DOI: 10.1111/j.1467-8659.2012.03158.x

COMPUTER GRAPHICS *forum*
Volume 31 (2012), number 8 pp. 2450–2477

State of the Art Report on Video-Based Graphics and Video Visualization

R. Borgo[1], M. Chen[2], B. Daubney[1], E. Grundy[1], G. Heidemann[3], B. Höferlin[4], M. Höferlin[4], H. Leitte[5], D. Weiskopf[4] and X. Xie[1]

[1]Swansea University, UK
{r.borgo, b.daubney, csed, x.xie}@swansea.ac.uk
[2]Oxford University, UK
min.chen@oerc.ox.ac.uk
[3]Universität Osnabrück, Germany
gheidema@uni-osnabrueck.de
[4]Universität Stuttgart, Germany
{benjamin.hoeferlin,markus.hoeferlin}@vis.uni-stuttgart.de
weiskopf@visus.uni-stuttgart.de
[5]Universität Heidelberg, Germany
heike.leitte@iwr.uni-heidelberg.de

Abstract

In recent years, a collection of new techniques which deal with video as 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 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 provide 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 modelling and rendering pipelines for graphics and visualization.

Keywords: video-based graphics, video visualization, video processing, video summarization, video abstraction, video retargeting, video-based modelling

ACM CCS: I.2.10 [Artificial Intelligence]: Vision and Scene Understanding—Video Analysis I.3.3 [Computer Graphics]: Picture/Image Generation— I.3.8 [Computer Graphics]: Applications—Visualization I.4.8 [Image Processing and Computer Vision]: Scene Analysis—Time-varying Imagery H.5.1 [Information Interface and Presentation]: Multimedia Information Systems—Video

1. Introduction

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

Video-based graphics is concerned with the manipulation and rendering of graphical models 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 refashioning* and *video-based scene modelling*. The former typically involves manipulation of the geometrical entities (e.g. object shape and distribution) and optical attributes (e.g. lighting, colour) of an input video, producing a new video that captures the essence of the input but in an expressive art form, such as by relighting the video scene with imaginary lights or mimicking hand-drawn cartoon animation. The latter, meanwhile, involves reconstruction of a three-dimensional (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 primary motivation for video-based graphics has been consumer multimedia applications, and the film and game 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 summarized visual representations. Video visualization is not intended to provide fully automatic solutions to the problem of making decisions about the contents of a video. Instead, it aims at offering a tool to assist users in their intelligent reasoning while removing the burden of viewing videos. This aim justifies deviation from the creation of realistic imagery (as found in video-based graphics), and allows simplifications and embellishments, to improve the understanding of the input video. In many ways, the subject of video visualization encompasses some aspects of video-based graphics. The development of the subject has been heavily influenced by many applications in science, medicine, sport 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 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 observe that these three fields share a substantial amount of functional components, although having dissimilar aims. This survey focuses on video-based graphics and video visualization. To provide readers with a brief background about various functional components found in the literature of image processing, computer vision and multimedia technology, we also provide an overview section on video analysis.

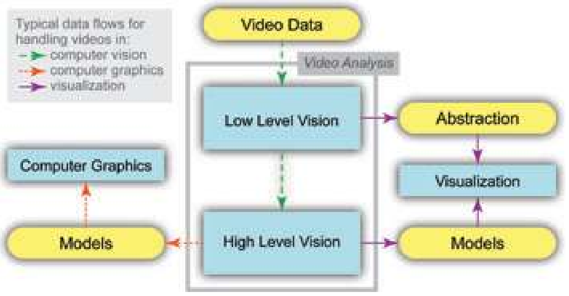


Figure 1: *Typical computational pipelines for video-based graphics, video visualization and video analysis. This survey focuses on the first two fields while giving a brief overview of techniques in video analysis.*

2. Taxonomy

Video-based graphics and video visualization are relatively new developments in visual computing. It is 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 accompany the input video and (iv) the level of automation.

Like categorization problems in many applications, classification schemes in a taxonomy cannot always ensure that every entity falls into only one category exclusively. In our case, it is unavoidable that some previous and future works may fall into a few classes, often due to adoption of a hybrid approach or generalization of a concept across different applications. Nevertheless, the majority of the works surveyed in this paper do not incur any ambiguity in classification, especially if we consider their primary features.

2.1. Classification by goals

As stated in Section 1, *video-based graphics* and *video visualization* differ by their goals. We define two distinguishable categories (Figure 2):

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.

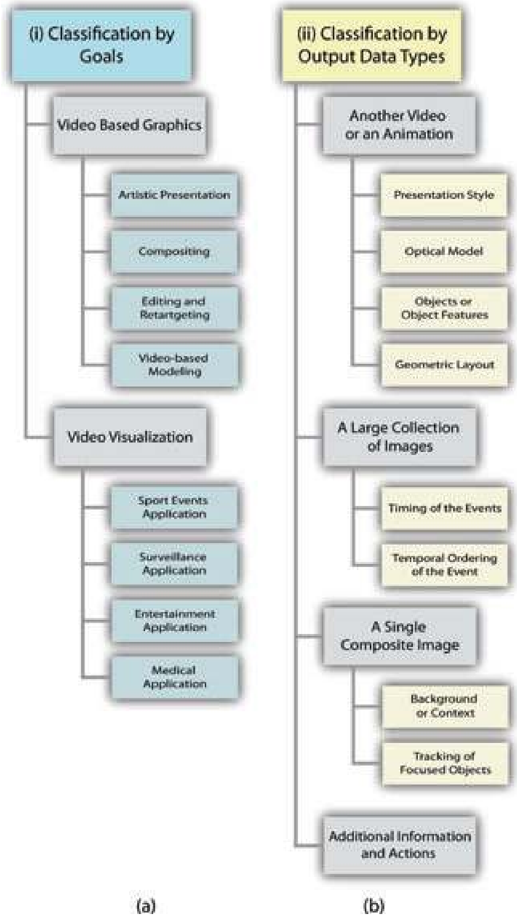


Figure 2: *First and second classifications proposed by our survey: (a) by goals; (b) by type of output data.*

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 (NPR) technique (e.g. painting) and transforming a video to a more 'expressive' cartoon representation.

A1.2. *Compositing*—focuses on creating an artistic image by selectively mixing content 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. relighting). Although, editing and retargeting can be done on a frame-by-frame basis, video-based approaches address the need for temporal coherence.

A1.4. *Video-based modelling*—focuses on creating graphical models from videos to enhance the perception of spatial and dynamic features of a scene. This ranges from video-based panorama composition to 3D object reconstruction. The primary use of this class of techniques is the modelling 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, computer vision has not yet been able to deliver automated technology to fulfill such goals in the general case. Video visualization, which keeps the user in the loop, is a 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

Although videos are the principal input to the techniques covered by this survey, the outputs can vary considerably. Typical data types of the output are (Figure 2):

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

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

B3. **A single composite image**—where the composite can be as simple as an annotated key frame, or as complex as a composite 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**—where informations and actions accompany 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' describes

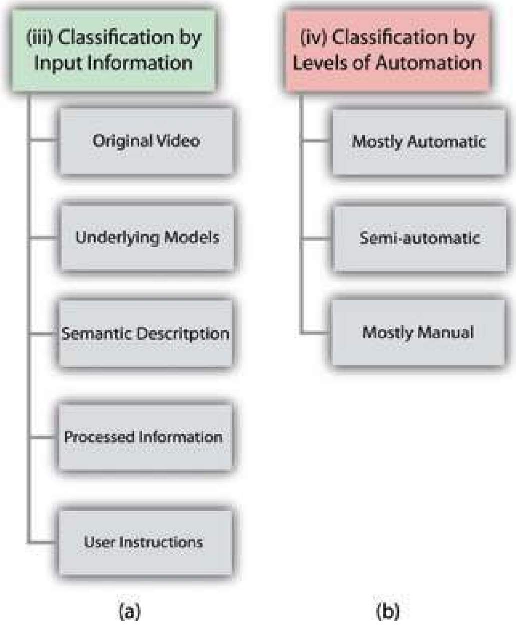


Figure 3: *Third and fourth classifications proposed by our survey: (a) by input data; (b) by level of automation.*

information attached to objects to facilitate interactive activities, such as hyperlinks and hot spots.

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 distinguishing 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:

B.1.1. *Presentation style*– e.g. photo-realistic, pen-and-ink, water-colour, etc.
B.1.2. *Optical model*—e.g. lighting, focus, atmospheric effects, etc.
B.1.3. *Objects or object features*—e.g. object replacement, etc.
B.1.4. *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:

B.2.1. *Timing of events*.
B.2.2. *Temporal ordering of events*.
B.3.1. *Background or context*.
B.3.2. *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 to supplement the original video. The users can also provide additional information manually. Hence, we can also consider a classification based on the input information, which may include (Figure 3):

C1. **Original video**.
C2. **Underlying models**—a floor plan, a 3D environmental model.
C3. **Semantic descriptions**—a face to be detected.
C4. **Processed information**—optical flow data.
C5. **User instructions**—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 (Figure 3):

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

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 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 subgoals, 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.

2.6. Terminology

The development of a new field is often accompanied by a confusion of terminology. Over the last decade, various terms have been used to encompass the body of the works surveyed in this paper. For example, *video abstraction* was sometimes used in the context of artistic transformation of videos as well as video summarization, even though the resulting imagery is not necessarily in an abstract form. There was also a proposal for using *video presentation* to encompass both *video-based graphics* and *video visualization*. However, this phrase can easily lead to confusion with activities for presenting visual information using videos. Perhaps even more actual would be the phrases *video representation* or *video transformation*. However, the former shares similarities with

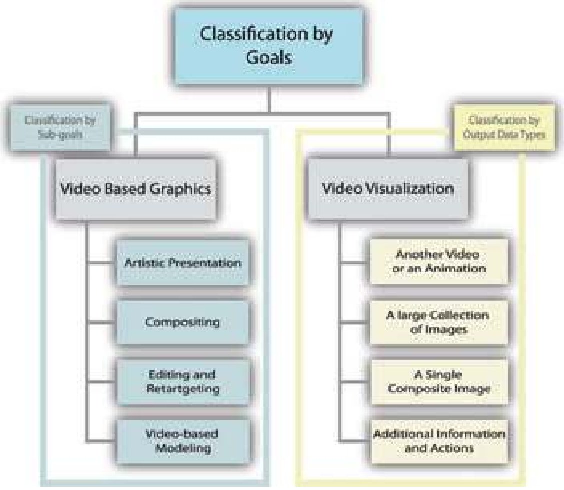


Figure 4: *Taxonomy used in the survey.*

video representation which has a very different meaning, although the latter does not encapsulate the purposes very well. Furthermore, both phrases also compass works that are not considered in this survey (e.g. video restoration). We hence adopt the phrases *video-based graphics* and *video visualization* in this survey.

3. 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 which 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.

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 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 on improving low-level methods as are invested in developing high-level approaches.

3.1. 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 designed to exploit it.

3.1.1. *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 neighbouring 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 MPEG4 [ZM97]. A comprehensive survey of block matching techniques is provided 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, however, are capable of achieving subpixel 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 given by 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. This is where locally the motion between two frames is ambiguous and can not be uniquely identified. This results in some regions for which the motion is unknown or poorly estimated, for example large motions can often be incorrectly observed along the edges of objects.

Global methods solve the same first-order Taylor expansion of image motion, but 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 imply large gradients (i.e. edges) in adjacent directions of the patch, 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 because 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 compared with global methods in sequences where there is little texture. Another approach is that of Bruhn *et al.* [BWS05], who investigate combining local and global methods to extract optical flow, which is achieved by using local confidence measures and effectively growing a dense representation.

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 neighbourhood 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 the reader to [WBBP06].

3.1.2. *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 colour/texture [SS04] and often the spatial location of the pixels are exploited preferring neighbouring 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. However, 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 methods must often 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 a particular scene to detect (segment) any pixels or regions that change temporally. An evaluation of current background subtraction techniques applied to the most challenging conditions, such as during gradual illumination changes or whilst observing a scene containing a dynamic background, is provided in [BHH11].

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, e.g. 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 leads to 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. Whilst methods that segment homogeneous regions can be treated as a black box and for images that contain little clutter can achieve acceptable results.

3.1.3. *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 colour, whilst local features describe the properties of only a small region.

The key advantage of local features is that extracted information 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. Although surprising, if applied to a tightly constrained problem, global features can yield encouraging results. For example, wildlife frames containing quadrupeds can be detected using the image's power spectrum [Sio07], which effectively describes the dominance of each frequency in constructing the image.

Some global features may be learnt adaptively for a specific video clip, for example statistical techniques such as principal component analysis can be used to project entire frames into a two-dimensional (2D) or 3D space allowing a complete video to be easily visualized. Furthermore, clustering this low-dimensional representation permits automatic key frame extraction [GCT02].

However, the limitation of global features to provide information about specific regions of an image constraints their use in video analysis; their strength lies in applications where the interest is in looking at large scale properties of an image sequence, e.g. 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 Kanade–Lucas–Tomasi feature tracker [ST94] discussed in Section 3.1.1 defines an interest feature as an image patch with a covariance with two large eigenvalues. Other standard interest feature detectors include the Harris corner detector [HS88], Förstner-Gülch [FG87] and the Reisfeld symmetry operator [RWY90].

Within the last decade, invariant local features have became popular including approaches like *SIFT* [Low04] or *SURF* [BETVG08] that rely for scale adaption on the scale-space theory introduced by Lindeberg [Lin98]. Other techniques, such as *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.* [MTS*05, MS05]. 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 [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 much simpler than their interest point detector counterparts and therefore less discriminative.

Thus far all features presented are only spatial in nature. However, often these features can be extended to the temporal domain, e.g. in the form of 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 by Laptev and Lindeberg [Lap05, LL06]. Whilst all of the above features are hand designed, a promising technique is to use machine learning techniques to automatically engineer low-level features [LZYN11].

3.2. 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*.

3.2.1. *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 as a 'one of *N*' classification problem. 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 cannot be achieved. A recognition task however, is often more constrained, e.g. given a text character, what letter is it most likely to be?

A recognition or detection system is composed of two parts, a set of low-level features such as those discussed in Section 3.1.3 and a classifier which will be trained using examples of each class. Popular classifiers include decision trees, neural networks, *AdaBoost*, support vector machines (SVM) and *k*-nearest neighbours. There are 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 labelled examples and cross-validation may be used to prevent overfitting 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 the 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]). Modelling context can also be used to improve detection accuracy (for a recent review see [DHH*09]).

For classifying sequential data hidden Markov models, 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]. 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].

One of the difficulties with action recognition is that often it is not clear, in a temporal sense, exactly where an action starts and where an action finishes. This can lead to difficulties in creating a consistent training set of positive and negative examples for a given action. However, methods such as multiple instance learning can be used to address this problem. This requires that for each positive example a positive event is known to have occurred without specifying the exact temporal location or duration. This has been applied to, for example, detecting shoppers picking items off of a shelf [HCL*09] and automatically learning sign language from TV subtitles [BEZ09].

3.2.2. *Tracking*

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, this is currently 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 number of established tracking algorithms, most notably the Kalman filter (tutorial provided in [WB95]), which assumes Gaussian noise and a linear dynamic model, and the particle filter (a tutorial is provided in [AMGC02]), 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 [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 more efficiently.

Most recent developments made in the field of tracking have been domain specific, in particular modelling 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 PCA [ZL08] have been used to learn action-specific models (e.g. walking) so that 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 [GRG10].

3.2.3. *3D reconstruction*

There are many methods used to extract a 3D representation of a scene or object observed in video. Well-established methods include approaches such as *structure from motion* (SfM) [DSTT00], space carving and stereo reconstruction. Whilst we briefly discuss these well-established techniques, we also discuss methods that typically attempt to reconstruct 3D structure from single images using cues such as shading, shape and texture. A good overview of vision-based 3D reconstruction is provided in [Sze11], and a recent survey, focusing on reconstruction from video, is provided in [SAB*07].

SfM takes a set of images and attempts to extract both a 3D reconstruction and the camera's motion within this. This is typically achieved by finding point correspondences across multiple images. 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*09a]. Furthermore, a dense reconstruction can be estimated through the use of *a priori* knowledge, such as assuming all surfaces are planar [FCSS09] or by applying stereo matching using the initial structure as a set of constraints. 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 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]. An approach that can cope with much larger deformations is to segment the object into a piecewise model and reconstruct each piece independently. The problem then becomes one of robustly segmenting the features, which can be achieved using energy minimization [RFA11]. 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]. There are many good tutorials and text books on structure from motion, such as that of Hartley and Zisserman [HZ04] or more recently Moons *et al.* [MGV09].

Another method, closely related to structure from motion, is vision-based Simultaneous Localization and Mapping (SLAM), where feature tracking is performed online to create a sparse map and also track the camera within this map. Typically, the focus in this work is on achieving real-time performance using a typical consumer web cam [DRMS07]. This has made the approach particularly applicable to creating augmented realities [CGCMC07, KM07] with SLAM systems that can operate on mobile phone devices [KM09]. For the purpose of augmented reality, a dense reconstruction is not typically needed as higher lever structures such as planar surfaces can be inferred from the extracted sparse representation. Whilst the typical SLAM frame work used for tracking a monocular camera tightly entwines the tracking and mapping into a single estimation problem, a recent approach is Parallel Tracking and Mapping [KM07]. This performs tracking and mapping as two separate tasks allowing the mapping to be done as a batch operation using a much larger set of features, whilst tracking is still performed on a frame-by-frame basis. This results in an approach that is more robust to tracking failure and produces more accurate maps.

Stereo matching across two images can be used to create a dense depth map. Typically, an important step in this process is image rectification, where the epipolar lines of the images are rectified to be parallel to the horizontal axis of the image. This allows matching between pixels to be performed along scan lines of the image rows. The difference in position between two pixels that observe the same point is called the disparity. The matching of pixels to estimate the disparity between two images can be performed via a number of standard optimization techniques such as dynamic programming [VMVPVG02], level sets [FK98], graphcuts [VTC05] and loopy belief propagation [FH06]. A survey and evaluation of existing stereo reconstruction methods is provided by Seitz *et al.* [SCD*06].

An alternative technique to dense reconstruction is space carving, which requires a predefined ‘search space’ to be constructed in which the object or scene of interest is assumed to be contained. 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. Another approach is to use foreground silhouettes to extract the visual hull of an object [Lau94]. Results from this method are typically poorer than those using colour consistency or texture, but the method can be used to initialize more complex approaches. A review of methods used to extract a 3D reconstruction of complex, often moving deformable, objects from multiple views in a studio setting is provided in [SMN*09]. A discussion of practical issues such as illumination and camera placement is also presented in this work.

It is 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 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 be 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 multiple views. However, for many tasks the results are qualitatively 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 that the area of 3D reconstruction coupled with machine-learning techniques will continue to receive much attention over the coming years.

4. Video-Based Graphics

Like images, videos can provide computer graphics with spatial information of the scene (e.g. in image-based modelling 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 tools to harvest. A useful overview of both techniques and terminology can be found in [Ass09b].

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 to create animations, e.g. oil-based painting, and watercolours. Such techniques are occasionally used in cinema to convey emotional effects. It is believed that automatic, 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 counterintuitive, 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.

Hertzmann and Perlin [HP00] address this by only 'repainting' 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 frame rates of 30Hz produce a video which can look 'too real' because 'the underlying motion and shape is integrated so well by the human visual system', and suggest frame rates of 10–15 Hz to accentuate the NPR feel.

Optical flow and mean-shift segmentation 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 (see Figure 5).

The use of optical flow in the above methods generally intends to solve two problems: segmentation and direction coherence. Wang *et al.* [WXSC04] employ a different method for segmenting the video data, and do not consider the problem of aligning brush strokes. The authors use a mean-shift segmentation 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 identified by the mean-shift operator. These segments then prevent flicker or popping artefacts from occurring during the segmentation stage of the abstraction process.

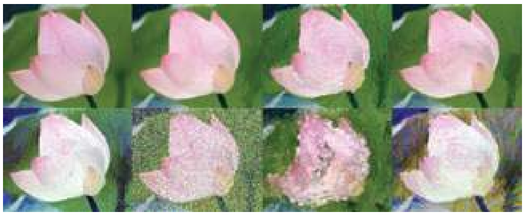


Figure 5: *Painterly rendering of a flower, from top-left in clockwise order: watercolour, Van Gogh, impressionism, abstract, pointillism and flower styles. (Image courtesy of Hays et al. [HE04], ©2004 ACM.)*

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 deposition 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] present 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, visual results similar to mean-shift segmentation are achieved.

4.2. Compositing

Time and space are intermixed components of a video, the 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 highlighting feature-events to enrich the video experience.

Pioneer work in the fieldis represented by the *multi-resolution 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 highlighting of varying features and events within a unique multivariate video. Finklestein *et al.* [FJS96] 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.* [FJS96] paved the way for the employment of videos in a variety of applications ranging from immersive environments with the use of interactive visualizations 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 major issues related to the selection of informative key frames or poses, maximization of screen space utilization and avoid cluttering or occlusion although maximizing the conveyed visual information. To address these issues, techniques like multi-resolution and clustering methods are borrowed from the visualization field to achieve coherence in time and space when visualizing highly detailed scenes at interactive frame rates.

4.2.1. *Mosaicing*

Mosaicing is the art of creating patterns or pictures by assembling small pieces of coloured 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 semantic similarity function beneath the mosaic structure. A video screen, as a collection of colour-varying pixels, is in itself an example of a digital mosaic. With the advent of digital photography pixels and tile materials could soon be replaced by collections of small images giving birth to what is now known as *image mosaic*. As a visual medium, image mosaics correspond to a carefully arranged collection of small images that when seen at a proper distance (or resolution) form a recognizable larger image [FR98]. The entertainment industry has exploited the idea behind image mosaics to create large film posters composed by carefully chosen and assembled video key frames; image tiles often undergo colour adjustment to improve the quality of the final result.

Beside being an aesthetically pleasing visual medium, *video posters* represent a powerful resource for interactive exploration of video sequences. Solutions have been developed that rely on video posters for video browsing to address the issue of minimizing user time whereas 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 (see Figure 6) of the action (*clip trailer*). Dynamic stills differ from standard image synopsis [IA98] as they allow self-occluding pose slices, although image mosaicing techniques usually rely on distribution and translation of objects trying to avoid replication or self-intersection (as in [IA98]).

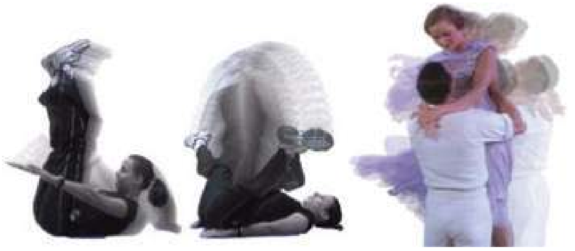


Figure 6: *Dynamic still and clips. The transparency of additional poses is based on their importance. Most informative poses (i.e. motion extreme points) are completely opaque. (Image courtesy of Caspi et al. [CAMG06] ©2006 SpringerLink.)*

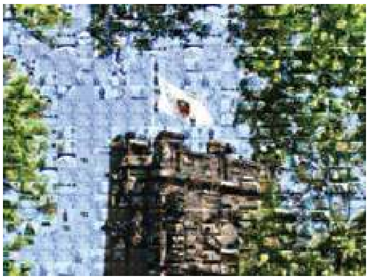


Figure 7: *A frame from a video mosaic. (Image courtesy of Klein et al. [KGFC02].)*

A quite different approach has been proposed by Klein *et al.* [KGFC02]; their technique denoted as *video mosaics* uses video frames rather than key frames as tiling units of the mosaic composition. Video mosaics stretch Finkelstein's multi-resolution video concept [FJS96] (see Figure 7): each video tile becomes a collection of layered images although 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 dimensions, so can space as in panoramic mosaicing or panoramas.

4.2.2. *Panoramas*

The concept of image panoramas dates back to the mid-19th century with 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 explicitly, 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*

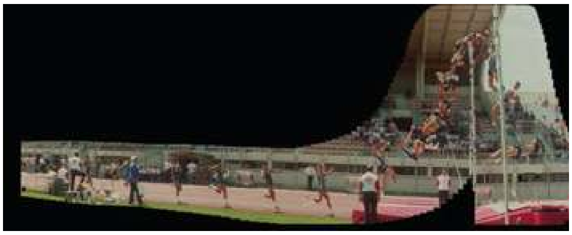


Figure 8: *Background and motion panoramas of a jumping athlete. (Image courtesy of Bartoli et al. [BDH04] ©2004 Wiley.)*

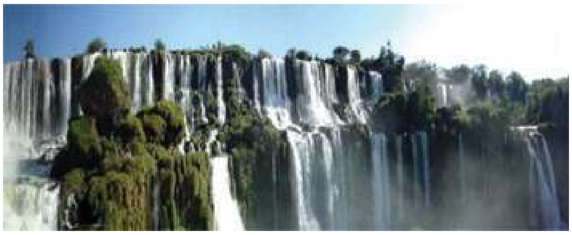


Figure 9: *Dynamosaic of a waterfall video. (Image courtesy of Rav-Acha et al. [RAPLP07] ©2007 IEEE.)*

layer corresponding to the moving objects and a *static layer* corresponding to the static background. The panoramic image output is composited by merging the two 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, key poses 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 explicitly as a video sequence. The resulting panorama becomes 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, *dynamosaicing* [RAPLP07] is better suited. *Dynamosaicing* (see Figure 9) recalls the video cube concept. First, an aligned space-time volume is constructed from the input video, secondly a continuous 2D plane (time front) is swept through that volume generating the sequence of images. Alignment is performed via key frame interpolation introducing a cost function to minimize artefacts due to chaotic moving objects. The natural step from dynamic panoramas to video textures is short as we can already see with the PVT. This intriguing aspect of extending video to augment visual appreciation of synthetic scene is explored in Sections 4.3.1 and 4.4.

4.2.3. *Cut-outs*

Video cut-outs is a hybrid approach between mosaics, panoramas and retargeting techniques (see Section 4.3.3). Video cut-out techniques allow for the extraction of foreground or background objects from video sequences for use in a variety of applications including compositing onto new backgrounds and NPR cartoon style rendering. Even when the continuous temporal information is lost, as in still shots, smooth and realistic motion can still be synthesized [XWL*08, SCRS09] by finding the motion path connecting the motion snapshots and generating for example cartoon like animations [WXSC04]. Reverse engineering this process allows for the extraction of moving objects from general backgrounds and for the development of sophisticated interactive systems [LSS05, WBC*05] for background substitution, object removal and reconstruction [RAKRF08].

A more sophisticated and commercially oriented version of the video cut-outs process is *video matting*, e.g. the process of extracting a high-quality alpha matte and foreground from a video sequence. Video matting concentrates on the problem of accurate foreground estimation in images and videos and represents a crucial operation in commercial television and film production giving a director the power to insert new elements seamlessly into a scene or to transport an actor into a completely new location. State of the art in video matting has significantly advanced recently, a good source of reference can be found in [WC07]. One of the latest achievements in interactive video editing is represented by Bai *et al.* [BWSS09] with their *SnapCut* system which extends state of the art algorithms for both object cut-outs and matting to videos.

4.3. Editing and retargeting

4.3.1. *Video textures*

Video textures [SSSE00] replace static images like digital photos with synthesized video sequences, enriching textured objects or scenes 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 transition 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] although [FNZ*09] used 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 greater control over the output animation sequence. For video texture mapping over a 3D model, as in [WZY*08], extension of parameterized texture mapping techniques is 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. PVT provide a continuous infinitely varying stream of images which easily extends to several applicative domains; PVT can be employed in the creation of contiguous video loops, single moving objects can be extracted and employed as *video sprites* [SE02] for feature-based texture mapping of 3D models [WZY*08], photo-realistic animation of human motion [VBMP08, FNZ*09] or 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. This sphere 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 (film production and special effects), CAD, augmented reality, face recognition, etc.

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.).

Akers *et al.* [ALK*03] describe the use of image relighting techniques to construct images which better convey the shape and texture of an object, one example being our moon, the image of which is constructed from a time-lapse sequence of the 12 phases occurring in one month (see Figure 10).

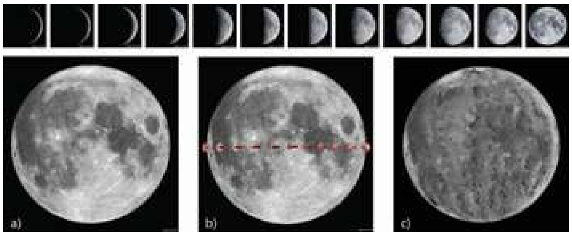


Figure 10: *Twelve photographs of the moon. (a) Unmodified photograph. (b) Control arrows to define a field of incident light direction. (c) Resulting composite photograph. (Image courtesy of Akers et al. [ALK*03] ©2003 IEEE.)*

Other methods for processing the lighting of a video have been described in what may roughly be 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] used pixel values from 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 perceptibility. Wang *et al.* [WDC*08a] supplemented low-quality digital video with an infrared video signal. As the infrared reflectance of a surface is less affected by the incoming light direction, this signal is used to provide edge and contrast information for areas of a scene with low light levels.

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 hardware is prohibitively expensive.

Rubinstein *et al.* [RGSS10] presented the first evaluation of retargeting algorithms, using both subjective (e.g. viewer’s preferences) and objective metrics. The perceptual study was based on a public available benchmark of images named RetargetMe [RGSS12].

4.3.3. *Video retargeting*

Video retargeting attempts to resize an input video to be 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 input devices, makes manual 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 manual 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

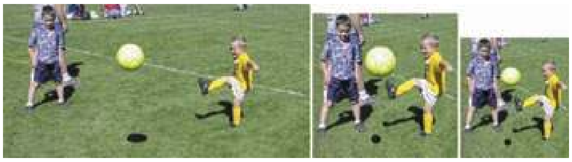


Figure 11: *Importance preserving image retargeting. The three important elements of the image are preserved as the image size is reduced. (Image courtesy of Setlur et al. [STR*05] ©2005 ACM.)*

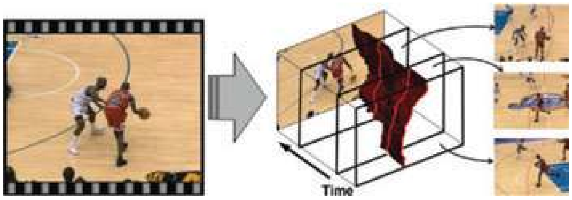


Figure 12: *Improved seam-carving accounts for frame changes over time, creating a consistent viewing experience. (Image courtesy of Rubinstein et al. [RSA08] ©2008 ACM.)*

affecting the narrative constructed by the director [LG06a] (which is also a common criticism of the manual method).

An alternative approach fully embraced the flexibility allowed by digital storage and avoided the need for homogenity in the retargeted scene; allowing an image to be warped, replacing large parts of the background with relatively smaller 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.* [SLNG07b].

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 chooses the most effective transformation method based on local properties [RSA09] to find the optimal retargeting of an input (see Figure 13). Wang *et al.* [WHSL11] supplement the optimization with automatic pan-and-scan and cropping techniques which prioritize motion information, reducing the amount of distortion applied to frames.

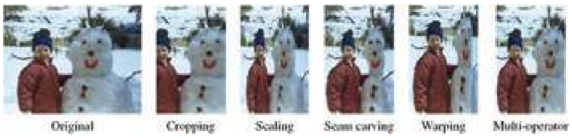


Figure 13: *The output of a number of retargeting methods, including the recent multi-operator. (Image courtesy of Rubinstein et al. [RSA09] ©2009 ACM.)*

4.4. Video-based modelling

Multi-resolution videos allow for interaction with the flat video environment; video panoramas and textures are employed to enhance the perception of spatial and dynamics feature of a scene. A natural step towards video appreciation is their extension to augmented reality and into different forms of virtual reality as in video-based modelling. Environment maps, with their 360° field of view, have been extensively used in crafting virtual reality environments and special effects, however the 2D nature only allows for single resolution display of the scene. The vast amount of optical devices that allow to capture video sequences make videos virtually unlimited resolution means and as such a source for arbitrary resolution photo-realistic imagery. Szeliski [Sze96] concentrated on depth recovery in the process of reconstructing a scene from a video sequence. An image panorama of the video sequence is constructed although the depth information of the depicted scene is recovered through stereographically projecting matching key frames 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

Obtaining a quick overview of a video is an important task in many applications. Whetheranalyzing surveillance videos, wanting a quick overview of a sports match or selecting a movie to watch from a large DVD collection, watching the entire sequence 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 motion flow and depict such information in a way that helps the viewer recognize certain patterns or unusual events in the video. We refer to these techniques collectively as video visualization.

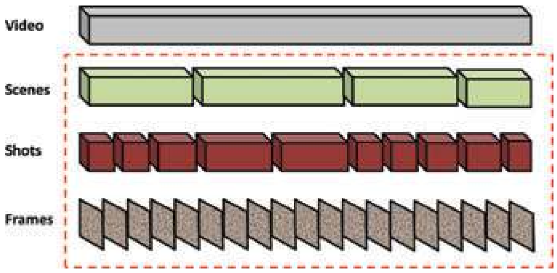


Figure 14: *Segments of a video.*

In this section, we categorize video visualization techniques according to the content and format of the output media. We will base our classification on the 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 key frame storyboards. We consider first the techniques for key frame selection, and then a collection of methods that enhance key frame-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. Key frame 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 14. Each video consists of a sequence of images, or *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 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 a 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 while others 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 key frames, i.e. representative frames of the video. In the following, we will first have a look at the different key frame selection techniques, continue with different depiction methods and finish with a number of techniques that incorporate additional information into key frames to enhance understanding. As mentioned before, key frame selection is typically the first step in image-based video visualization. Key frame 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 representative image for each shot'. As in most optimization procedures, two different strategies can be pursued when choosing relevant images. 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 key frames are to be displayed on a single page or transmitted to a mobile device at a low transmission rate. The error rate approach is applied when looking for the best set of images meeting the optimality criterion. In both techniques, manipulating one parameter affects the other. Commonly, the number of key frames and the error rate are correlated, i.e. if we allow a larger number of key frames to be selected the error will drop and if we increase the allowed error in the second technique, we will obtain more images. Hence, when choosing a strategy, we have to decide what is more important: a fixed number of images or a limit on the error.

No matter which technique we choose, in both cases an optimality criterion has to be defined. The simplest one 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 refer to [TV07].

- *Sufficient content change*: Choose key frames such that they mutually represent different visual content. With the error criterion, we sequentially go through the video and select a frame as key frame whenever it largely differs from the previous key frames. Alternatively, we can look for the n frames that represent sequences of equal variance.
- *Maximum frame coverage*: Select key frames such that they represent a maximum number of frames that are not key frames.
- *Feature space analysis*: Treat each frame as a point in high-dimensional feature space. One optimization strategy is based on point clustering, where the key frames 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 key frames 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 key frames with high information content. They might analyse motion patterns, look for faces or high spatial complexity.

5.2. Another video or an animation

In this subsection, we consider a group of techniques that alleviate the problem of watching videos without leaving the video output domain. There are 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 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 compression is achieved by the cost of information loss. However, the available parts maintain spatial context information.

5.2.1. *Video navigation*

Many proposals have been made regarding the problem of watching videos in a time-saving and efficient manner. Basic video browser controls include *Play, Pause, Fast-Forward, Seek, Skip-to-beginning* and *Skip-to-end of video*. Li *et al.* [LGS*00] add enhanced controls. The most important features include the support for modifying the playback speed between 50 % and 250 % of the original speed while preserving the pitch of the audio, an automatical 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 [RB03] focused on controlling videos with pressure-sensitive digitizer tablets. Beside fading in and out annotations and several interaction possibilities, they present a variation of the fish-eye view called *Twist Lens* to seek in video streams. The timeline 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 (see Figure 15). The occlusion of the frames in the vicinity of the selected one is decreased and an accurate selection of the time position is feasible.

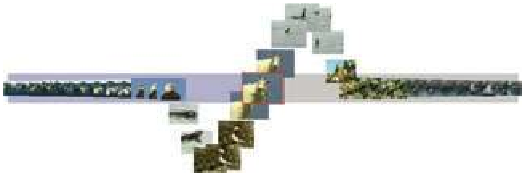


Figure 15: *Twist Lens. (Image courtesy of Ramos et al. [RB03] ©2003 ACM.)*

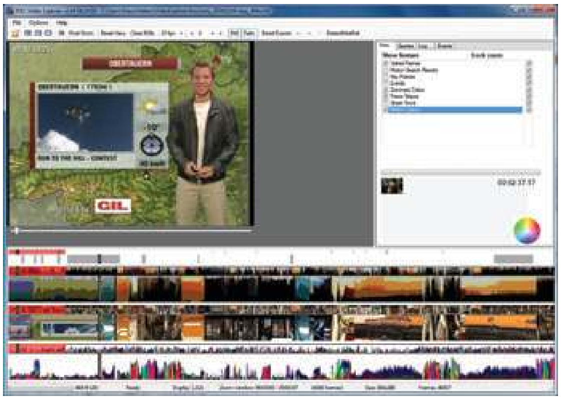


Figure 16: *Video browsing using interactive navigation summaries. (Image courtesy of Schoeffmann et al. [SB09].)*

In [SB09] a timeline 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 timeline at one glance. Navigation summaries can be visited frames, dominant colours, frame stripes or a motion layout (see Figure 16).

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 browse videos 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 (see Figure 17). In [KDG*07] and [GKV*07] scrubbing is also allowed by object manipulation on a floor plan.

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 recommended showing every 64th frame of a video for fast-forward surrogates. Even at lower speeds, the user's abilities in object recognition (graphical), action recognition, linguistic comprehension (full text) and visual comprehension decrease. This problem leads us to different approaches to adapt the video playback speed by video content.

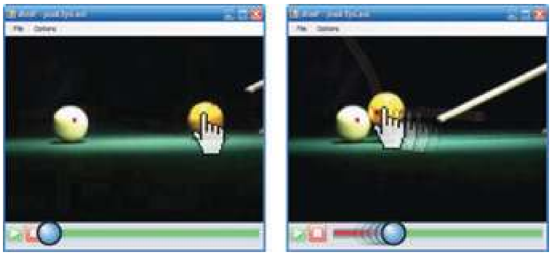


Figure 17: *Video browsing by direct manipulation. (Image courtesy of Dragicevic et al. [DRB*08] ©2008 ACM.)*

Peker *et al.* adapted 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. Höferlin *et al.* [HHWH10] adapted the playback speed according to the temporal information, which allows users to adjust the information load according to their personal abilities, consider static changes and is more robust to video noise than motion.

An adaptive playback speed based on similarity to a target clip is described in [PJH05]. 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 although game interruption scenes (e.g. showing spectators) are highly accelerated.

In [CLCC09] the playback speed is adapted based on three criteria: 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.

In [BBPP10] SLAM techniques are used to generate a 3D reconstruction of the footages captured by typical consumer video cameras. The work is an example of how SLAM techniques can achieve real-time performance in the creation of free viewpoint video transition. The approach heavily relies on colour priors, e.g. foreground objects to have to have specific shape or colours, to avoid artefacts in the reconstruction.

5.2.2. *Video montage and video synopsis*

Kang *et al.* introduced a technique for video abstraction called *video montage* [KCMT06]. They extract visual informative space-time portions from video and merge these parts. Their technique changes the temporal and the spatial occurrence of the information and results in a shorter video clip with condensed information.

One of the method's drawbacks is the loss of spatial context. A method preserving spatial positions was proposed in [RAPP06], [PRAGP07] and[PRAP08]. In their approaches, objects are detected, tracked and temporally 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. The process builds upon what was previously described as *key frame selection* (see Section 5.1).

Truong *et al.* identified a five-step process for automatic 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), *multi-modal integration* (combine skims for different features such as image, audio and text into the final skim) and *assembly* (temporally arrange independent video skim parts, e.g. chronological).

Correa *et al.* [CM10] introduced *Video Narratives* single compositions of dynamic mosaics organized along a linear timeline. The system supports speed varying skimming of videos as well as the generation of storyboard or dynamic video summaries.

The field of video skimming covers a huge research area; we refer to [TV07] for further reading.

5.3. A large collection of images

The easiest direct depiction of key frames is the storyboard technique, where equally sized images are arranged 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 key frames in a hierarchical manner [LSB*00, SKK*01]. 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 they do not provide information about the relevance of individual snapshots. To include such semantics, the images can be scaled according to their importance to the video [YY97, UFGB99]. Yeung and Yeo [YY97], for example, use the number of frames being represented by a key frame, which is equivalent to the subset's length, to scale the key frames 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] presented another approach to video summarization called *Tapestries*, merging the

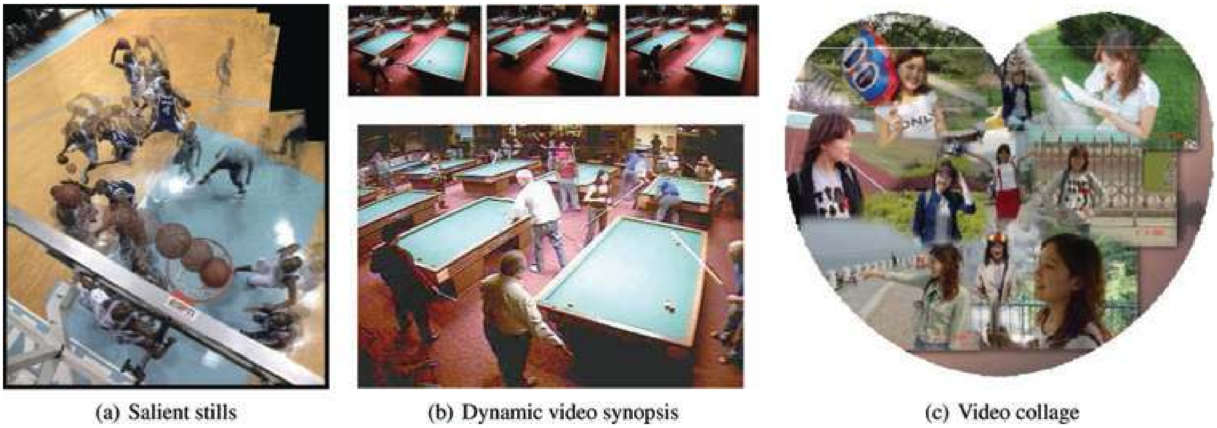

(a) Salient stills (b) Dynamic video synopsis (c) Video collage

Figure 18: *Reassembled depictions of key frames: (a) Salient stills compute the background from a number of frames and add local information about relevant events (image courtesy of Teodosio et al. [TB05] ©2005 ACM). (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)(image courtesy of Pritch et al. [PRAP08] ©2008 IEEE). (c) A video collage rearranges elements of different key frames in a new image of arbitrary shape (image courtesy of Mei et al. [MYYH08] ©2008 SpringerLink).*

structure of DVD chapter menus with the timeline representation of video editing tools.

5.4. A single composite image

All methods in the previous category are common in that they do not alter the contents of the individual key frames. Reassembled depictions, by contrast, 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 (see Section 3.2.1), 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 18a). 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.* [PRAP08] who concentrate on the simultaneous depiction of events happening at different times in the video (Figure 18b).

An alternative approach is taken by techniques that extract relevant subsections of the key frames and reassemble the subimages to form 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 a Voronoi tessellation of the data. This approach was extended in the video collage algorithm [MYYH08] (Figure 18c) and autocollage [RBHB06] where a combination of template-based arrangement and an energy minimization algorithm is used to find good locations for the different subimages. Although the first concentrates on boundaries of arbitrary shape (Figure 18c), the second concentrates on seamless transitions between the different subimages.

5.5. Additional information and actions

In our last category of key frame depictions techniques, we will summarize methods that add additional information to the extracted keyframes.

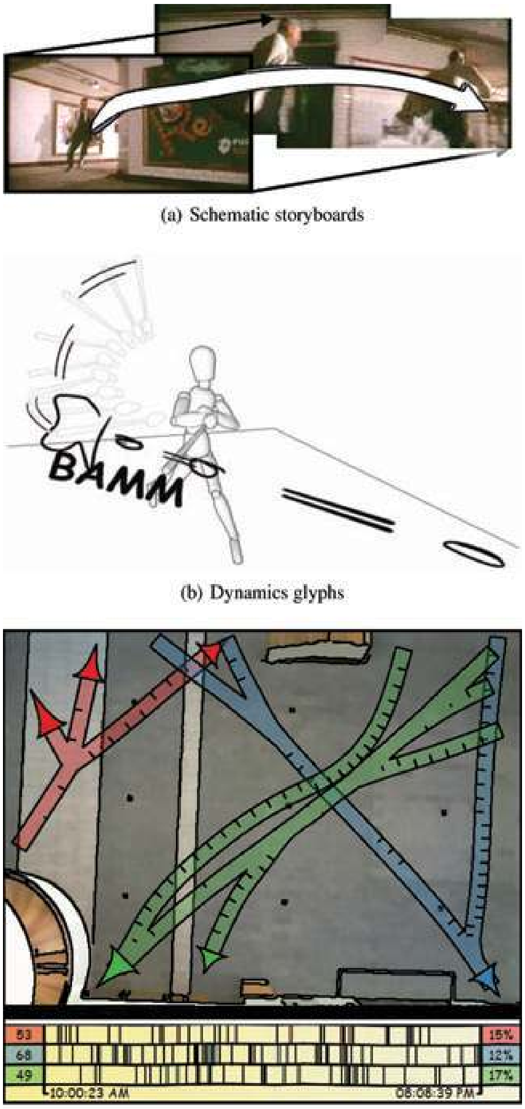
5.5.1. *Enhanced stills*

A well known approach is schematic storyboards (Figure 19a), where annotations are added to illustrate the movement of persons or the camera [GCSS06]. Nienhaus and Dollner [ND05] (Figure 19b) take a similar approach using additional dynamics glyphs. Further, image-based video visualization that enhance the raw data are graph-based approaches that depict, additionally to the key frames, the interaction between different characters or the use of different scenes in a graph [ACCO05].

A hierarchical exploration technique for surveillance videos, the *Interactive Schematic Summaries* (Figure 19c), is introduced by Höferlin *et al.* [HHWH11]. They apply scatter/gather browsing to trajectories for video exploration and use an abstract representation that bundles the trajectories of the clusters.

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



(a) Schematic storyboards

(b) Dynamics glyphs

(c) Interactive Schematic Summaries

Figure 19: *Enhanced stills: (a) Schematic storyboards enhance the displayed key frames with additional information on characters and camera movement (image courtesy of Goldman et al. [GCSS06] ©2006 ACM). (b) Additional dynamics glyphs are used to enhance understanding (image courtesy of Nienhaus et al. [ND05] ©2005 IEEE). (c) Trajectories extracted from video are grouped together and depicted by schematic summaries (image courtesy of Höferlin et al. [HHWH11] ©2011 ACM).*

not display objects in an intuitive manner, but the abstract visual representation can convey temporal attributes more effectively than discrete key frame displays.

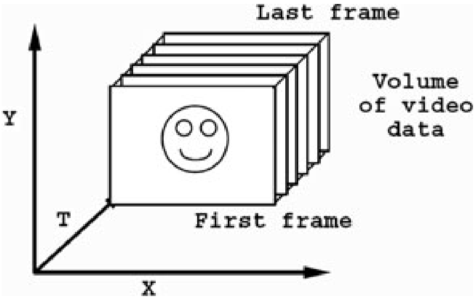


Figure 20: *Interactive Video Cubism. (Image courtesy of Fels et al. [FM99] ©1999 ACM).*

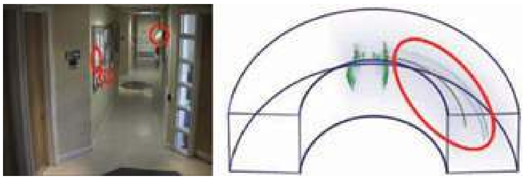


Figure 21: *Video Visualization - changes that remain for a period. (Image courtesy of Daniel et al. [DC03] ©2003 IEEE).*

A popular approach interprets video data as a space-time volume. This idea was first published by Fels and Mase [FM99]. Here, the spatial axes *x* and *y* are combined with time as the third axis (see Figure 20). Within this representation, they define *cut planes* to intersect the video volume. Cut planes can be arbitrarily defined to watch the video in a different way. Normally, watching video in this context is nothing but applying 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 (see Section 3.2.3) or NPR rendering [KSFC02].

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. 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 (see Figure 21), walking with moving arms or an opened door.

Chen *et al.* [CBH*06] introduced *visual signatures* as abstract visual features to depict individual objects and motion events. Therefore, they apply and evaluate flow visualization techniques to video volume visualization. Example visual signatures they used to evaluate their approach are a temporal visual hull, a colour coded difference volume, glyphes and streamlines (see Figure 22, where a sphere moves towards the upright corner of the image frame).

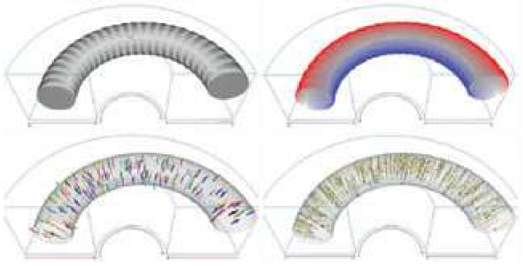


Figure 22: *Visual Signatures in Video Visualization. (Image courtesy of Chen et al. [CBH*06] ©2006 IEEE).*

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.

6. 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 change from one subgoal to another, although new subgoals will surely emerge.
- Video visualization can have applications in many disciplines including science, engineering, sports, medicine and security. However, most of these applications share a common goal: to reduce the time needed for watching videos and to assist the users in gaining insight and making decisions in a cost-effective manner. Different output data types reflect the diversity of means to achieve such a common goal. With the rapid increase of captured video data, there will be a continuous increase in 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- 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 3 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.

Acknowledgements

Authors R. Borgo, M. Chen, E. Grundy and H. Leitte acknowledge the support from UK EPSRC Grant EP/G006555 on video visualization. M. Chen, B. Daubney and J. Xie acknowledge the support from Wales Assembly Government for Research Institute of Visual Computing (RIVIC). G. Heidemann, B. Höferlin, M. Höferlin and D. Weiskopf acknowledge support from the German Research Foundation (DFG) as part of the Priority Program ‘Scalable Visual Analytics’ (SPP 1335).

References

[AA04] Aggarwal M., Ahuja N.: Split aperture imaging for high dynamic range. *International Journal of Computer Vision 58*, 1 (2004), 7–17.

[ACCO05] Assa J., Caspi Y., Cohen-Or D.: Action synopsis: Pose selection and illustration. *ACM Transaction on Graphics 24*, 3 (2005), 667–676.

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

[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* (Washington, DC, USA, 2003), IEEE Computer Society, p. 46.

[AMGC02] Arulampalam S., Maskell S., Gordon N., Clapp T.: A tutorial on particle filters for on-line non-linear/non-gaussian Bayesian tracking. *IEEE Transactions on Signal Processing 50*, 2 (2002), 174–188.

[ARS08] Andriluka M., Roth S., Schiele B.: People-tracking-by-detection and people-detection-by-tracking.

In *IEEE conference on Computer Vision and Pattern Recognition* (2008), pp. 1–8.

[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.

[AS07] Avidan S., Shamir A.: Seam carving for content-aware image resizing. *ACM Transaction on Graphics 26*, 3 (2007), 10:1–10:11.

[ASS*09a] Agarwal S., Snavely N., Simon I., Seitz S., Szeliski R.: Building Rome in a day. In *IEEE International Conference on Computer Vision* (2009), pp. 72–79.

[Ass09b] Assa J.: *Enriching Visual Expressiveness in Medium Transformations* (1st edition). Tel Aviv University, Tel Aviv, Israel, 2009.

[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.

[AZP*05] Agarwala A., Zheng K. C., Pal C., Agrawala M., Cohen M., Curless B., Salesin D., Szeliski R.: Panoramic video textures. *ACM Transaction on Graphics 23*, 11 (2005), 821–827.

[BBPP10] Ballan L., Brostow G. J., Puwein J., Pollefeys M.: Unstructured video-based rendering: Interactive exploration of casually captured videos. *ACM Transaction on Graphics 29* (July 2010), 87:1–87:11.

[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.

[BDH04] Bartoli A., Dalal N., Horaud R.: Motion panoramas. *Computer Animation and Virtual Worlds 15*, 5 (2004), 501–517.

[BETVG08] Bay H., Ess A., Tuytelaars T., van Gool L.: Speeded-up robust features (surf). *Computer Vision and Image Understanding 110* (June 2008), 346–359.

[BEZ09] Buehler P., Everingham M., Zisserman A.: Learning sign language by watching TV (using weakly aligned subtitles). In *IEEE Conference on Computer Vision and Pattern Recognition* (2009), pp. 2961–2968.

[BFB94] Barron J., Fleet D., Beauchemin S.: Performance of optical flow techniques. *International Journal of Computer Vision 12*, 1 (1994), 43–77.

[BFL06] Boykov Y., Funka-Lea G.: Graph cuts and efficient image segmentation. *International Journal of Computer Vision 70*, 2 (2006), 109–131.

[BGSF10] Barnes C., Goldman D. B., Shechtman E., Finkelstein A.: Video tapestries with continuous temporal zoom. *ACM Transaction on Graphics 29*, 4 (2010), 1–9.

[BHH11] Brutzer S., Höferlin B., Heidemann G.: Evaluation of background subtraction techniques for video surveillance. In *IEEE Conference on Computer Vision and Pattern Recognition* (2011), pp. 1937–1944.

[Bis06] Bishop C. M.: *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, Secaucus, NJ, USA, 2006.

[BK75] Brandel C., Kozarski A.: *Warsaw Panorama*. Poland, 1873 [1875].

[BM05] Bennett E. P., Mcmillan L.: Video enhancement using per-pixel virtual exposures. *ACM Transaction on Graphics 24*, 3 (2005), 845–852.

[BNTS07] Bousseau A., Neyret F., Thollot J., Salesin D.: Video watercolourization using bidirectional texture advection. *ACM Transaction on Graphics 26*, 3 (2007), 104:1–104:7.

[BSHK04] Bhat K. S., Seitz S. M., Hodgins J. K., Khosla P. K.: Flow-based video synthesis and editing. *ACM Transaction on Graphics 23*, 3 (2004), 360–363.

[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.

[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.

[BWSS09] Bai X., Wang J., Simons D., Sapiro G.: Video snap cut: Robust video object cutout using localized classifiers. *ACM Transaction on Graphics 28*, 3 (2009), 1–11.

[CAMG06] Caspi Y., Axelrod A., Matsushita Y., Gamliel A.: Dynamic stills and clip trailers. *The Visual Computer 22*, 9 (2006), 642–652.

[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), 1093–1100.

[CGCMC07] Chekhlov D., Gee A. P., Calway A., Mayol-Cuevas W.: Ninja on a plane: Automatic discovery of physical planes for augmented reality using visual slam. In *Proceedings of the Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07)* (2007), pp. 1–4.

[CGL04] Chiu P., Girgensohn A., Liu Q.: Stained-glass visualization for highly condensed video summaries. In *Proceedings of the IEEE International Conference on Multimedia and Expo* (2004), pp. 2059–2062.

[CJSW01] Cheng H., Jiang X., Sun Y., Wang J.: Color image segmentation: Advances and prospects. *Pattern Recognition 34*, 12 (2001), 2259–2281.

[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* (New York, NY, USA, 2009), ACM, pp. 789–798.

[CLCVG08] Cornelis N., Leibe B., Cornelis K., van Gool L.: 3D urban scene modeling integrating recognition and reconstruction. *International Journal of Computer Vision 78* (2008), 121–141.

[CM10] Correa C. D., Ma K.-L.: Dynamic video narratives. *ACM Transactions on Graphics 29*, 3 (2010), 88:1–88:9.

[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.

[CRH05] Collomosse J., Rowntree D., Hall P.: Stroke surfaces: Temporally coherent artistic animations from video. *IEEE Transactions on Visualization and Computer Graphics 11*, 5 (September–October 2005), 540–549.

[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.

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

[DC03] Daniel G., Chen M.: Video visualization. In *IEEE Visualization, 2003* (2003), pp. 409–416.

[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.

[DHH*09] Divvala S. K., Hoiem D., Hays J. H., Efros A. A., Hebert M. H.: An empirical study of context in object detection. In *IEEE Conference on Computer Vision and Pattern Recognition* (2009), pp. 1271–1278.

[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* (New York, NY, USA, 2008), ACM, pp. 237–246.

[DRCB05] Dollar P., Rabaud V., Cottrell G., Belongie S.: Behavior recognition via sparse spatio-temporal features. In *IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance* (2005), pp. 65–72.

[DRMS07] Davison A., Reid I., Molton N., Stasse O.: Monoslam: Real-time single camera slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence 29*, 6 (2007), 1052–1067.

[DSTT00] Dellaert F., Seitz S., Thorpe C., Thrun S.: Structure from motion without correspondence. In *IEEE Conference on Computer Vision and Pattern Recognition* (Los Alamitos, CA, USA, 2000), IEEE Computer Society, vol. 2, pp. 557–564.

[DT05] Dalal N., Triggs B.: Histograms of orientated gradients for human detection. In *IEEE conference on Computer Vision and Pattern Recognition* (San Diego, CA, USA, 2005), IEEE Computer Society, vol. 1, pp. 886–893.

[FCSS09] Furukawa Y., Curless B., Seitz S. M., Szeliski R.: Reconstructing building interiors from images. In *IEEE International Conference on Computer Vision* (2009), pp. 80–87.

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

[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.

[FH06] Felzenszwalb P. F., Huttenlocher D. P.: Efficient belief propagation for early vision. *International Journal of Computer Vision 70* (2006), 41–54.

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

[FJS96] Finkelstein A., Jacobs C. E., Salesin D. H.: Multiresolution video. In *Proceedings of SIGGRAPH 96* (August 1996), pp. 281–290.

[FK98] Faugeras O., Keriven R.: Complete dense stereovision using level set methods. In *European Conference on Computer Vision* (1998), pp. 379–393.

[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 International Conference on Information and Knowledge Management* (New York, NY, USA, 1999), ACM, pp. 78–82.

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

[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.

[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* (Heidelberg, DE, 1998), Springer-Verlag, pp. 11–22.

[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.

[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. 83–90.

[GCSS06] GOLDMAN D. B., CURLESS B., SALESIN D., SEITZ S. M.: Schematic storyboarding for video visualization and editing. *ACM Transaction on Graphics 25*, 3 (2006), 862–871.

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

[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.

[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* (New York, NY, USA, 2007), ACM, pp. 423–432.

[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), pp. 195–204.

[GRG10] GALL J., RAZAVI N., GOOL L. V.: Online adaption of class-specific codebooks for instance tracking. In *British Machine Vision Conference* (2010), 55:1–55:12.

[HCG05] HANNUNA S., CAMPBELL N., GIBSON D.: Identifying quadruped gait in wildlife video. In *IEEE International Conference on Image Processing* (2005), pp. 713–716.

[HCL*09] HU Y., CAO L., LV F., YAN S., GONG Y., HUANG T.: Action detection in complex scenes with spatial and temporal ambiguities. In *IEEE International Conference on Computer Vision* (2009), pp. 128–135.

[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.

[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.

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

[HHWH10] HÖFERLIN B., HÖFERLIN M., WEISKOPF D., HEIDEMANN G.: Information-based adaptive fast-forward for visual surveillance. *Multimedia Tools and Applications 55*, 1 (2010), 1–24.

[HHWH11] HÖFERLIN M., HÖFERLIN B., WEISKOPF D., HEIDEMANN G.: Interactive schematic summaries for exploration of surveillance video. In *Proceedings of the ACM International Conference on Multimedia Retrieval (ICMR'11)* (New York, NY, USA, 2011), ACM, pp. 9:1–9:8.

[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.

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

[HS88] Harris C., Stephens M.: A combined corner and edge detector. *Alvey Vision Conference 4* (1988), 147–151.

[HZ04] Hartley R. I., Zisserman A.: *Multiple View Geometry in Computer Vision* (2nd edition). Cambridge University Press, Cambridge, UK, 2004.

[IA98] Irani M., Anandan P.: Video indexing based on mosaic representations. In *Proceedings of the IEEE* (june 1998), vol. 86:5, pp. 21–28.

[IAH95] Irani M., Anandan P., Hsu S.: Mosaic based representations of video sequences and their applications. In *Proceedings of the Fifth International Conference on Computer Vision* (Los Alamitos, CA, USA, 1995), IEEE Computer Society, p. 605.

[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, pp. 29–32.

[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.

[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, pp. 1331–1338.

[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.

[KGFC02] Klein A. W., Grant T., Finkelstein A., Cohen M. F.: Video mosaics. In *NPAR 2002: Second International Symposium on Non Photorealistic Rendering* (June 2002), pp. 21–28.

[KM07] Klein G., Murray D.: Parallel tracking and mapping for small AR workspaces. In *Proceedings of the Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07)* (2007), pp. 225–234.

[KM09] Klein G., Murray D.: Parallel tracking and mapping on a camera phone. In *Proceedings of the Eigth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'09)* (2009), pp. 83–86.

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

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

[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 22*, 3 (July 2003), 277–286.

[KSFC02] Klein A., Sloan P., Finkelstein A., Cohen M.: Stylized video cubes. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer animation* (New York, NY, USA, 2002), ACM, pp. 15–22.

[KSH05] Ke Y., Sukthankar R., Hebert M.: Efficient visual event detection using volumetric features. In *IEEE International Conference on Computer Vision* (2005), vol. 1, pp. 166–173.

[KST*09] Koutsourakis P., Simon L., Teboul O., Tziritas G., Paragios N.: Single view reconstruction using shape grammars for urban environments. In *IEEE International Conference on Computer Vision* (2009), pp. 1795–1802.

[KTZ05] Kumar M. P., Torr P. H. S., Zisserman A.: Object cut. In *IEEE conference on Computer Vision and Pattern Recognition* (2005), pp. 18–25.

[KUWS03] Kang S. B., Uyttendaele M., Winder S., Szeliski R.: High dynamic range video. *ACM Transaction on Graphics 22*, 3 (2003), 319–325.

[KWLB08] Karrer T., Weiss M., Lee E., Borchers J.: Dragon: A direct manipulation interface for frame-accurate in-scene video navigation. In *Proceedings of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems* (2008), pp. 247–250.

[Lap05] Laptev I.: On space-time interest points. *International Journal of Computer Vision 64*, 2–3 (2005), 107–123.

[Lau94] Laurentini A.: The visual hull concept for silhouette-based image understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligience 16* (1994), 150–162.

[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 (February 1997), 130–145.

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

[LG06a] 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.

[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* (New York, NY, USA, 2000), ACM, pp. 169–176.

[Lin98] LINDEBERG T.: Feature detection with automatic scale selection. *International Journal of Computer Vision 30*, 2 (1998), 79–116.

[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.

[LK81] LUCAS B., KANADE T.: An iterative image registration technique with an application to stereo vision. *International Joint Conference on Artificial Intelligence 3* (1981), 674–679.

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

[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.

[Low04] LOWE D. G.: Distinctive image features from scale-invariant key points. *International Journal of Computer Vision 60*, 2 (2004), 91–110.

[LSB*00] LEE H., SMEATON A. F., BERRUT C., MURPHY N., MARLOW S., O'Connor N. E.: Implementation and analysis of several keyframe-based browsing interfaces to digital video. In *Proceedings of the 4th European Conference on Research and Advanced Technology for Digital Libraries* (Heidelberg, DE, 2000), ECDL '00, Springer-Verlag, pp. 206–218.

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

[LTF*05] LIU C., TORRALBA A., FREEMAN W. T., DURAND F., ADELSON E. H.: Motion magnification. *ACM Transactions on Graphics* (New York, NY, USA, 2005), 519–526.

[LZYN11] LE Q. V., ZOU W. Y., YEUNG S. Y., NG A. Y.: Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *IEEE Conference on Computer Vision and Pattern Recognition* (2011), pp. 3361–3368.

[McI00] MCIVOR A.: Background subtraction techniques. *Image and Vision Computing* (2000).

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

[MGV09] MOONS T., GOOL L. J. V., VERGAUWEN M.: 3D reconstruction from multiple images: Part 1—principles. *Foundations and Trends in Computer Graphics and Vision 4*, 4 (2009), 287–404.

[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.

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

[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. *International Journal on Computer Vision 65*, 1–2 (2005), 43–72.

[MYYH08] MEI T., YANG B., YANG S.-Q., HUA X.-S.: Video collage: Presenting a video sequence using a single image. *The Visual Computer 25*, 1 (2008), 39–51.

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

[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, pp. 2055–2058.

[PDS01] PEKER K., DIVAKARAN A., SUN H.: Constant pace skimming and temporal sub-sampling of video using motion activity. In *Proceedings of IEEE International Conference on Image Processing* (2001), pp. 414–417.

[PESvG09] PELLEGRINI S., ESS A., SCHINDLER K., VAN GOOL L.: You'll never walk alone: Modeling social behavior for multi-target tracking. In *IEEE International Conference on Computer Vision* (2009), pp. 261–268.

[Pic04] PICCARDI M.: Background subtraction techniques: A review. In *IEEE International Conference on Systems, Man and Cybernetics* (2004), vol. 4, pp. 3099–3104.

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

[PRAGP07] Pritch Y., Rav-Acha A., Gutman A., Peleg S.: Webcam synopsis: Peeking around the world. In *Proceedings of ICCV* (2007), pp. 1–8.

[PRAP08] 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.

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

[RAAKR05] Radke R., Andra S., Al Kofahi O., Roysam B.: Image change detection algorithms: A systematic survey. *IEEE Transactions on Image Processing 14*, 3 (2005), 294–307.

[RAKRF08] Rav-Acha A., Kohli P., Rother C., Fitzgibbon A.: Unwrap mosaics: A new representation for video editing. *ACM Transaction on Graphics 27*, 3 (2008), 17:1–17:11.

[RAPLP07] Rav-Acha A., Pritch Y., Lischinski D., Peleg S.: Dynamosaicing: Mosaicing of dynamic scenes. *IEEE Transaction on Pattern Analysis and Machine Intelligence 29*, 10 (2007), 1789–1801.

[RAPP06] Rav-Acha A., Pritch Y., Peleg S.: Making a long video short: Dynamic video synopsis. In *CVPR06* (2006), pp. 435–441.

[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* (New York, NY, USA, 2003), ACM, pp. 105–114.

[RBHB06] Rother C., Bordeaux L., Hamadi Y., Blake A.: Autocollage. *ACM Transaction on Graphics 25*, 3 (2006), 847–852.

[RFA11] Russell C., Fayad J., Agapito L.: Energy based multiple model fitting for non-rigid structure from motion. In *IEEE Conference on Computer Vision and Pattern Recognition* (2011), pp. 3001–3008.

[RFZ07] Ramanan D., Forsyth D., Zisserman A.: Tracking people by learning their appearance. *IEEE Transactions on Pattern Analysis and Machine Intelligence 29*, 1 (2007), 65–81.

[RGSS10] Rubinstein M., Gutierrez D., Sorkine O., Shamir A.: A comparative study of image retargeting. *ACM Transaction on Graphics 29* (December 2010), 160:1–160:10.

[RGSS12] Rubinstein M., Gutierrez D., Sorkine O., Shamir A.: Retarget me a benchmark for image retargeting. http://people.csail.mit.edu/mrub/retargetme/, Accessed on February 2012.

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

[RSA09] Rubinstein M., Shamir A., Avidan S.: Multi-operator media retargeting. *ACM Transactions on Graphics 28*, 3 (2009), 23:1–23:11.

[RV05] Romdhani S., Vetter T.: Estimating 3D shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. In *IEEE Conference on Computer Vision and Pattern Recognition* (2005), vol. 2, pp. 986–993.

[RWY90] Reisfeld D., Wolfson H., Yeshurun Y.: Detection of interest points using symmetry. In *IEEE International Conference on Computer Vision* (1990), 62–65.

[SAB*07] Stoykova E., Alatan A. A., Benzie P., Grammalidis N., Malassiotis S., Ostermann J., Piekh S., Sainov V., Theobalt C., Thevar T., Zabulis X.: 3D time-varying scene capture technologies—A survey. *IEEE Transactions on Circuits and Systems for Video Technology 17*, 11 (2007), 1568–1586.

[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.

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

[SCD*06] Seitz S. M., Curless B., Diebel J., Scharstein D., Szeliski R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition* (2006), pp. 519–528.

[SCM02] Sun X., Chen C.-W., Manjunath B. S.: Probabilistic motion parameter models for human activity recognition. *IEEE International Conference on Pattern Recognition 1* (2002), 463–446.

[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), pp. 47–54.

[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.

[Sio07] SION H.: *Quadruped Gait Detection in Low Quality Wildlife Video*. PhD thesis, Univeristy of Bristol, 2007.

[SKK*01] SULL S., KIM J.-R., KIM Y., CHANG H. S., LEE S. U.: Scalable hierarchical video summary and search. In *Storage and Retrieval for Media Databases* (2001), pp. 553–561.

[SLNG07a] SETLUR V., LECHNER T., NIENHAUS M., GOOCH B.: Retargeting images and video for preserving information saliency. *IEEE Computer Graphics and Application 27*, 5 (2007), 80–88.

[SLNG07b] SETLUR V., LECHNER T., NIENHAUS M., GOOCH B.: Retargeting images and video for preserving information saliency. *IEEE Computer Graphics and Application 27*, 5 (2007), 80–88.

[SMN*09] STARCK J., MAKI A., NOBUHARA S., HILTON A., MATSUYAMA T.: The multiple-camera 3D production studio. *IEEE Transactions on Circuits and Systems for Video Technololgy 19* (2009), 856–869.

[SS04] SEZGIN M., SANKUR B.: Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging 13* (2004), 146–168.

[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.

[ST94] SHI J., TOMASI C.: Good features to track. In *IEEE Conference on Computer Vision and Pattern Recognition* (1994), pp. 593–600.

[Ste04] STEIN F.: Efficient computation of optical flow using the census transform. *Lecture Notes in Computer Science 3175* (2004), 79–86.

[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.

[Sze96] SZELISKI R.: Video mosaics for virtual environments. *IEEE Computer Graphics and Applications 16* (1996), pp. 22–30.

[Sze11] SZELISKI R.: *Computer Vision: Algorithms and Applications* (1st edition). Springer, 2011.

[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.

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

[TB05] TEODOSIO L., BENDER W.: Salient stills. *ACM Transaction on Multimedia, Computing, Communications and Applications 1*, 1 (2005), 16–36.

[TV07] TRUONG B., VENKATESH S.: Video abstraction: A systematic review and classification. *ACM Transactions on Multimedia Computing, Communications and Applications 3*, 1 (2007), 1–37.

[TYAB01] TORRESANI L., YANG D., ALEXANDER E., BREGLER C.: Tracking and modeling non-rigid objects with rank constraints. In *IEEE Conference on Computer Vision and Pattern Recognition* (2001), vol. 1, pp. 493–500.

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

[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.

[VBMP08] VLASIC D., BARAN I., MATUSIK W., Popović J.: Articulated mesh animation from multi-view silhouettes. *ACM Transaction on Graphics 27*, 3 (2008), 1–9.

[VJ01] VIOLA P., JONES M. J.: *Robust Real-Time Object Detection*. Tech. Rep. CRL 2001/01, Cambridge Research Laboratory 2, 2001.

[VMVPVG02] VAN Meerbergen G., VERGAUWEN M., POLLEFEYS M., Van Gool L.: A hierarchical symmetric stereo algorithm using dynamic programming. *International Journal of Computer Vision 47* (2002), 275–285.

[VTC05] VOGIATZIS G., TORR P., CIPOLLA R.: Multi-view stereo via volumetric graph-cuts. In *IEEE Conference on Computer Vision and Pattern Recognition* (2005), vol. 2, pp. 391–398.

[WB95] WELCH G., BISHOP G.: *An introduction to the Kalman filter*. Tech. rep., University of North Carolina at Chapel Hill, 1995.

[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.

[WBBP06] WEICKERT J., BRUHN A., BROX T., PAPENBERG N.: A survey on variational optic flow methods for small displacements. *Mathematics in Industry 10* (2006), 103–136.

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

[WC07] WANG J., COHEN M. F.: Image and video matting: A survey. *Foundations and Trends in Computer Graphics and Vision 3*, 2 (2007), 97–175.

[WDC*08a] WANG O., DAVIS J., CHUANG E., RICKARD I., DE MESA K., DAVE C.: Video relighting using infrared illumination. *Computer Graphics Forum 27*, 2 (2008), 271–279.

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

[WHSL11] WANG Y.-S., HSIAO J.-H., SORKINE O., LEE T.-Y.: Scalable and coherent video resizing with per-frame optimization. *ACM Transaction on Graphics 30* (August 2011), 88:1–88:8.

[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* (Washington, DC, USA, 2003), IEEE Computer Society, pp. 221–230.

[WOG06] Winnemöller H., OLSEN S. C., GOOCH B.: Real-time video abstraction. *ACM Transaction on Graphics 25*, 3 (2006), 1221–1226.

[WRL*04] WANG J., REINDERS M. J. T., LAGENDIJK R. L., LINDENBERG J., KANKANHALLI M. S.: Video content representation on tiny devices. *IEEE International Conference on Multimedia and Expo (ICME'04) 3* (2004), 1711–1714.

[WXSC04] WANG J., XU Y., SHUM H.-Y., COHEN M. F.: Video tooning. *ACM Transaction on Graphics 23*, 3 (2004), 574–583.

[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. 1–6.

[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 (December 2008), 117:1–117:8.

[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 (October 1997), 771–785.

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

[ZM97] ZHU S., MA K.: A new diamond search algorithm for fast block matching motion estimation. In *International Conference on Information, Communications and Signal Processing* (1997), vol. 1, pp. 287–290.