruction [RAKRF08].

### 4.2.4. Video Matting

Together with object cut-outs video matting is a technique of great importance for both image editing and film production applications. The technique itself concentrates on the problem of accurate foreground estimation in both images and videos. State of the art in video matting has significantly advanced recently, a good source of reference is represented by [WC07]. One of the latest achievements in interactive video editing is represented by Bai et al. [BWSS09] *SnapCut* systems which extends state of the art algorithms for both object cutouts and matting to videos.

### 4.3. Editing and Retargeting

### 4.3.1. Video Textures

Video textures [SSSE00] replace the use of static images like digital photo with synthesized video sequences enriching textured objects or scene with dynamic qualities and living action. The concept at the base of video textures is the one of Markov processes, where states correspond to video frames and probabilities to the likelihood of transitions from one frame to the other. The choice of transition points is a major challenge in creating a video texture, morphing based techniques are employed by [SSSE00] while [FNZ*09] use a similarity metric based on 3D marker trajectories and their 2D projection into the video. The use of markers is better suited for tracking of human motion as it allows for a lager control over the output animation sequence. For video texture mapping over a 3D model as in [WZY*08] extension of parameterized texture mapping techniques are a simpler choice. In [WZY*08] a mosaic of video textures is created via visual tracking, the 3D model is then parameterized over the video mosaic through an optimization function for minimizing the geometric distortion. As providing a continuous infinitely varying stream of images, easily extends to several applicative domains. Panoramic video textures can be employed in the creation of contiguous video loops, single moving objects can be extracted and employed
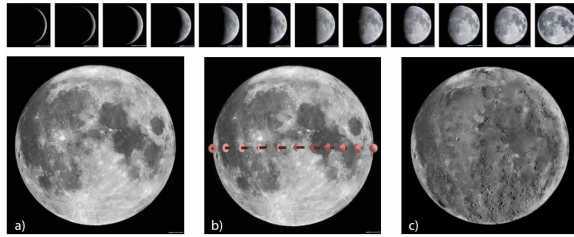
**Figure 10:** *Twelve photographs of the moon. a) Unmodified photograph. b) Controll arrows to define a field of incident light direction. c) Resulting composite photograph [ALK\*03]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

as *video sprites* [SE02] for feature based texture mapping of 3D models [WZY\*08], photorealistic animation of human motion [VBMP08, FNZ\*09], reconstruction of natural phenomena exhibiting cyclic and continuous patterns of behaviour [BSHK04, RAPLP07].

#### 4.3.2. Video Relighting

Image relighting is a general term given to describe methods which alter the lighting conditions of a scene without knowledge of the geometric or material properties of the objects which constitute the scene Typical methods require that a reflective sphere be placed in the scene to capture the light information. These spheres can then be lit under different conditions and provide the mapping from the original lighting conditions to the new conditions. Given these mappings, new objects can also be inserted into scenes and lit correctly using these methods.

Typical applications of image relighting include the entertainment industry (for film special effects), CAD, augmented reality and so on.

Video relighting is not seen as a separate problem (indeed, many methods require image sequences of varying lighting conditions), although the use of video does introduce the special problems described in previous sections (i.e. temporal coherence, frame to frame registration, etc.

Ackers et al. [ALK\*03] describe the use of image relighting techniques to constuct images which better convey they shape and texture of an object, one example being our Moon, the image of which is constructed from a time-lapse sequence of the twelve phases occuring in one month (see figure 10).

Other methods for processing on the lighting of a video have been described which may be roughly grouped under "video relighting" although distinct from image based methods. These methods attempt to process the video signal to improve the information content.

Bennett and McMillan ( [BM05]) use pixel values from



**Figure 11:** *Importance preserving image retargeting. The three important elements of the image are preserved as the image size is reduced [STR\*05]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

previous frames to increase the light level of low contrast regions. In this work, the light level of a pixel is integrated along several frames to improve perceptability. Want et al ( [Dav08]) supplement low quality digital video with an infrared video signal. As the infra-red reflectance of a surface is less affected by the incoming light direction, they use this signal to provide edge and contrast information for areas of a scene with low light levels (NB: probably needs clarification here).

Both of these methods show a trend for improving video content in low light areas. A similar trend for reducing the effects of light saturation levels in images resulted in high dynamic range photography (HDR). Some research has been conducted on HDR video [KUWS03, AA04], but at present the technology is prohibitively expensive.

#### 4.3.3. Video Re-targeting

Video retargeting attempts to resize an input video so it is more appropriate for a given display. Traditionally, this activity has been performed when films are converted from cinema (2.39:1 or 1.85:1 width to height ratio), to television (4:3 or 16:9 ratio), by manually cropping redundant elements from the scene. The wide range of digital display devices, and variety of inputs, makes manually retargeting unrealistic. As a result, automatic retargeting methods for static images and video sequences have become an active research area.

Initial video retargeting attempted to replicate the pan-and-scan methods used for converting cinema films to television. These methods used saliency maps [FXZM03] or attention models [WRL\*04] to decide how to cut the "virtual" shots introduced into the video. The aim of duplicating manual methods resulted in an introduction of new zoom and pan shots along with new cuts into the video, preserving the on-screen spatial relationship between content, but possibly affecting the narrative constructed by the director [LG06] (which is also a common critcism of the manual method.

An alternative approach fully embraced the flexibility allowed by digtal storage and rejected the need for homogenity in the retargeted scene; allowing an image to be warped, replacing large parts of the background with relatively smaller
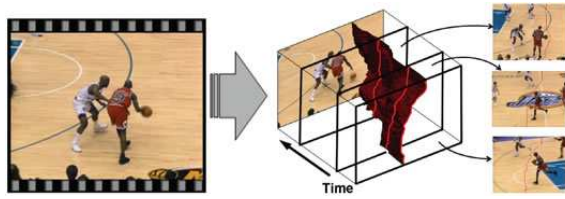
**Figure 12:** *Improved seam-carving accounts for frame changes over time, creating a more consistent viewing experience [RSA08]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*



Original　　Cropping　　Scaling　　Seam carving　　Warping　　Multi-operator
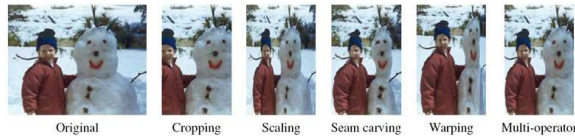
**Figure 13:** *The ouput of a number of retargeting methods, including the recent multi-operator, described by Rubinstein et al [RSA09]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

details of interesting objects [STR*05] (see figure 11 for an example). This approach was extended to video by Wolf et al. [WGCO07]. Typically in these methods, the importance of a pixel is determined by combining outputs from saliency, face detection and motion detection algorithms into a single scalar value, which allows a great deal of flexibility in the definition of "importance" as any contribution can be weighted, replaced, or even augmented with a new measure. Pipelines for these methods are described by Setlur et al. [SLNG07].

An improved representation introduced the concept of seam-carving to images [AS07], which was extended to videos via the video cube representation [RSA08]. Borrowing ideas from rotoscoping and video synthesis [KSE*03], this method preserves important regions of the video by sacrificing background content. The major contribution is the temporal coherence of the curve used to carve the video (see figure 12).

These methods have recently been combined, along with geometric image resizing methods, into a single algorithm which choses the most effective transformation method based on local properties [RSA09] to find the optimal retargeting of an input.

### 4.4. Video-based Modelling

Multiresolution videos allow for interaction with the video flat environment, video panoramas and textures are employed to enhance the perception of spatial and dynamics feature of a scene, natural step toward video apprecia-

tion is their extension to augmented reality and into different form of virtual reality experience. Environment maps, with their $360^o$ field of view, have been extensively used in crafting vr-environments and special effects, however their 2D nature allow only for single resolution display of the scene. The vast amount of optical devices that allow to capture video sequences make videos themselves to virtually be unlimited resolution means and as such a source for arbitrary resolution photorealistic imagery. In [SC96] concentrates on depth recovery in the process of reconstructing a scene from a video sequence. An image panorama of the video sequence is constructed while the depth information of the depicted scene is recovered through stereographically projecting matching keyframes pairs. Combining stereo-matching with video textures it is possible to recreate and navigate through a remote space through a virtual environment [AS99] or artwork [JPA07].

### 5. Video Visualization

Getting a quick overview over a video is an important task in many applications. No matter if analyzing surveillance videos, wanting a quick overview over a sports match or selecting a movie to watch from a large DVD collection, watching the entire movie is usually not an option. Instead, one wants a quick summary of the crucial events happening in the video. This can be done by summarizing the video by a number of short sequences like in a cinema trailer or by creating an image narrating the story. In some situations, one can also extract meaningful information, such as the motion flow, encoded in the video, and depict such information in such a way that it helps the viewer to recognize certain patterns or unusual events in the video. We refer these techniques collectively as video visualization.

In this section, we categorize video visualization techniques according to the contents and formats of the output media. We will base our classification on teh taxonomy presented in Section 2.2. In the first section we will examine methods that generate new videos as an output media, which is more "cost-effective" to view in comparison with the original videos. The following sections instead will concentrate on the common methods which summarize videos using keyframe storyboards. We consider first the techniques for keyframe selection, and then a collection of methods that enhance keyframe-based representations. This is followed by a review of techniques for generating abstract visualization, where information in the temporal feature space is displayed to aid summarization and understanding of a video.

### 5.1. Keyframe Selection

Before going into detail about the different techniques, we will have a quick look at the structure of a video. Let us consider a video of a certain length *t* that consists of several minutes or hours of film material as depicted in figure

| paper and remark | B: output | | | | C: input | | | D: Automation | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | video | collection | image | abs | video | model | edit | auto | semi | manual |
| [DC03] | | | x | x | x | | | x | | |
| [BBS*08] | | | x | x | x | | | x | | |
| [ACCO05] | | | x | | x | | | x | | |

**Table 2:** *Example of Classification B.*
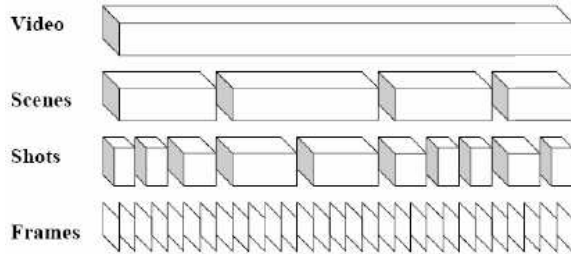


**Figure 14:** *Segments of a video. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

14. Each such video consists of a sequence of images, so called frames. Most movies consist of 24 to 30 frames per second and when watched at that rate the human eye perceives a smooth motion. Higher frame rates are for example used with high speed cameras. When one or more frames, depicting a continuous action in time and space, are combined in a contiguous recording, this is called a shot [PS97]. The assembly of subsequent shots of a semantic unit is called scene. Both, shots and scenes, can be of arbitrary length and the single units usually differ in length, i.e., there are scenes in a video that only take a split second and others that might take several minutes.

Image-based video visualization commonly operates on the three lower levels: frames, shots and sequences. For example, several frames might be selected and presented to the user or the contents of a shot or sequence might be summarized in a single image. A crucial step for all these applications is the selection of keyframes, i.e., representative frames of the video. In the following, we will first have a look at the different keyframe selection techniques, continue with different depiction methods and finish with a number of techniques that incorporate additional information into keyframes to enhance understanding.

As mentioned before, keyframe selection is commonly the first step in image-based video visualization. Keyframe selection means that we are looking for a set of images that optimally represents the contents of the video according to a specified criterion such as "find a representing image for each shot". Like in most optimization procedures, two different strategies can be pursued when choosing relevant images

from a video. Either, a maximum number of frames is given or an error rate to be met. The maximum number criterion is commonly used when dealing with limited resources. For example, when the keyframes are to be displayed on a single page or transmitted to a mobile device at a low transmission rate. The error rate is commonly applied when looking for the best set of images meeting the optimality criterion. In both techniques, manipulating the parameter effects the other one as well. Commonly, the number of keyframes and the error rate are correlated, i.e., if we allow a larger number of keyframes to be selected the error will drop and if we increase the allowed error in the second technique, we will receive more images. Hence, when choosing a strategy, we have to decide what is more important us: a fixed number of images or a limit on the error.

No matter which technique we choose, in both cases a optimality criterion has to be defined. The easiest would be to uniformly select images from the movie, but this might easily lead to missing short key sequences or several depictions of long uninteresting scenes. Truong and Venkatesh [TV07] classified a number of partly overlapping criteria for the optimization, which we summarize in the following five categories. For a comprehensive list of references please refer to [TV07].

- *Sufficient content change*: Choose keyframes such that they mutually represent different visual content. With the error criterion, we sequentially go through the video and select a frame as keyframe whenever it largely differs from the previous keyframes. Alternatively we can look for the *n* frames that represent sequences of equal variance.
- *Maximum frame coverage*: Select keyframes such that they represent a maximum number of frames that are not keyframes.
- *Feature space analysis*: Treat each frame as a point in high-dimensional feature space. One optimization strategy is based on point clustering, where the keyframes are the representative points of the clusters. Alternatively, the video can be seen as a path in high-dimensional space connecting subsequent frames and we look for a simplified path with minimal error.
- *Minimum correlation*: Choose keyframes such that they feature a minimum amount of correlation between each other.
- *"Interesting" Events*: Methods in this category take semantics into account and try to identify keyframes with

| (a) Salient stills | (b) Dynamic video synopsis | (c) Video collage |
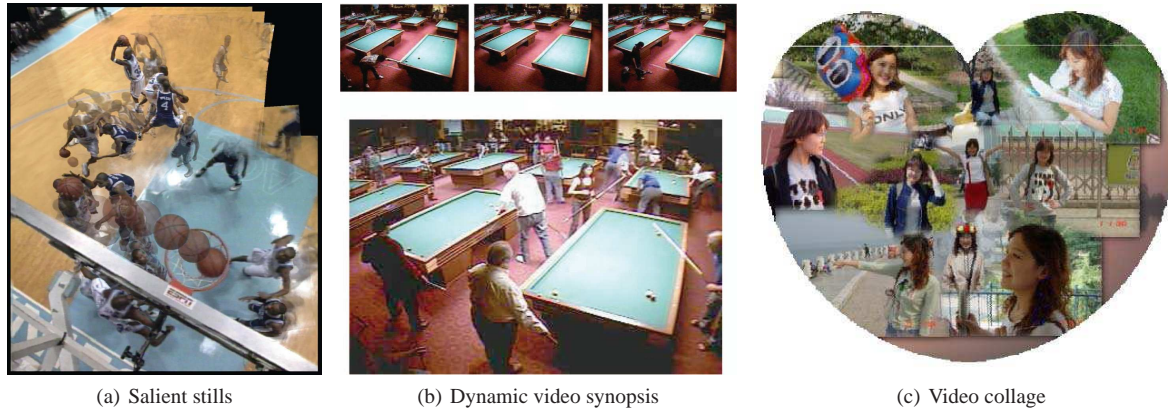
**Figure 15:** *Reassembled depictions of keyframes: (a) Salient stills compute the background from a number of frames and add local information about relevant events [TB05]. (b) A similar approach is taken by dynamic video synopsis with the focus on the concurrent display (lower image) of events at different times(upper images). (c) A video collage rearranges elements of different keyframes in a new image of arbitrary shape [MYYH08]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

high information content. They might analyze motion patterns, look for faces, or have high spatial complexity.

## 5.2. Another Video and An Animation

In this subsection, we consider a group of techniques that alleviating the problem of watching videos without leaving the video output domain. There three different approaches, differing in the way they maintain the content of the video.

The first category contains video navigation techniques. Here, the full content of the video is maintained. Content control and time compression is achieved via video browsing approaches and fast-forward techniques.

Within the second category, video montage and video synopsis, a new video with a shorter duration is created by combining different spatial and temporal video parts. Spatial and temporal context information may be lost using this technique while the occurring actions are preserved.

The third category covers video skimming techniques which skips uninteresting parts of the video to create shorter clips with the purpose of video abstraction. Due to the absence of whole video parts, time condensation is made by the cost of information loss. However, the available parts maintain spatial context information.

### 5.2.1. Video Navigation

A lot of proposals have been made regarding the problem of watching videos in a reasonable matter. Basic video browser controls include *Play, Pause, Fast-Forward, Seek, Skip-to-beginning* and *Skip-to-end of video* [LGS*00]. Li *et al.* adds additionally enhanced controls. The most important features include support to modify the playback speed between 50% and 250% of the original speed while preserving the pitch of the audio, an automatically pause removal feature that enables the user to remove parts of the video where pauses in continuous speech occur, and the possibility to select shots of the video to jump to their temporal positions [LGS*00].

Ramos and Balakrishnan focused in [RB03] to control videos with pressure-sensitive digitizer tablets. Beside fading in and out annotations and several interaction possibilities they presented a variation of the fish eye view called *Twist Lens* to seek in video streams. Therefore the time line slider consists of several sampled frames semi-occluded by each other. If the user coarsely selects a frame and increases the pressure, the slider is smoothly morphed around this frame into a sinusoidal shape (cf. Fig. 16). The occlusion of the frames in the vicinity of the selected one is decreased and an accurately selection of the time position is feasible.
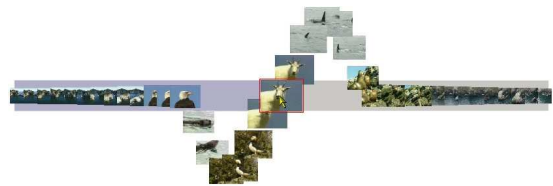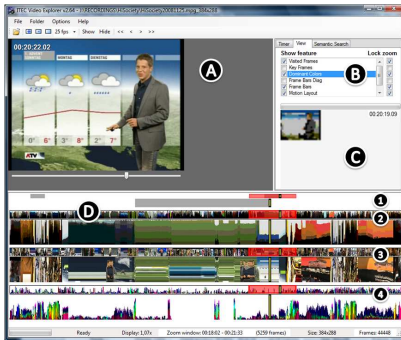


**Figure 16:** *Twist Lens [RB03]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

In [SB09] a time line slider is created as a combination of an arbitrary number of navigation summaries. This enables the user to see several content abstractions of the video in the time line at one glance. Navigation summaries can be e.g.

visited frames, dominant colors, frame stripes or a motion layout (cf. Fig. 17).



**Figure 17:** *Video Browsing Using Interactive Navigation Summaries [SB09]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

Another possibility to browse through videos is given by direct object manipulation approaches (e.g. [KDG*07, GKV*07, DRB*08, GGC*08, KWLB08]). To allow browsing video in this way objects and their movements are extracted in a pre-processing step. Afterwards, objects can be picked in the video window. The video is directly scrubbed by moving the selected object to another position (cf. Fig. 18). In [KDG*07] and [GKV*07] scrubbing is also allowed by object manipulation on a floor plan.
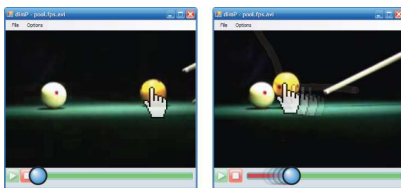


**Figure 18:** *Video Browsing by Direct Manipulation [DRB*08]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

As mentioned above, fast-forward is a basic control for video browsing. Wildemuth *et. al* evaluated in [WMY*03] how fast too fast is. They recommend to show every 64ths frame of a video for fast-forward surrogates. But even in less fast-forward speeds the user abilities in object recognition (graphical), action recognition, linguistic gist comprehension (full text), and visual gist comprehension decreases. This problem leads us to different approaches to adapt the video playback speed by video content.

Peker *et. al* adapts the playback speed relative to the motion in the videos [PDS01, PD04]. Parts of the video with less motion are played faster than parts with more motion.

An adaptive playback speed based on similarity to a target clip is proposed in [PJH05]. If a part of the video is similar to the target clip the playback speed decreases, else it increases. One example application they propose for this type of adaptive video playback is a football game. The user feed the system with a target clip of the game. Scenes of the ongoing game will then be displayed in normal speed while game interruption scenes (e.g. showing spectators) are highly accelerated.

In [CLCC09] the playback speed is adapted based on three causes: motion speed, semantic rules and user input. Motion in the video has a similar effect as in [PD04]. The manually defined semantic rules lead the playback speed to slow down while the video passes those parts. The user can manually increase or decrease the speed while the video player learns these user preferences and further adapts the speed.

### 5.2.2. Video Montage and Video Synopsis

Kang *et. al* introduced a technique for video abstraction in [KCMT06] called video montage. He extracts visual informative space-time portions from video and merges the parts together. Their technique changes the temporal and the spatial occurrence of the information and results in a shorter video clip with condensed information (cf. Fig. 19).
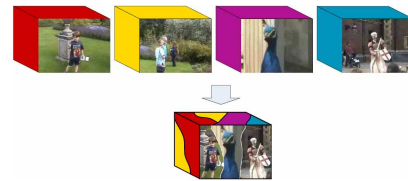


**Figure 19:** *Space-Time Video Montage [KCMT06]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

One of the method's drawbacks is the loss of spatial context. A method preserving spatial positions was proposed in [RAPP06], [PRAGP07], and [PRAP08a]. For their approaches objects are detected, tracked, and temporal rearranged. The recomposed video shows different actions, occurring at different temporal positions, at the same time. Even if the trajectory of the object has a long time duration it is cut into several pieces all displayed at the same time.

### 5.2.3. Video Skimming

The goal of video skimming is to create a short summarization of a given video stream. Therefore, less interesting parts of the video are discarded.

Truong *et. al* identified a five-step process for automatically video skim generation [TV07]. For some video skimming techniques steps are skipped or combined in a different

variation, but the basics remain. These five steps are *segmentation* (extract shots, scenes, events, parts of continuous speech, etc.), *selection* (choose 'interesting' parts for summarization), *shortening* (reduce the time duration for the selected parts further, e.g. by cutting), *multimodal integration* (combine skims for different features like image, audio, and text into the final skim), and *assembly* (temporal arrange independent video skim parts, e.g. chronological).

Due to the fact that the field of video skimming is a huge research area and a lot of publications have been made there, we refer to [TV07] for further reading.

### 5.3. A Large Collection of Images

The easiest direct depiction of keyframes is the storyboard technique, where equally sized images are arrange on a regular grid, e.g., three by four images on a page [BT07]. This technique can be extended to allow for different levels of temporal detail when presenting the keyframes in a hierarchical manner [Parshin and Chen 2000; Sull et al. 2001]. At the top level a single frame represents the entire film and at the lowest level, all frames are included. Although easy to apply and understand, both techniques have the disadvantage, that the do not provide information about the relevance of individual snapshots. To include such semantics, the images can be scaled according to their importance for the video [YY97, UFGB99]. Yeung and Yeo [YY97], for example, use the number of frames being represented by a keyframe, which is equivalent to the subset's length, to scale the keyframes of a sequence and arrange them according to predefined design patterns in a video poster. The illustration of several video posters in temporal order summarizes the content of a sequence. Barnes et al. [BGSF10] presents a novel approach to video summarization called *Tapestries* merging the structure of DVD chapter menus with the timeline representation of video editing tools.

### 5.4. A Single Composite Image

All methods belonging to the previous category have in common that they do not alter the contents of the individual keyframes. Reassembled depictions, on the contrary, combine the contents of several images to create a new one. An early goal in this area was to reconstruct the background of a scene. Methods to achieve such a reconstruction [IAH95, TAT97, LCL*97, JDD99], sometimes called mosaics, combine several successive video frames and reconstruct the scene while correcting for camera movement and zooming. Salient stills [TB05] extend this technique and add additional information about temporal changes (figure 15(a)). Therefore, salient regions of interest are extracted and seamlessly arranged on the background such that the temporal structure of the video content is preserved. A similar approach is followed by Pritch et al. [PRAP08b] who concentrate on the simultaneous depiction of events happening at different times in the video (figure 15(b)).



(a) Schematic storyboards
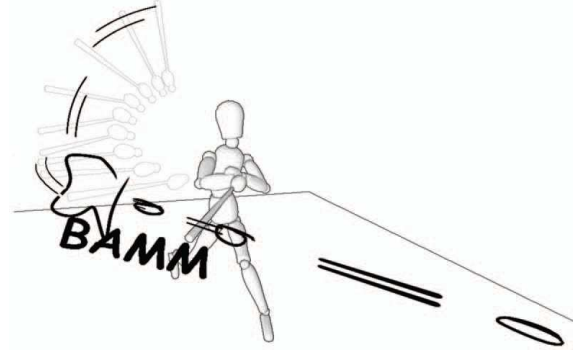


(b) Dynamics glyphs

**Figure 20:** *Enhanced stills: (a) Schematic storyboards enhance the displayed keyframes with additional information on characters and camera movement [GCSS06]. (b) In [ND05] additional dynamics glyphs are used to enhance understanding. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

An alternative approach is taken by techniques that extract relevant subsections of the keyframes and reassemble the sub-images in a new image. The video collage technique [CGL04] first arranges the important components on a page and fills the gaps in between with image data according to the Voronoi tessellation of the data. This approach was extended in the video collage algorithm [MYYH08] and auto-Collage [RBHB06] where a combination of template-based arrangement and an energy minimization algorithm is used to find good locations for the different sub-images. While the first causes on boundaries of arbitrary shape (figure 15(c)), the second concentrates on seamless transitions between the different sub-images.

### 5.5. Additional Information and Actions

### 5.5.1. Enhanced Stills

In our last category of keyframe depictions techniques, we will summarize methods that add additional information to the extracted keyframes. A well known approach are schematic storyboards (figure 20(a)), where annotations are added to illustrate the movement of persons or the camera [GCSS06]. Nienhaus and Dollner [ND05] take a similar

approach using additional dynamics glyphs. Further image-based video visualization that enhance the raw data are graph-based approaches that additional to the keyframes depict the interaction between different characters or the use of different scenes in a graph [ACCO05].

### 5.5.2. Video Abstraction

In some cases, abstract attributes, such as changes in a scene, changes between frames, motion flow, and pixel clusters, can be depicted visually to aid the understanding of a video using only one or a few visualizations. Such visualization may not display objects in an intuitive manner, but the abstract visual representation can convey temporal attributes more effectively than discrete keyframe displays.

A popular approach interprets video data as a space-time volume. This idea was first published by Fels and Mase in [FM99]. Here, the spatial axes x and y are combined with time as the third axis (cf. Fig. 21). Within this representation, they define *cut planes* to intersect the video volume. Cut planes can be defined arbitrary to watch the video in a different way. Normally watching video in this context is nothing else but using a cut plane parallel to the x-y axes that is moving along the z axis. The principle of cut planes through a video volume were refined for other applications like cut outs (e.g. [WBC*05a]) or non-photorealistic rendering (e.g. [KSFC02]).
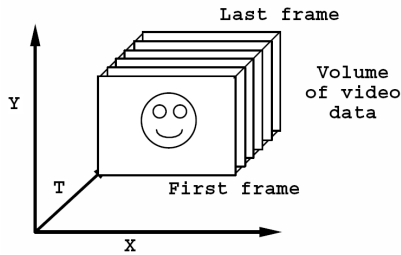


**Figure 21:** *Interactive Video Cubism [FM99]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

Daniel and Chen proposed to employ volume visualization techniques to visualize the video volume with the aim of summarization [DC03]. They transformed the video volume into other shapes, e.g. a horseshoe view, to convey more information at the same time. A change detection filter was applied and the results were displayed in the volume. Within this visualization, several visual patterns can be identified indicating related events like changes that remain for a period (cf. Fig. 22), walking with moving arms or an opened door.

In [CBH*06] Chen *et. al* introduced *visual signatures* as abstract visual features to depict individual objects and motion events. Therefore he applies and evaluates flow visualization techniques to video volume visualization. Example
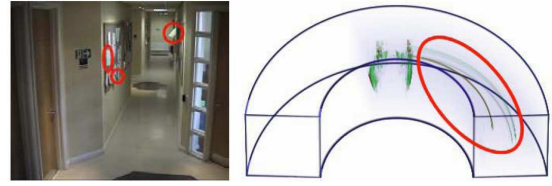


**Figure 22:** *Video Visualization - changes that remain for a period [DC03]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

visual signatures they used to evaluate their approach are a temporal visual hull, a colour coded difference volume, glyphes, and streamlines. An example of the four visual signatures can be seen in Fig. 23. In this case, a sphere moves towards the upright corner of the image frame.
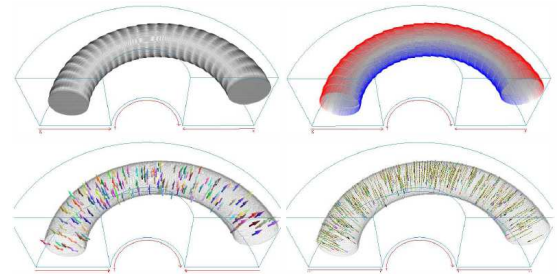


**Figure 23:** *Visual Signatures in Video Visualization [CBH*06]. (Permission to publish any of the pictures presented in this STAR will be obtained prior to publication.)*

A further enhancement was done by Botchen *et. al* in [BBS*08]. In this work, the video volume visualization approach has been further enhanced with semantic annotations. This approach is discussed in section.

### 6. Video Analysis

In this section we present state of the art methods used in the field of computer vision to extract information from image sequences. Primarily we review those techniques that we believe to be of principal interest to the graphics and visualization community. These methods are broadly split into two subgroups, *low-level* and *high-level vision*. Low-level vision techniques often operate at the pixel level of an image and are generally employed to reduce the dimensionality/complexity of an image so that it can be processed by higher-level, often more complex, algorithms. Low-level vision can be interpreted as a filtering step used to remove redundant information that is often of little or no interest.

The typical output of these algorithms may be a set of interest features, optical flow vectors or an image segmentation. However, this information alone often provides little useful insight as regards to the contents of an image sequence, such as a semantic label or description.

Alternatively, high-level algorithms that almost exclusively operate on the output of low-level vision approaches, can be used to automatically extract some high-level information from a video sequence. Such as a list of events that have taken place, a set of locations where objects have been detected or alternatively a 3D reconstruction of the scene depicted in the sequence. It is this high-level extraction of data that is the primary goal of most computer vision practitioners. However one of the principal difficulties encountered is in overcoming errors produced by low-level algorithms. As a result currently approximately equal effort is spent by the vision community in improving low-level methods as are invested in developing high-level approaches.

### 6.1. Low-Level

In this section we describe *low-level* vision techniques that are particularly relevant to the domain of video analysis, we group these into three principal areas: *optical flow estimation*, *image segmentation* and *feature extraction*. Whilst optical flow estimation and image segmentation provide a well defined output that can be treated as a complete tool in the visualization or graphics pipeline, feature extraction will often produce a more abstract output that is only of benefit to the higher-level algorithms that are designed to exploit it.

#### 6.1.1. Optical Flow Estimation

Motion estimation is one of the most fundamental techniques relevant to video analysis since it exploits the key element that distinguishes video from single images, the temporal dimension. Whilst the focus of this section will be on commonly used differential methods, block matching can also be used to extract motion information and should briefly be mentioned. In its simplest formulation block matching takes each image patch and exhaustively compares it against it's neighboring frames to find the best matching location. This approach is typically used for video compression and is therefore not concerned about the correctness of the estimated motion, only that matched blocks closely resemble one another. Various methods have been proposed to perform block matching more efficiently such as the diamond search adopted for the reference implementation of MPEG-4 [ZM97]. A comprehensive survey of block matching techniques is given by Huang *et. al.* [HCT*06].

The most popular methods for motion estimation between two consecutive frames are differential methods. These approximate optical flow using a first order Taylor expansion of image motion and as such assume only small displacements between consecutive frames, though are capable of

achieving sub-pixel accuracy. Differential methods to estimate optical flow can be split into *local* and *global* methods, whilst local methods attempt to solve the motion for small regions of the image independently global methods attempt to solve motion for the entire image in one instance.

Perhaps the most popular local method is that of Lucas and Kanade [LK81], this is an iterative approach that uses Newton-Raphson gradient descent to minimize the dissimilarity between patches in consecutive images. The shortcoming of this approach is that it fails to address the aperture problem, where often an ambiguity exists such that motion can not be uniquely identified. This results in some regions for which the motion is unknown.

Global methods solve the same first order Taylor expansion of image motion, however, introduce a regularization term or smoothness penalty. The addition of the smoothness penalty allows the optical flow to be estimated in regions where local methods would fail as a result of the Aperture problem. This allows dense flow to be estimated, however, this method is particularly sensitive to image noise [BFB94, BWS05]. The most notable global method is that of Horn and Schunk [HS81].

Whilst the local method of Lucas and Kanade fails to solve the aperture problem their formulation provides a method to test how well a particular image patch could be tracked, this is achieved by examining the eigenvalues of the covariance of the image gradients [ST94]. Two large eigenvalues implies large gradients (i.e. edges) in adjacent directions of the patch (think "X") which represent a good feature to track. Using this method each motion vector can have a level of certainty attached to it about how reliable the feature used can be tracked. This is often invaluable for higher level algorithms since noisy data can automatically be discarded. Some methods have been suggested to 'densify' the sparse output of the Lucas-Kanade method using interpolation [HCG05], which provides better dense motion estimation than global methods in sequences where there is little texture. Another approach is that of Bruhn *et. al.* who investigate combining local and global methods to extract optical flow, this is achieved by using local confidence measures and effectively growing a dense representation [BWS05].

Other local methods use local spectral phase differences to estimate motion displacements between images [FJ90] and a real-time approach using the census transform to represent a pixel neighborhood is proposed by Stein [Ste04]. An evaluation of optical flow methods can be found in [BFB94] and [GMN*98]. For a comprehensive survey on global optical flow methods we refer to [WBBP06].

#### 6.1.2. Image Segmentation

Image segmentation is a generic term for grouping pixels in an image or video into a number of predefined classes, such as those that belong to a particular object or those that are

part of the foreground. Pixels are classified using image cues such as color/texture [SS04] and often the spatial location of the pixels are exploited preferring neighboring pixels to be members of the same class. These include methods such as split and merge, region growing and edge based techniques (comprehensive surveys can be found in [CJSW01, LM01]). These approaches often result in a segmented image being represented as a set of blobs, each blob representing a different homogeneous region, though each blob may not necessarily have a semantic meaning.

In general image segmentation is not a well defined problem in that a good segmentation is itself somewhat subjective and dependent on what the user requires, for this reason often methods must be trained for the task for which they are required (e.g. skin detection [KMB07]). Perhaps one of the most popular uses of segmentation in video is background subtraction [Pic04, McI00] or more generally change detection [RAAKR05], where the segmentation algorithm is trained on the particular scene it is observing so that it is able to detect (segment) any pixels or regions that change temporally.

Further methods for image segmentation include dynamic programming [Fel05, CYES00], graph cuts [BFL06] and level sets [CRD07]. These approaches allow segmentation to be formulated as an energy minimization problem and have the advantage that they allows the inclusion of complex shape priors specific to the task for which they are required, for example segmenting cows [KTZ05], leaves [Fel05] or hands [CYES00]. These methods are particularly robust to noise and background clutter and it is the inclusion of the aforementioned shape priors that provides this robustness.

The drawback to these energy minimization approaches is that they can not be used 'out of the box' and must be trained to the specific task for which they are required which can be somewhat involved. Whilst methods that segment based only on extracting homogeneous regions can be treated largely as a black box and for images that do not contain too much clutter can achieve acceptable results.

### 6.1.3. Feature Extraction

In this section we describe low-level features commonly used in computer vision algorithms, these can be subdivided into two principal categories: global and local features. Global features describe a property of the entire image, such as statistics about the luminance or color, as opposed to local features that describe the properties of only a small region.

The key advantage of local over global features is that information extracted can be attributed to a particular location in the image, this is crucial if for example an object is being tracked or detected within an image. Though surprisingly, if applied to a tightly constrained problem global features can yield encouraging results. For example detecting wildlife frames containing quadrupeds using just the 2D power spectrum of an image [Sio07].

Some global features may be learnt adaptively for a specific video clip, for example statistical techniques such as principal component analysis (PCA) can be used to project entire frames into a two or three dimensional space allowing a complete video to be easily visualized. Furthermore, clustering this low dimensional representation permits automatic key frame extraction [GCT02].

However, the failure of global features to provide information about specific regions of an image mean they are of limited use in video analysis; their strength lies in applications where the interest is in looking at large scale properties of an image sequence, for example to detect shot boundaries, or for classification problems where the domain is very constrained.

Low-level features can either be generated exhaustively at every point in the image, in which case a higher-level learning algorithm can be used to select the set of features that are most relevant to a particular problem, or interest point detectors can be used to automatically detect image regions of interest. Different interest point detectors regard interesting features in different ways, for example the Kande-Lucas-Tomasi feature tracker [ST94] discussed in Section 6.1.1 defines an interest feature as an image patch with a covariance with two large eigenvalues. Other standard interest feature detectors include Harris corner detector [HS88], Förstner-Gülch [FG87] and the Reisfeld symmetry operator [RWY90].

Within the last decade invariant local features have became very popular including approaches like *SIFT* [Low04] or *SURF* [BTG06] that rely for scale adaption on the scale-space theory introduced by Lindeberg [Lin98]. Other techniques e.g. *MSER*s [MCUP04] intrinsically adapt the detected region size. A variety of affine interest point detectors as well as suitable region descriptors are evaluated by Mikolajczyk *et al.* in [MTS*05] and [MS05] respectively. A recent evaluation of the matching performance of several detector-descriptors combinations for 3D object features is provided by [MP07].

Low-level features used by machine learning techniques to train classifiers/detectors include simple rectangular features which are fast to compute and can capture large scale structure as well as some information on image texture [VJ01], Histogram of orientated gradients (HOG) features [DT05] which primarily capture information about image gradients and local binary Patterns (LBP) [AHP04] which capture texture. These features are designed to be fast to compute and offer some robustness to noise or small changes in for example, the illumination or orientation of the object. These features are often very much simpler than their interest point detector counterparts, though as a result less discriminative.

Thus far all features presented are only spatial in nature, however, often these features can be extended to the temporal domain. These include a temporal extension of the SIFT

feature [SAS07], temporal Gabor filters [DRCB05], temporal Harris corner features [Lap05] and temporal simple rectangular features [KSH05]. Typical uses for these types of features are for video retrieval or action recognition. A discussion on spatio-temporal interest-points and an evaluation of volume descriptors is presented in the work of Laptev and Lindeberg: [Lap05] and [LL06].

### 6.2. High-Level

In this section we review high-level methods used to extract information from video sequences. These are split into three categories *recognition and detection*, *tracking* and *3D reconstruction*.

### 6.2.1. Recognition and Detection

Recognition and Detection can both be seen as a classification problem. However, the difference between them is that a detection problem can be seen as a two choice classification problem and recognition a 'one of N' classification problem. Perhaps counterintuitively this does not imply that detection is an easier problem. For example take a pedestrian detector, whilst the positive class is well defined the negative (no pedestrian) class must represent every possible image that does not contain a pedestrian, of course this image class is infinite and this can not be achieved. A recognition task however, is very often more constrained e.g. Given a text character what letter is it most likely?

A recognition or detection system is composed of two parts, a set of low-level features such as those discussed in Section 6.1.3 and a classifier which will be trained using examples of each class. Popular classifiers includedecision trees, neural networks, *AdaBoost*, support vector machines (SVM) and k-nearest neighbours (KNN). There exist several well documented implementations of all of these classifiers and a good introductory text to machine learning is provided by Bishop [Bis06]. All of the above methods are trained using a set of positive and negative labeled examples and cross-validation may be used to prevent over fitting to the training data.

The typical approach to object detection is using a sliding window to exhaustively test whether an object is located at each pixel location in the image at varying scales. For example this method has been used for face detection using AdaBoost combined with rectangular features [VJ01] and pedestrian detection using a SVM combined with HOG features [DT05]. For detection of objects that exhibit a lot of variation in appearance due to changes in orientation or articulation a part based method may achieve improved results (e.g. [FMR08]). Another approach that shows promise is to model context to improve object detection (for a recent review see [SKD09]).

For classifying sequential data hidden Markov models (HMM) commonly used in speech recognition remain a popular choice for example to classify the trajectories of the hands performing different gestures [WB99] or martial art actions [SCM02]. However, recently combining temporal features and using classifiers such as those discussed in the previous paragraphs have became popular [KSH05, Lap05, DRCB05]. For example temporal corners are used to detect sudden changes in motion present in actions such as walking or bouncing a ball [Lap05]. Subtle actions such as grooming, eating and sleeping performed by rodents have been recognized using Gabor filters applied to the temporal dimension of an image sequence [DRCB05].

### 6.2.2. Tracking

Surprisingly tracking and detection are closely related. If detection was 100% accurate tracking would be redundant, an object could simply be located in an image in each frame independently. However, currently this is not the case and tracking exploits knowledge of an object's location in a previous time instance to make a prediction and thus narrow the search space of the object's location at the present time. Most tracking algorithms assume detection or initialization in the first frame to be a separate problem and the integration of the tracking and detection into a common framework remains an open problem in computer vision though some recent attempts have been made (e.g. [ARS08]).

There are a small handful of established tracking algorithms, most notably the Kalman filter (an excellent tutorial is provided in [WB95]) which assumes Gaussian noise and a linear dynamic model and the particle filter (See [AMGC02] for a good tutorial) which is a stochastic approach and as such makes no assumption about the underlying probability distributions or dynamics. Each has a number of variations, the most popular is the extended Kalman filter (again see [WB95]) which is an extension of the Kalman filter to incorporate non-linear dynamics and the annealed particle filter [DBR00] which uses the method of simulated annealing to allow the stochastic search of the Particle Filter to be performed in a structured, layered approach.

Most recent developments made in the field of tracking have been domain specific, in particular modeling the solution space or system dynamics of a particular problem. As examples in the case of 3D human pose estimation methods such as Gaussian process models [UFF06] or principal component analysis (PCA) [ZL08] have been used to learn action specific models (e.g. walking) so tracking can take place in a much lower dimensional space. For the domain of tracking individuals in crowded environments models of social interaction have been learnt to predict how people will behave which can be used to improve the performance of tracking algorithms [PESvG09].

Tracking can also be made more robust by learning the appearance of the object online. For example learning the appearance of individual limbs whilst tracking articulated

objects [RFZ07] or adapting an offline trained classifier to a specific instance of an object observed during run time [GRG].

### 6.2.3. 3D Reconstruction

Well established techniques to extract a 3D reconstruction from video or multiple images include Structure from Motion $SfM$ [DSTT00], where point correspondences between frames can be used to recreate a sparse 3D reconstruction and space carving approaches where a voxel representation is typically used to create a dense 3D reconstruction. Whilst we briefly discuss these well established techniques we also discuss more recent methods that typically attempt to reconstruct 3D structure from single images that could also be applicable to video.

The main benefit of the SfM approach is that it is relatively inexpensive both in terms of computation and memory, to date entire cities have been reconstructed [ASS*09]. Furthermore a dense reconstruction can be estimated through the use of *a priori* knowledge, such as assuming all surfaces are planar [FCSS09]. The principal assumption in most SfM algorithms is that the scene is rigid and any motion observed is due to either camera motion or from the entire scene moving as a rigid entity. Non-rigid motion of, for example, the face or simple deformable objects such as a shoe have been accommodated in the SfM framework by extracting a set of rigid 3D basis shapes allowing the object in each frame to be constructed from a linear combination of these basis shapes [TYAB01, TB02]. The algorithms used in SfM are relatively mature and well understood and a number of commercially available software packages exist. As the process of 3D reconstruction becomes automated it is desirable to be able to exclude objects that are not wanted in a final 3D reconstruction, for example in a reconstruction of a city cars and pedestrians could be automatically detected and removed [CLCVG08].

In contrast to the SfM approach, space carving approaches require a predefined 'search space' to be constructed in which the object or scene of interest is assumed to be contained within. This space is split into voxels, each voxel is projected into every frame and a measure of consistency is extracted, if a voxel is consistent across all views it is assumed to be on the surface of the object of interest otherwise it is discarded [KS00]. In this approach it is typically assumed the cameras are fully calibrated and the surface of the object is Lambertian. The space carving framework shares many similarities with image segmentation, though extended to 3D, and the inclusion of a regularization term to enforce surface smoothness allows it to also be formulated as energy minimization. As such a variety of standard approaches have been used to perform the segmentation such as level-sets [FK] and graph cuts [VTC05].

Whilst voxel based methods provide a dense reconstruction its principal limitation is the expense in both computation and memory use compared to the sparse SfM methodology. Alternative approaches such as creating a 3D reconstruction using a sparse set of planar patches have shown some promise in overcoming the shortcomings of each [ZPQS07].

It is perhaps worth mentioning approaches to extract 3D structure from single images that could be applied to video sequences. Whilst cues such as shading [DFS08] or texture [LF06] can be used to extract some information about 3D structure independently most approaches tend to achieve accurate results by making assumptions about the scene or object being viewed. For example in estimating the 3D shape of a human face a 3D geometric prior model may first be learnt to constrain the solution space [RV05]. Machine learning approaches are also popular for example to learn a regression from 2D binary silhouettes to 3D human poses [AT06]. To allow reconstruction of more unconstrained images a classifier may be learnt to identify different image elements such as sky, ground or buildings which allow simple pop-up 3D models to reconstructed [HEH05]. For reconstruction of structured objects such as buildings a grammar can be learnt that describes how different architectural features should relate to one another [KST*09].

In the majority of cases current monocular approaches tend to achieve quantitatively poor results compared to those using SfM or space carving, however, for many tasks qualitatively the results are acceptable. Furthermore for sequences where very little texture exists making assumptions about the environment being viewed may be the only method to resolve many of the ambiguities that exist. It is likely the area of 3D reconstruction coupled with machine learning techniques will continue to receive much attention over the coming years.

### 7. Summary and Conclusions

We have examined the state-of-the-art of video-based graphics and video visualization, and proposed a new taxonomy to categorize the concepts and methods in this newly emerged field. We have the following observations:

- The developments in video-based graphics have been following a certain trend in parallel with that in the digital entertainment industry. It is driven primarily by the demand for novel and creative digital contents as well as the need for consumer multimedia applications. This trend is expected to continue, and hence provide new inspiration and stimulus for further research and development. However, the focus will likely to change from one sub-goal to another, while new sub-goals will emerge.
- Video visualization can have applications in many disciplines including science, engineering, sports, medicine and security. However, most these applications share a common goal, that is, to reduce the time needed for watching videos and to assist the users to gain insight and make

decisions in a more cost-effective manner. Different output data types reflected diversity of the means to achieve such a common goal. With the rapid increase of captured video data, there will be a continuous increase of demand for video visualization to address the shortcoming of automated video analysis. The research in this area also faces a huge challenge of scalability in terms of space, time, and interaction required for viewing visualization.

• Like artificial intelligence, automated video analysis is an ultimate ambition in computer science. Although the realization of such an ambition will require a long term effort, the research and development in video analysis has resulted in a large collection of low-level and high-level techniques. Many techniques, such as optical flow estimation and 3D model reconstruction, have already been adopted for pre-processing data in video-based graphics and video visualization. Many more are yet to be integrated into systems for video-based graphics and video visualization. Hopefully, the brief overview of video analysis in Section 5 will enthuse researchers to explore various techniques originally developed for automated video analysis.

In addition, there is an emerging interest in handling stereo video streams, which is not surveyed in this report. The process of making movies such as "Avatar" in stereo is raising many research challenges on how to manipulate stereo footage in the process. We believe that video-based graphics and video visualization will continue to be fruitful areas of research.

## References

[AA04]  AGGARWAL M., AHUJA N.: Split aperture imaging for high dynamic range. *Int. J. Comput. Vision 58*, 1 (2004), 7–17. 8

[ACCO05]  ASSA J., CASPI Y., COHEN-OR D.: Action synopsis: pose selection and illustration. *ACM Trans. Graph. 24*, 3 (2005), 667–676. 14

[AHP04]  AHONEN T., HADID A., PIETIKAINEN M.: Face recognition with local binary patterns. In *European Conference on Computer Vision* (2004), pp. 469–481. 16

[ALK*03]  AKERS D., LOSASSO F., KLINGNER J., AGRAWALA M., RICK J., HANRAHAN P.: Conveying shape and features with image-based relighting. In *VIS '03: Proceedings of the 14th IEEE Visualization 2003 (VIS'03)* (Washington, DC, USA, 2003), IEEE Computer Society, p. 46. 8

[AMGC02]  ARULAMPALAM S., MASKELL S., GORDON N., CLAPP T.: A tutorial on particle filters for on-line nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing 50*, 2 (2002), 174–188. 17

[ARS08]  ANDRILUKA M., ROTH S., SCHIELE B.: People-tracking-by-detection and people-detection-by-tracking. pp. 1 –8. 17

[AS99]  AKESSON K.-P., SIMSARIAN K.: Reality portals. In *VRST '99: Proceedings of the ACM symposium on Virtual reality software and technology* (New York, NY, USA, 1999), ACM, pp. 11–18. 9

[AS07]  AVIDAN S., SHAMIR A.: Seam carving for content-aware image resizing. In *SIGGRAPH '07: ACM SIGGRAPH 2007 papers* (New York, NY, USA, 2007), ACM, p. 10. 9

[ASS*09]  AGARWAL S., SNAVELY N., SIMON I., SEITZ S., SZELISKI R.: Building rome in a day. pp. 72 –79. 18

[AT06]  AGARWAL A., TRIGGS B.: Recovering 3d human pose from monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence 28*, 1 (2006), 44–58. 18

[AZP*05]  AGARWALA A., ZHENG K. C., PAL C., AGRAWALA M., COHEN M., CURLESS B., SALESIN D., SZELISKI R.: Panoramic video textures. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers* (New York, NY, USA, 2005), ACM, pp. 821–827. 7

[BBS*08]  BOTCHEN R. P., BACHTHALER S., SCHICK F., CHEN M., MORI G., WEISKOPF D., ERTL T.: Action-based multifield video visualization. *IEEE Transactions on Visualization and Computer Graphics 14*, 4 (2008), 885–899. 14

[BDH04]  BARTOLI A., DALAL N., HORAUD R.: Motion panoramas: Research articles. *Comput. Animat. Virtual Worlds 15*, 5 (2004), 501–517. 7

[BFB94]  BARRON J., FLEET D., BEAUCHEMIN S.: Performance of optical flow techniques. *International journal of computer vision 12*, 1 (1994), 43–77. 15

[BFL06]  BOYKOV Y., FUNKA-LEA G.: Graph cuts and efficient nd image segmentation. *International Journal of Computer Vision 70*, 2 (2006), 109–131. 16

[BGSF10]  BARNES C., GOLDMAN D. B., SHECHTMAN E., FINKELSTEIN A.: Video tapestries with continuous temporal zoom. *ACM Trans. Graph. 29*, 4 (2010), 1–9. 13

[Bis06]  BISHOP C. M.: *Pattern Recognition and Machine Learning (Information Science and Statistics)*. 2006. 17

[BK75]  BRANDEL C., KOZARSKI A.: Warsaw panorama, poland 1873, 1875. 6

[BM05]  BENNETT E. P., MCMILLAN L.: Video enhancement using per-pixel virtual exposures. *ACM Trans. Graph. 24*, 3 (2005), 845–852. 8

[BNTS07]  BOUSSEAU A., NEYRET F., THOLLOT J., SALESIN D.: Video watercolorization using bidirectional texture advection. In *SIGGRAPH '07: ACM SIGGRAPH 2007 papers* (New York, NY, USA, 2007), ACM, p. 104. 5

[BSHK04]  BHAT K. S., SEITZ S. M., HODGINS J. K., KHOSLA P. K.: Flow-based video synthesis and editing. *ACM Trans. Graph. 23*, 3 (2004), 360–363. 8

[BT07]  BAILER W., THALLINGER G.: A framework for multimedia content abstraction and its application to rushes exploration. In *CIVR '07: Proceedings of the 6th ACM international conference on Image and video retrieval* (New York, NY, USA, 2007), ACM, pp. 146–153. 13

[BTG06]  BAY H., TUYTELAARS T., GOOL L. V.: *SURF: Speeded Up Robust Features*, computer science ed., vol. Volume 3951/2006 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 7 2006, ch. Computer Vision - ECCV 2006, pp. 404–417. 16

[BWS05]  BRUHN A., WEICKERT J., SCHNÖRR C.: Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *International Journal of Computer Vision 61*, 3 (2005), 211–231. 15

[BWSS09]  BAI X., WANG J., SIMONS D., SAPIRO G.: Video snapcut: robust video object cutout using localized classifiers. *ACM Trans. Graph. 28*, 3 (2009), 1–11. 7

[CAMG06]  CASPI Y., AXELROD A., MATSUSHITA Y., GAMLIEL A.: Dynamic stills and clip trailers. *Vis. Comput. 22*, 9 (2006), 642–652. 5, 6

[CBH*06] CHEN M., BOTCHEN R., HASHIM R., WEISKOPF D., ERTL T., THORNTON I.: Visual signatures in video visualization. *IEEE Transactions on Visualization and Computer Graphics 12*, 5 (2006), 1093. 14

[CGL04] CHIU P., GIRGENSOHN A., LIU Q.: Stained-glass visualization for highly condensed video summaries. In *Proc. IEEE Intl. Conf. on Multimedia and Expo* (2004), pp. 2059–2062. 13

[CJSW01] CHENG H., JIANG X., SUN Y., WANG J.: Color image segmentation: advances and prospects. *Pattern Recognition 34*, 12 (2001), 2259–2281. 16

[CLCC09] CHENG K., LUO S., CHEN B., CHU H.: Smartplayer: user-centric video fast-forwarding. In *Proceedings of the 27th international conference on Human factors in computing systems* (2009), ACM New York, NY, USA, pp. 789–798. 12

[CLCVG08] CORNELIS N., LEIBE B., CORNELIS K., VAN GOOL L.: 3d urban scene modeling integrating recognition andǎreconstruction. *International Journal of Computer Vision 78* (2008), 121–141. 18

[CRD07] CREMERS D., ROUSSON M., DERICHE R.: A review of statistical approaches to level set segmentation: integrating color, texture, motion and shape. *International Journal of Computer Vision 72*, 2 (2007), 195–215. 16

[CRH05] COLLOMOSSE J., ROWNTREE D., HALL P.: Stroke surfaces: temporally coherent artistic animations from video. *Visualization and Computer Graphics, IEEE Transactions on 11*, 5 (Sept.-Oct. 2005), 540–549. 5

[CYES00] COUGHLAN J., YUILLE A., ENGLISH C., SNOW D.: Efficient deformable template detection and localization without user initialization. *Computer Vision and Image Understanding 78*, 3 (2000), 303–319. 16

[Dav08] DAVE C.: Video relighting using infrared illumination. *Computer Graphics Forum 27* (April 2008), 271–279(9). 8

[DBR00] DEUTSCHER J., BLAKE A., REID I.: Articulated body motion capture by annealed particle filtering. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2000), pp. 126–133. 17

[DC03] DANIEL G., CHEN M.: Video visualization. In *Visualization, 2003. VIS 2003. IEEE* (2003), pp. 409–416. 14

[DFS08] DUROU J.-D., FALCONE M., SAGONA M.: Numerical methods for shape-from-shading: A new survey with benchmarks. *Computer Vision and Image Understanding 109*, 1 (2008), 22–43. 18

[DRB*08] DRAGICEVIC P., RAMOS G., BIBLIOWITCZ J., NOWROUZEZAHRAI D., BALAKRISHNAN R., SINGH K.: Video browsing by direct manipulation. In *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems* (Florence, Italy, 2008), ACM, pp. 237–246. 12

[DRCB05] DOLLAR P., RABAUD V., COTTRELL G., BELONGIE S.: Behavior recognition via sparse spatio-temporal features. In *Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on* (2005), pp. 65–72. 17

[DSTT00] DELLAERT F., SEITZ S., THORPE C., THRUN S.: Structure from motion without correspondence. vol. 2, pp. 557 –564. 18

[DT05] DALAL N., TRIGGS B.: Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2005), vol. 1, pp. 886–893. 16, 17

[FCSS09] FURUKAWA Y., CURLESS B., SEITZ S. M., SZELISKI R.: Reconstructing building interiors from images. pp. 80 –87. 18

[Fel05] FELZENSZWALB P.: Representation and detection of deformable shapes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 27*, 2 (feb. 2005), 208 –220. 16

[FG87] FÖRSTNER W., GÜLCH E.: A fast operator for detection and precise location of distinct points, corners and centers of circular features. In *Proceedings of the ISPRS Intercommission Workshop on Fast Processing of Photogrammetric Data* (1987), pp. 281–305. 16

[FJ90] FLEET D., JEPSON A.: Computation of component image velocity from local phase information. *International Journal of Computer Vision 5*, 1 (1990), 77–104. 15

[FJS96] FINKELSTEIN A., JACOBS C. E., SALESIN D. H.: Multiresolution video. In *Proceedings of SIGGRAPH 96* (aug 1996), pp. 281–290. 6

[FK] FAUGERAS O., KERIVEN R.: Complete dense stereovision using level set methods. In *Computer Vision Ǔ ECCV'98*, Lecture Notes in Computer Science. 18

[FM99] FELS S., MASE K.: Interactive video cubism. In *Proceedings of the 1999 workshop on new paradigms in information visualization and manipulation in conjunction with the eighth ACM internation conference on Information and knowledge management* (1999), ACM New York, NY, USA, pp. 78–82. 14

[FMR08] FELZENSZWALB P., MCALLESTER D., RAMANAN D.: A discriminatively trained, multiscale, deformable part model. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2008), pp. 1–8. 17

[FNZ*09] FLAGG M., NAKAZAWA A., ZHANG Q., KANG S. B., RYU Y. K., ESSA I., REHG J. M.: Human video textures. In *I3D '09: Proceedings of the 2009 symposium on Interactive 3D graphics and games* (New York, NY, USA, 2009), ACM, pp. 199–206. 7, 8

[FR98] FINKELSTEIN A., RANGE M.: Image mosaics. In *EP '98/RIDT '98: Proceedings of the 7th International Conference on Electronic Publishing, Held Jointly with the 4th International Conference on Raster Imaging and Digital Typography* (London, UK, 1998), Springer-Verlag, pp. 11–22. 6

[FXZM03] FAN X., XIE X., ZHOU H.-Q., MA W.-Y.: Looking into video frames on small displays. In *MULTIMEDIA '03: Proceedings of the eleventh ACM international conference on Multimedia* (New York, NY, USA, 2003), ACM, pp. 247–250. 8

[GCS02] GOOCH B., COOMBE G., SHIRLEY P.: Artistic vision: painterly rendering using computer vision techniques. In *NPAR '02: Proceedings of the 2nd international symposium on Non-photorealistic animation and rendering* (New York, NY, USA, 2002), ACM, pp. 83–ff. 5

[GCSS06] GOLDMAN D. B., CURLESS B., SALESIN D., SEITZ S. M.: Schematic storyboarding for video visualization and editing. *ACM Trans. Graph. 25*, 3 (2006), 862Ǔ871. 13

[GCT02] GIBSON D., CAMPBELL N., THOMAS B.: Visual abstraction of wildlife footage using gaussian mixture models and the minimum description length criterion. In *ICPR* (2002), pp. 814–817. 16

[GGC*08] GOLDMAN D. B., GONTERMAN C., CURLESS B., SALESIN D., SEITZ S. M.: Video object annotation, navigation, and composition. In *Proceedings of the 21st annual ACM symposium on User interface software and technology* (Monterey, CA, USA, 2008), ACM, pp. 3–12. 12

[GKV*07] GIRGENSOHN A., KIMBER D., VAUGHAN J., YANG T., SHIPMAN F., TURNER T., RIEFFEL E., WILCOX L., CHEN F., DUNNIGAN T.: Dots: support for effective video surveillance. In *Proceedings of the 15th international conference on Multimedia* (2007), ACM New York, NY, USA, pp. 423–432. 12

[GMN*98]  GALVIN B., MCCANE B., NOVINS K., MASON D., MILLS S.: Recovering motion fields: An evaluation of eight optical flow algorithms. In *British Machine Vision Conference* (1998), vol. 98. 15

[GRG]  GALL J., RAZAVI N., GOOL L. V.: On-line adaption of class-specific codebooks for instance tracking, year=2010. 18

[HCG05]  HANNUNA S., CAMPBELL N., GIBSON D.: Identifying quadruped gait in wildlife video. In *International Conference on Image Processing* (September 2005), IEEE. 15

[HCT*06]  HUANG Y., CHEN C., TSAI C., SHEN C., CHEN L.: Survey on block matching motion estimation algorithms and architectures with new results. *The Journal of VLSI Signal Processing 42*, 3 (2006), 297–320. 15

[HE04]  HAYS J., ESSA I.: Image and video based painterly animation. In *NPAR '04: Proceedings of the 3rd international symposium on Non-photorealistic animation and rendering* (New York, NY, USA, 2004), ACM, pp. 113–120. 4, 5

[HEH05]  HOIEM D., EFROS A. A., HEBERT M.: Automatic photo pop-up. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers* (2005), pp. 577–584. 18

[HP00]  HERTZMANN A., PERLIN K.: Painterly rendering for video and interaction. In *NPAR '00: Proceedings of the 1st international symposium on Non-photorealistic animation and rendering* (New York, NY, USA, 2000), ACM, pp. 7–12. 5

[HS81]  HORN B., SCHUNCK B.: Determining optical flow. *Computer Vision 17* (1981), 185–203. 15

[HS88]  HARRIS C., STEPHENS M.: A combined corner and edge detector. *Alvey Vision Conference* (1988), 147–151. 16

[IA98]  IRANI M., ANANDAN P.: Video indexing based on mosaic representations. In *Proceedings of the IEEE* (jun 1998), pp. 21–28. 6

[IAH95]  IRANI M., ANANDAN P., HSU S.: Mosaic based representations of video sequences and their applications. *Computer Vision, IEEE International Conference on 0* (1995), 605. 13

[JDD99]  JONES R. C., DEMENTHON D., DOERMANN D. S.: Building mosaics from video using mpeg motion vectors. In *MULTIMEDIA '99: Proceedings of the seventh ACM international conference on Multimedia (Part 2)* (New York, NY, USA, 1999), ACM, p. 29Ů32. 13

[JPA07]  JACQUEMIN C., PLANES B., AJAJ R.: Shadow casting for soft and engaging immersion in augmented virtuality artworks. In *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia* (New York, NY, USA, 2007), ACM, pp. 477–480. 9

[KCMT06]  KANG H., CHEN X., MATSUSHITA Y., TANG X.: Space-time video montage. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2006), vol. 2. 12

[KDG*07]  KIMBER D., DUNNIGAN T., GIRGENSOHN A., SHIPMAN F., TURNER T., YANG T.: Trailblazing: Video playback control by direct object manipulation. In *2007 IEEE International Conference on Multimedia and Expo* (2007), pp. 1015–1018. 12

[KGFC02]  KLEIN A. W., GRANT T., FINKELSTEIN A., COHEN M. F.: Video mosaics. In *NPAR 2002: Second International Symposium on Non Photorealistic Rendering* (jun 2002), pp. 21–28. 6

[KMB07]  KAKUMANU P., MAKROGIANNIS S., BOURBAKIS N.: A survey of skin-color modeling and detection methods. *Pattern Recognition 40*, 3 (2007), 1106 – 1122. 16

[KS00]  KUTULAKOS K. N., SEITZ S. M.: A theory of shape by space carving. *International Journal of Computer Vision 38* (2000), 199–218. 18

[KSE*03]  KWATRA V., SCHÖDL A., ESSA I., TURK G., BOBICK A.: Graphcut textures: Image and video synthesis using graph cuts. *ACM Transactions on Graphics, SIGGRAPH 2003 22*, 3 (July 2003), 277–286. 9

[KSFC02]  KLEIN A., SLOAN P., FINKELSTEIN A., COHEN M.: Stylized video cubes. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2002), ACM New York, NY, USA, pp. 15–22. 14

[KSH05]  KE Y., SUKTHANKAR R., HEBERT M.: Efficient visual event detection using volumetric features. In *Tenth IEEE International Conference on Computer Vision, 2005. ICCV 2005* (2005), vol. 1. 17

[KST*09]  KOUTSOURAKIS P., SIMON L., TEBOUL O., TZIRITAS G., PARAGIOS N.: Single view reconstruction using shape grammars for urban environments. pp. 1795 –1802. 18

[KTZ05]  KUMAR M. P., TORR P. H. S., ZISSERMAN A.: Obj cut. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2005), pp. 18–25. 16

[KUWS03]  KANG S. B., UYTTENDAELE M., WINDER S., SZELISKI R.: High dynamic range video. In *SIGGRAPH '03: ACM SIGGRAPH 2003 Papers* (New York, NY, USA, 2003), ACM, pp. 319–325. 8

[KWLB08]  KARRER T., WEISS M., LEE E., BORCHERS J.: Dragon: a direct manipulation interface for frame-accurate in-scene video navigation. *Proc. CHI* (2008), 247–250. 12

[Lap05]  LAPTEV I.: On space-time interest points. *Int. J. Comput. Vision 64*, 2-3 (2005), 107–123. 17

[LCL*97]  LEE M.-C., CHEN W.-G., LIN C., GU C., MARKOC T., ZABINSKY S., SZELISKI R.: A layered video object coding system using sprite and affine motion model. *Circuits and Systems for Video Technology, IEEE Transactions on 7*, 1 (Feb 1997), 130–145. 13

[LF06]  LOBAY A., FORSYTH D. A.: Shape from texture without boundaries. *International Journal of Computer Vision 67*, 1 (2006), 71–91. 18

[LG06]  LIU F., GLEICHER M.: Video retargeting: automating pan and scan. In *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia* (New York, NY, USA, 2006), ACM, pp. 241–250. 8

[LGS*00]  LI F., GUPTA A., SANOCKI E., HE L., RUI Y.: Browsing digital video. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2000), ACM New York, NY, USA, pp. 169–176. 11

[Lin98]  LINDEBERG T.: Feature detection with automatic scale selection. *Int. J. Comput. Vision 30*, 2 (1998), 79–116. 16

[Lit97]  LITWINOWICZ P.: Processing images and video for an impressionist effect. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1997), ACM Press/Addison-Wesley Publishing Co., pp. 407–414. 5

[LK81]  LUCAS B., KANADE T.: An iterative image registration technique with an application to stereo vision. 674–679. 15

[LL06]  LAPTEV I., LINDEBERG T.: *Local Descriptors for Spatio-temporal Recognition*, vol. Volume 3667/2006 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2006, pp. 91–103. 17

[LM01]  LUCCHESE L., MITRA S.: Colour image segmentation: a state-of-the-art survey. *PROCEEDINGS-INDIAN NATIONAL SCIENCE ACADEMY PART A 67*, 2 (2001), 207–222. 16

[Low04]  LOWE D. G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision 60*, 2 (2004), 91–110. 16

[LSS05]  LI Y., SUN J., SHUM H.-Y.: Video object cut and paste. *ACM Trans. Graph. 24*, 3 (2005), 595–600. 7

[McI00]  MCIVOR A.: Background subtraction techniques. 16

[MCUP04]  MATAS J., CHUM O., URBAN M., PAJDLA T.: Robust wide-baseline stereo from maximally stable extremal regions. In *Image and Vision Computing* (9 2004), vol. 22, British Machine Vision Computing 2002, pp. 761–767. 16

[MP07]  MOREELS P., PERONA P.: Evaluation of features detectors and descriptors based on 3d objects. *International Journal of Computer Vision 73*, 3 (2007), 263–284. 16

[MS05]  MIKOLAJCZYK K., SCHMID C.: A performance evaluation of local descriptors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 27*, 10 (10 2005), 1615–1630. 16

[MTS*05]  MIKOLAJCZYK K., TUYTELAARS T., SCHMID C., ZISSERMAN A., MATAS J., SCHAFFALITZKY F., KADIR T., GOOL L. V.: A comparison of affine region detectors. *Int. J. Comput. Vision 65*, 1-2 (2005), 43–72. 16

[MYYH08]  MEI T., YANG B., YANG S.-Q., HUA X.-S.: Video collage: presenting a video sequence using a single image. *Vis. Comput. 25*, 1 (2008), 39Ů51. 11, 13

[ND05]  NIENHAUS M., DOLLNER J.: Depicting dynamics using principles of visual art and narrations. *IEEE Comput. Graph. Appl. 25*, 3 (2005), 40–51. 13

[PD04]  PEKER K., DIVAKARAN A.: Adaptive fast playback-based video skimming using a compressed-domain visual complexity measure. In *2004 IEEE International Conference on Multimedia and Expo, 2004. ICME'04* (2004), vol. 3. 12

[PDS01]  PEKER K., DIVAKARAN A., SUN H.: Constant pace skimming and temporal sub-sampling of video using motion activity. In *Proc. IEEE IntŠl Conf. on Image Processing, Thessaloniki, Greece Oct* (2001). 12

[PESvG09]  PELLEGRINI S., ESS A., SCHINDLER K., VAN GOOL L.: You'll never walk alone: Modeling social behavior for multi-target tracking. pp. 261 –268. 17

[Pic04]  PICCARDI M.: Background subtraction techniques: a review. In *2004 IEEE International Conference on Systems, Man and Cybernetics* (2004), vol. 4. 16

[PJH05]  PETROVIC N., JOJIC N., HUANG T.: Adaptive video fast forward. *Multimedia Tools and Applications 26*, 3 (2005), 327–344. 12

[PRAGP07]  PRITCH Y., RAV-ACHA A., GUTMAN A., PELEG S.: Webcam synopsis: Peeking around the world. In *Proc. ICCV* (2007), pp. 1–8. 12

[PRAP08a]  PRITCH Y., RAV-ACHA A., PELEG S.: Nonchronological video synopsis and indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence 30*, 11 (2008), 1971–1984. 12

[PRAP08b]  PRITCH Y., RAV-ACHA A., PELEG S.: Nonchronological video synopsis and indexing. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 30*, 11 (Nov. 2008), 1971–1984. 13

[PS97]  PATEL N. V., SETHI I. K.: Video shot detection and characterization for video databases. *Pattern Recognition 30* (1997), 538–592. 10

[RAAKR05]  RADKE R., ANDRA S., AL-KOFAHI O., ROYSAM B.: Image change detection algorithms: a systematic survey. *Image Processing, IEEE Transactions on 14*, 3 (March 2005), 294–307. 16

[RAKRF08]  RAV-ACHA A., KOHLI P., ROTHER C., FITZGIBBON A.: Unwrap mosaics: a new representation for video editing. In *SIGGRAPH '08: ACM SIGGRAPH 2008 papers* (New York, NY, USA, 2008), ACM, pp. 1–11. 7

[RAPLP07]  RAV-ACHA A., PRITCH Y., LISCHINSKI D., PELEG S.: Dynamosaicing: Mosaicing of dynamic scenes. *IEEE Trans. Pattern Anal. Mach. Intell. 29*, 10 (2007), 1789–1801. 7, 8

[RAPP06]  RAV-ACHA A., PRITCH Y., PELEG S.: Making a long video short: Dynamic video synopsis. In *CVPR06* (2006), pp. 435–441. 12

[RB03]  RAMOS G., BALAKRISHNAN R.: Fluid interaction techniques for the control and annotation of digital video. In *Proceedings of the 16th annual ACM symposium on User interface software and technology* (2003), ACM New York, NY, USA, pp. 105–114. 11

[RBHB06]  ROTHER C., BORDEAUX L., HAMADI Y., BLAKE A.: Autocollage. *ACM Trans. Graph. 25*, 3 (2006), 847–852. 13

[RFZ07]  RAMANAN D., FORSYTH D., ZISSERMAN A.: Tracking people by learning their appearance. *Pattern Analysis and Machine Intelligence 29*, 1 (2007), 65 –81. 18

[RSA08]  RUBINSTEIN M., SHAMIR A., AVIDAN S.: Improved seam carving for video retargeting. *ACM Transactions on Graphics (SIGGRAPH) 27*, 3 (2008), 1–9. 9

[RSA09]  RUBINSTEIN M., SHAMIR A., AVIDAN S.: Multioperator media retargeting. *ACM Transactions on Graphics (Proceedings SIGGRAPH 2009) 28*, 3 (2009), 1–11. 9

[RV05]  ROMDHANI S., VETTER T.: Estimating 3d shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. vol. 2, pp. 986 – 993. 18

[RWY90]  REISFELD D., WOLFSON H., YESHURUN Y.: Detection of interest points using symmetry. *Computer Vision, 1990. Proceedings, Third International Conference on* (12 1990), 62–65. 16

[SAS07]  SCOVANNER P., ALI S., SHAH M.: A 3-dimensional sift descriptor and its application to action recognition. In *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia* (New York, NY, USA, 2007), ACM, pp. 357–360. 17

[SB09]  SCHOEFFMANN K., BOESZOERMENYI L.: Video browsing using interactive navigation summaries. *International Workshop on Content-Based Multimedia Indexing 7* (2009), 243–248. 11, 12

[SC96]  SZELISKI R., CORPORATION M.: Video mosaics for virtual environments. *IEEE Computer Graphics and Applications 16* (1996), 22–30. 9

[SCM02]  SUN X., CHEN C.-W., MANJUNATH B. S.: Probabilistic motion parameter models for human activity recognition. *International Conference on Pattern Recognition 1* (2002). 17

[SCRS09]  SHESH A., CRIMINISI A., ROTHER C., SMYTH G.: 3d-aware image editing for out of bounds photography. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (05 2009). 7

[SE02]  SCHÖDL A., ESSA I. A.: Controlled animation of video sprites. In *SCA '02: Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation* (New York, NY, USA, 2002), ACM, pp. 121–127. 8

[Sio07]  SION H.: Quadruped gait detection in low quality wildlife video. *Ph.D. Thesis, Univeristy of Bristol* (2007). 16

[SKD09]  SANTOSH K. DIVVALA DEREK HOIEM J. H. H. A. A. E. M. H.: An empirical study of context in object detection. In *Computer Vision and Pattern Recognition* (June 2009). 17

[SLNG07] SETLUR V., LECHNER T., NIENHAUS M., GOOCH B.: Retargeting images and video for preserving information saliency. *IEEE Comput. Graph. Appl. 27*, 5 (2007), 80–88. 9

[SS04] SEZGIN M., SANKUR B.: Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging 13* (2004), 146. 16

[SSSE00] SCHÖDL A., SZELISKI R., SALESIN D. H., ESSA I.: Video textures. In *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 2000), ACM Press/Addison-Wesley Publishing Co., pp. 489–498. 7

[ST94] SHI J., TOMASI C.: Good features to track. In *CVPR* (1994), pp. 593–600. 15, 16

[Ste04] STEIN F.: Efficient computation of optical flow using the census transform. *Lecture notes in computer science* (2004), 79–86. 15

[STR*05] SETLUR V., TAKAGI S., RASKAR R., GLEICHER M., GOOCH B.: Automatic image retargeting. In *MUM '05: Proceedings of the 4th international conference on Mobile and ubiquitous multimedia* (New York, NY, USA, 2005), ACM, pp. 59–68. 8, 9

[TAT97] TANIGUCHI Y., AKUTSU A., TONOMURA Y.: Panoramaexcerpts: extracting and packing panoramas for video browsing. In *MULTIMEDIA '97: Proceedings of the fifth ACM international conference on Multimedia* (New York, NY, USA, 1997), ACM, pp. 427–436. 13

[TB02] TORRESANI L., BREGLER C.: Space-time tracking. In *European Conference on Computer Vision* (2002), pp. 801–812. 18

[TB05] TEODOSIO L., BENDER W.: Salient stills. *ACM Trans. Multimedia Comput. Commun. Appl. 1*, 1 (2005), 16–36. 11, 13

[TV07] TRUONG B., VENKATESH S.: Video abstraction: A systematic review and classification. *ACM Transactions on Multimedia Computing, Communications and Applications 3*, 1 (2007). 10, 12, 13

[TYAB01] TORRESANI L., YANG D., ALEXANDER E., BREGLER C.: Tracking and modeling non-rigid objects with rank constraints. vol. 1, pp. 493–500. 18

[UFF06] URTASUN R., FLEET D. J., FUA P.: 3d people tracking with gaussian process dynamical models. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2006), pp. 238–245. 17

[UFGB99] UCHIHASHI S., FOOTE J., GIRGENSOHN A., BORECZKY J.: Video manga: generating semantically meaningful video summaries. In *MULTIMEDIA '99: Proceedings of the seventh ACM international conference on Multimedia (Part 1)* (New York, NY, USA, 1999), ACM, pp. 383–392. 13

[VBMP08] VLASIC D., BARAN I., MATUSIK W., POPOVIĆ J.: Articulated mesh animation from multi-view silhouettes. *ACM Trans. Graph. 27*, 3 (2008), 1–9. 8

[VJ01] VIOLA P., JONES M. J.: *Robust real-time object detection*. Tech. Rep. CRL 2001/01, Cambridge Research Laboratory, 2 2001. 16, 17

[VTC05] VOGIATZIS G., TORR P., CIPOLLA R.: Multi-view stereo via volumetric graph-cuts. vol. 2, pp. 391 – 398. 18

[WB95] WELCH G., BISHOP G.: *An Introduction to the Kalman Filter*. Tech. rep., 1995. 17

[WB99] WILSON A. D., BOBICK A. F.: Parametric hidden markov models for gesture recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence 21*, 9 (1999), 884–900. 17

[WBBP06] WEICKERT J., BRUHN A., BROX T., PAPENBERG N.: A survey on variational optic flow methods for small displacements. *MATHEMATICS IN INDUSTRY 10* (2006), 103. 15

[WBC*05a] WANG J., BHAT P., COLBURN R., AGRAWALA M., COHEN M.: Interactive video cutout. In *International Conference on Computer Graphics and Interactive Techniques* (2005), ACM New York, NY, USA, pp. 585–594. 14

[WBC*05b] WANG J., BHAT P., COLBURN R. A., AGRAWALA M., COHEN M. F.: Interactive video cutout. *ACM Trans. Graph. 24*, 3 (2005), 585–594. 7

[WC07] WANG J., COHEN M. F.: Image and video matting: a survey. *Found. Trends. Comput. Graph. Vis. 3*, 2 (2007), 97–175. 7

[WGCO07] WOLF L., GUTTMANN M., COHEN-OR D.: Nonhomogeneous content-driven video-retargeting. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on* (Oct. 2007), pp. 1–6. 9

[WMY*03] WILDEMUTH B., MARCHIONINI G., YANG M., GEISLER G., WILKENS T., HUGHES A., GRUSS R.: How fast is too fast? evaluating fast forward surrogates for digital video. In *Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries* (2003), IEEE Computer Society Washington, DC, USA, pp. 221–230. 12

[WOG06] WINNEMÖLLER H., OLSEN S. C., GOOCH B.: Real-time video abstraction. *ACM Trans. Graph. 25*, 3 (2006), 1221–1226. 5

[WRL*04] WANG J., REINDERS M. J. T., LAGENDIJK R. L., LINDENBERG J., KANKANHALLI M. S.: Video content representation on tiny devices. 1711–1714. 8

[WXSC04] WANG J., XU Y., SHUM H.-Y., COHEN M. F.: Video tooning. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers* (New York, NY, USA, 2004), ACM, pp. 574–583. 5, 7

[WZY*08] WANG X., ZHANG Q., YANG R., SEALES B., CARSWELL M.: Feature-based texture mapping from video sequence. In *I3D '08: Proceedings of the 2008 symposium on Interactive 3D graphics and games* (New York, NY, USA, 2008), ACM, pp. 1–1. 7, 8

[XWL*08] XU X., WAN L., LIU X., WONG T.-T., WANG L., LEUNG C.-S.: Animating animal motion from still. *ACM Transactions on Graphics 27*, 5 (dec 2008), 117:1–117:8. 7

[YY97] YEUNG M., YEO B.-L.: Video visualization for compact presentation and fast browsing of pictorial content. *Circuits and Systems for Video Technology, IEEE Transactions on 7*, 5 (Oct 1997), 771–785. 13

[ZL08] ZHAO X., LIU Y.: Generative tracking of 3d human motion by hierarchical annealed genetic algorithm. *Pattern Recognition 41*, 8 (2008), 2470 – 2483. 17

[ZM97] ZHU S., MA K.: A new diamond search algorithm for fast block matching motionestimation. 15

[ZPQS07] ZENG G., PARIS S., QUAN L., SILLION F.: Accurate and scalable surface representation and reconstruction from images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 29*, 1 (2007), 141 –158. 18