Appearance based Key-Shot Selection for a Hand Held Camera

Bram Alefs, Judith Dijk
TNO Defense, Security and Safety, P.O. Box 96864, 2509 JG, Den Haag, Netherlands
bram.alefs@tno.nl

ABSTRACT

Automatic selection of key-shots is an important step for video data processing. Depending on the purpose, key-shot selection provides user feed back on recorded data, storage reduction and viewpoint selection and it can be used for panoramic image stitching and 3D-reconstruction. In particular, investigating scenes of crime or accidental investigations involves large amount of data, containing information on physical arrangement of objects, details on surface geometry and appearances. This paper proposes an efficient method for automatic selection of key-shot, providing onsite feedback on recorded segments and automatic selection of view-points for 3D-reconstruction. It uses appearance based object and scene modeling for a freely moving, hand held camera. The camera motion is determined on two levels, comparing appearances of local image regions and full 3D reconstruction. On the lower level, the 2D-warp between subsequent video frames is used to determine local change of image appearance and derive a set of motion key frames. These key-frames than are used to determine full 3D motion and to reconstruct objects. Furthermore, key-frames are used for fast indexation and detection of loop closures. Examples for automatic key-frame selection are given for an re-enacted crime scene, and compared to manual selection.

Keywords: Video segmentation, key-frame selection, crime scene investigation

1. INTRODUCTION

Optical recording has become a major tool for investigating scenes of crime and accidentals. In order to achieve full coverage of the scene of investigation (SOI) recorded data need to overview physical arrangement of items, including details on object structure and surfaces. Especially if using a video camera, the amount of recorded data accumulates fast, putting high demand on local storage devices and human effort needed to review data afterwards. Even during recording, the person handling the camera easily loses track on what has been recorded, which typically results in unnecessary repetitions and misses. Computer vision techniques strongly aid the process of recording SOI, for example by (supervised) storage reduction and feedback on recorded scene shot. Such techniques are especially valuable if processed on-site and near real-time.

3D-reconstruction has been an important tool for analyzing SOIs. A basic distinction can be made between laser based (active) techniques and (stereo) vision based passive techniques. Typically, laser based approaches miss the quality for visual interpretation of surface structure and color, but provides more accurate distance measures than camera systems do. Many techniques have been proposes, using laser scanning and computer vision [1][2], covering a wide spread of recording conditions. Computer vision techniques are often designed for reconstruction with indoor construction and turn-table based recording setup. These usually do not work for recordings with a hand held device in outdoor conditions. Camera systems often miss the accuracy for automatic determination of the recording location within the scene.

One particular application of reconstruction is the simultaneous localization and mapping (SLAM) for the recorded scene [3]. These techniques mostly based use cameras positioned on a (robotic) vehicle. Results are difficult to reproduces using a hand held camera, for which motion parameters are less restricted. For crime scenes, vehicles are difficult to navigate and may not guarantee full coverage without human interaction. Using hand-held cameras, only few techniques are proposed for 3D-reconstruction [5, 15]. Mostly, these techniques use matching of feature points to determine ego-motion between subsequent camera view points. Typically, robust determination of the ego-motion is difficult and computational intensive, badly suitable for hand-held computer systems [4, 9]. Furthermore, the choice of key-frames, i.e. the set of perspectives for which feature points are matched is critical for the reconstruction result. Usually, the set of key-frames is selected by hand. Automatic key-frame selection causes errors, especially for long data sequences of hand-held recordings, which contain interruptions and may show a low consistency [6, 7].

Visual Information Processing XVIII, edited by Zia-Ur Rahman, Stephen E. Reichenbach, Mark Allen Neifeld,
Proc. of SPIE Vol. 7341, 734105 · © 2009 SPIE · CCC code: 0277-786X/09/$18 · doi: 10.1117/12.818480

This paper discusses a technique for on-site data processing of key-frames serving two purposes. First, redundant recording and missed items or views are minimized and *key-shots* are selected for purpose of posterior visualization. Second, coverage of view-points and *key-frames* are selected for purpose of 3D-reconstruction of individual objects and reconstruction of physical arrangement of the scene. The techniques are illustrated for recordings of a crime scene, using a hand-held stereo video head as is typically used for efficient recording and 3D-reconstruction purposes.

The purpose of determining key-shot is slightly different to that of determining key-frames, namely, providing an overview of the recorded data. However, techniques determining key-shots largely relate to selection of key-frames as used for video compression [11]. Many key-shot techniques are based on cuts in the subsequent recordings, providing useful results for compression or editing [12]. For SOI, recordings are typically continuously, i.e. without cuts. Segmentation techniques need to be more advanced and, in an ideal case, based on the contents of the recorded scene. Since the latter is difficult to obtain for general purposes, this paper suggests a method for key-shot selection based on the ego-motion of the camera. The proposed method uses a measure for similarity of the object's appearance, in order to segment the video sequence in key-shots for visualization, as well as it provides a set of key-frames for 3D-reconstruction.

The paper is organized as follows. The remainder of this section discusses some background on ego-motion and 3D-reconstruction techniques as apply for video sequence. Section 2 discusses the proposed method for video segmentation, key-shot selection and loop-closure identification. Section 3 shows some results on ego-motion determination and first steps for 3D visualization, for a typical crime scene recording with a hand-held camera and user feedback on the recorded scenes.

1.1 Ego-motion

The proposed measure for key-frame selection uses the ego-motion of the camera as caused by the path and view point direction, at which the scene is recorded. Many papers discuss multi-view analysis using a freely moved hand held camera, moved with the purpose of navigation, scene registration or stitching or creating panoramic images [4, 15]. Restricting the purpose to either one, the geometrical constraints for determining the ego-motion can be simplified to either motion in a preferred direction, specific rotation or assumptions on stationary of the recorded objects [5, 6]. In case of image registration, different techniques are proposed for key-frame selection, such as based on image contents, amount of detected and tracked feature points [8].

Figure 1 illustrates a typical motion pattern for recording SOI with a hand-held camera. The first part of the camera trajectory (concave with respect to a position behind the camera) provides different viewpoints of the same object and the related video segment can be used for 3D-reconstruction (indicated in the lower bar). The second part of the camera trajectory (convex with respect to the space behind the camera) provides a panoramic view of the scene. For this part, the translation is relatively small with respect to its rotation, and subsequent frames show less overlap than in the first case. This segment is less useful for 3D-reconstruction, since it provides fewer view points of the same object. Typically, such ego-motion can be used for creating a panoramic view of the area, e.g. by using image stitching techniques.

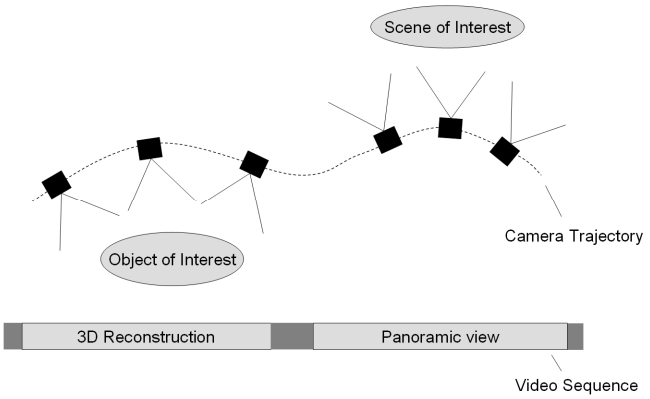

Fig. 1. Example of camera motion while recording a scene of investigation. The lower bar indicates relevant scene shot in the video sequence.

On a mathematical level, ego-motion can be described using a planar projective transformation. For the isotropic case, i.e. ignoring lens distortions, transformation between points {x,y} to {x',y'} in subsequent video frames is given by:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} r\cos\theta & -r\sin\theta & t_x \\ r\sin\theta & r\cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{1}$$

where, $\{t_x, t_y\}$ are the translation coefficients in terms of image coordinates and $\{r, \theta\}$ are parameters related to scaling and rotation (in the image plane), respectively. It can be shown, that ego-motion, as determined by (1) does not suffice for 3D-reconstruction of an object, other than a plane. For full 3D-reconstruction, the epipolar geometry and some calibration parameters, relating the object size to the distance of the camera needs to be known. Let $P = K[R \mid t]$ be a decomposition of the camera projection $\vec{x} = P\vec{X}$, where $\vec{x}$ ={x,y,1} a point in the image, and $\vec{X}$ ={X,Y,Z,1} the corresponding point in world coordinates, K is a 3x3 camera calibration matrix and $[R \mid t]$ is the 3x4 matrix consisting of the rotation matrix R and translation parameters t for the camera motion, respectively. Now, $\vec{x}$ can be expressed in normalized coordinates $\hat{x} = K^{-1}\vec{x}$, and the essential matrix can be solved from the equation:

$$\hat{x}'^{\mathrm{T}} \mathrm{E} \hat{x} = 0 \tag{2}$$

Where, $\hat{x}$ and $\hat{x}'$ are corresponding points in subsequent video frames and E is the essential matrix. The essential matrix mainly differs from the fundamental matrix since it has less degree of freedom and therefore it can be solved more robustly.

1.2 3D-reconstruction

Outdoor registration techniques often base on simultaneous localization and mapping (SLAM) [2][3]. Here, global constraints on ego-motion, such as planarity and camera rotation are used to interpret the most likely camera trajectory, thereby mapping surrounding objects to a single reference system. For hand-held cameras, with fast rotations and changing distance to the objects, global constraints are usually not fulfilled and SLAM only works partly. More local oriented reconstruction techniques, iteratively apply equation (2) for a subsequent set of key-frames, while minimizing drift and error accumulation base on local constraints [15]. For irregular recorded video sequences, like recordings of SOI, such techniques require human interaction on segmentation and key-frame selection. This has two reasons. First, the video sequence needs to be segmented for shots with concave camera motion (as indicated by the first part of the camera trajectory in Fig. 1). Second, key-frames need to be selected out of this segment, for which sufficient corresponding points are found. If only few key-frames are included per segment, the number of corresponding points is low and hence the estimation error for solving equation 2. If many, or all frames are included as key-frame, equation 2 needs to be solved for each (subsequent) pair, resulting in either a large drift (if only subsequent pairs are included) or a large computational complexity (if eq.2 is solved for each frame against all others). In both cases, results are sensitive for mismatches and localization errors of the feature points. In order to reduce such errors, feature points can be found in 3D, e.g. using a stereo camera system. The matching procedure is illustrated in Fig. 2. Each row shows the left and right image of the stereo camera. First, matches are found in between left and right image for each frame set individually (green, cyan dotted horizontal lines). Second, corresponding points are established between properly matched features for subsequent key-frames (magenta vertical lines) [10].

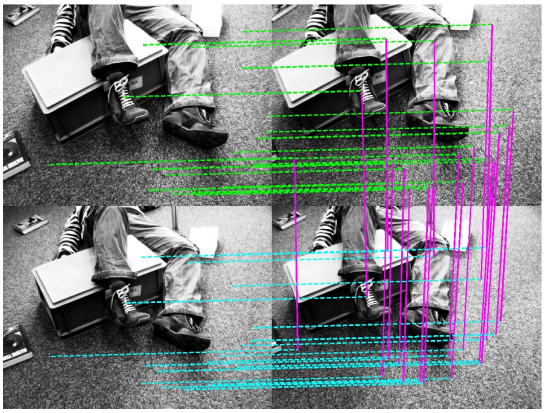

Fig. 2. Matched feature points form left to right in the stereo frame and from the previous stereo frame to the current frame.

Robust video segmentation requires (1) determination of segments that can be used for 3D-reconstruction, while providing necessary key-frames and (2) provides a set of key-shots for each segment to the user while recording. In order to be robust for sequence parts with low structure, as resulted by unintentional recording, motion blur and blending of direct sun-light, camera-motion is approximated by eq. 1, which robustly includes all types of camera motion as depicted by Fig. 1. In a second stage, 3D reconstruction is applied to relevant segments, using a selected set of key-frames and solving eq. 2 for a limited set of key-frame pairs.

2. VIDEO SEGMENTATION

This section describes the method for segmentation and key-shot selection. Video data is processed in three steps. First, the key-shots and key-frames are selected using a similarity measure based on the frame-to-frame motion estimation. Second, video segments are compared and a selection of key-shots is presented to the user. Loop closures are identified, and the user is able to indicate segments for 3D-reconstruction. The last step includes the actual 3D-reconstruction, using the stereo image pair for each selected key-shots. Apart from the last step, algorithms are designed for real-time processing, providing on-site feedback on the recorded sequence.

2.1 Key-frame selection

The process of key-frame selection includes all steps for determining key-shots and video segments. The method is designed for a monocular image sequence, running real-time on a laptop PC. Beside selection of key-frames, key-shots and video segments, it creates a data stack, with local appearance and motion information, which can be used for comparison between segments and detection of loop closures. The method consists of following steps:

(1) Determination of frame-to-frame motion

(2) Determination of the local change of appearance

(3) Segmentation and key-frame selection

Frame-to-frame motion is determined using a 2D-homography between subsequent images. First, a set of feature points are determined for low resolution images, and a correspondence between feature points is established. Second, the set corresponding feature points is used to solve the following motion equation:

$$x' = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} x \tag{3}$$

Where, x and x' are the homogeneous coordinates for a point on the original and projected image R=[a b; -b a], a=$r\cos\theta$, b=$r\cos\theta$, contain the scaling and rotation parameters and where, t=[t_x t_y]^T are the translation parameters, as for eq. 1. As noted in the introduction, eq. 3 describes a 2D-homography, assuming all points to be on a

planar surface. Fig. 3 exemplifies typical transformations obeying eq. 3, showing translation scaling and in-plane rotation. After determining the frame-to-frame motion, a warping can be applied that optimally aligns subsequent images, given the motion constraints. For typical video frame rates, and recording of static objects, constraints of eq. 3 suffice to determine a proper alignment. Fig. 4 shows the results on aligning two subsequent images. The upper panel indicates the current frame, including motion vectors (green lines). The lower left panel shows the image of the previous frame, warped to the current. The centermost lower panel shows a superposition of both previous and current image, without motion correction by 2D warping. The rightmost lower panel shows a superposition after motion correction.

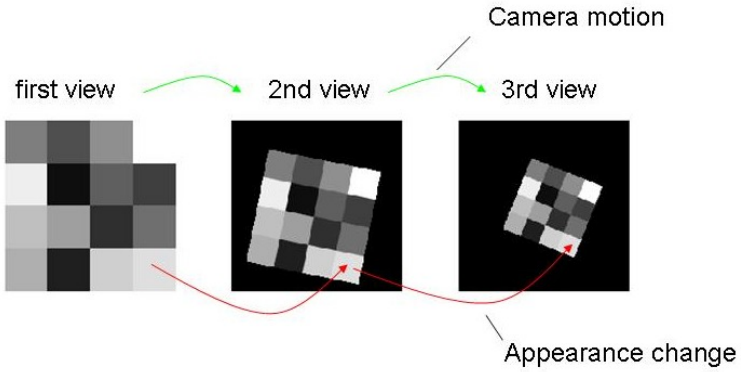

Fig. 3. Visualization of planar transformation (following eq. 1).

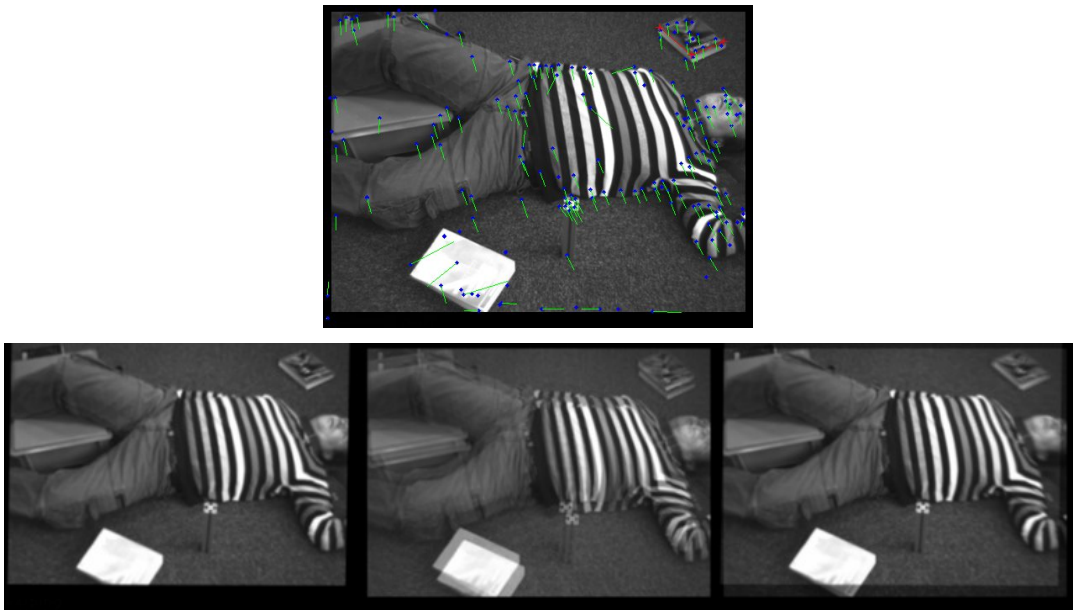

Fig. 4. Example of motion correction based using plane assumption.

Generalizing R of eq. 3 to a non-symmetric 2x2 matrix would allow any affine transformation, i.e. additional freedom for pan and tilt angles of the object plane. In practice, recorded objects are typically non-planar, feature points may be sparse and correspondences can be erroneous. Determining full ego-motion results often in suboptimal solutions causing erroneous results. Using only four degrees of freedom, solutions properly convergence to an acceptable result, even if only few corresponding points are found.

After motion correction, the local image appearance is compared. For this, the image is subdivided in rectangular regions, as indicated by the colored regions in the left panel of Fig.3. For each image region, a local descriptor is determined, resulting in a histogram of descriptor values. Typically, a gray-value histogram suffices for a reasonable

comparison. Given histograms for two subsequent images, for which the first is aligned to the second, as depicted in Fig. 4, the change of appearance can be determined from the histogram intersection:

$$S = \sum_i \arg\min\{H(t)_i, H_P(t-1)_i\} \tag{4}$$

Where, S is a measure for the local similarity $H(t)$ and $H_P(t-1)$ are the histograms for the image at time t and the warped image at time $t-1$, respectively. Since each image is divided in $m \times n$ sub regions, a data stack can be designed with $m \times n \times (t-1)$ entries, where each entry denotes the similarity between frame t and $t-1$ for a given image position. Given the data stack of similarity values and the local motion transformations, time-space segmentation is performed by integrating local similarity vales over time, while correcting for a spatial shift according to the estimated motion.

Space-time segmentation is implemented as follows: for each frame at t, a Gaussian window is determined centered at t and all previous and current values are added within $\{t-\delta, t+\delta\}$. Positions are initially defined by the $m \times n$ grid at time t, but subsequently transformed according to the image motion. Each time step, the grid value is chosen at the nearest position of the $m \times n$ grid, or set to zero if out of view. This way, a $m \times n \times (t-1)$ volume is created, for which each entry depicts the amount of similarity of the depicted object region over time. As to be expected, homogeneous regions show large similarity, since their descriptors do not change with changing view-points. On the other hand, appearance of colorful, non-planar objects changes largely, partly because colorful objects change appearance for different perspective, partly because planar motion correction for non-planar objects causes a larger spatial accuracy. Fig. 5 shows a segmentation example for one frame of the video sequence using an integration width δ=32, a 32x24 sized grid, using gray-value histograms (10FPS recording). The upper panel indicates a 2D projection of the xyt-data stack (columns display data for image with different spatial position, as given by the motion vectors). The lower left panel shows the soft segmentation, as given by the accumulation of similarity values (a bright value indicates large dissimilarity). The lower right panel shows the gray-value image of the actual frame.

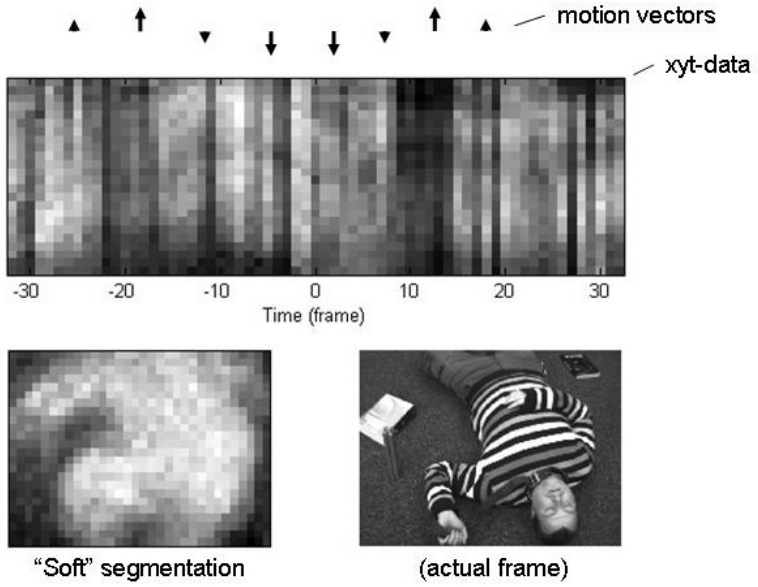

Fig. 5. Segmentation based on local image similarity.

Finally, the video sequence is segmented by analyzing the average value of the accumulated similarity for each frame. For each grid point, the similarity value is accumulated over time. If the accumulated similarity is large, the region has a consistent appearance. If the accumulated similarity is low, the regions changes largely for subsequent frames. Key-shots are defined by analyzing the inconsistency averaged over all grid points. If the on the average, regions are dissimilar, one

or more key shots can be determined. If the average inconsistency is low, regions are similar and the segment can be described by adjacent key-shots. Fig. 6 show an example for key-shot selection. The left panel shows the consistence (upper blue line) for a each frame. Segment boundaries are set at frames that show locally a minimal consistence. Furthermore, by analyzing the smoothness of the peaks in the consistency (lower black line), a set of key-frames (red *) is defined. The right panel shows the similarity matrix of the selected key-frames to all other frames, as derived from the xyt-stack. A red value indicates a large similarity, and red horizontal bars indicate segments with similar contents. A set of intermediate key-frames is determined, by randomly frames within each consistent segment.

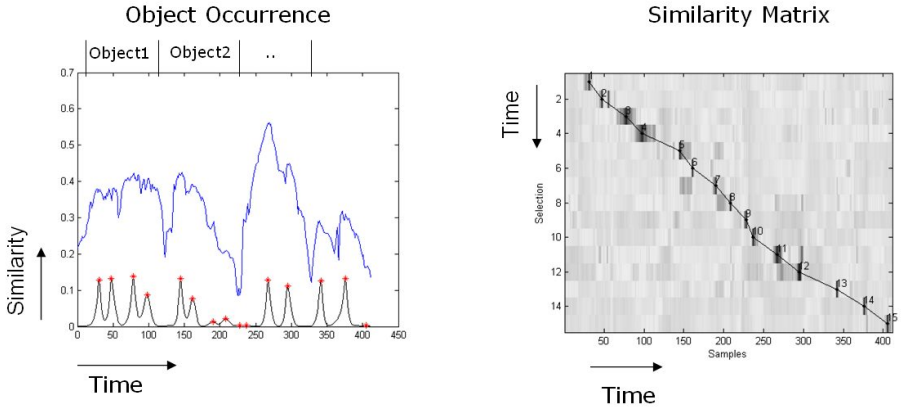

Fig. 6 Left panel: consistency value (blue line), indicating object presence and key-frames. Right panel: similarity matrix, indicating consistent segments (red regions).

2.2 Scene modeling

In order to determine re-occurrences of objects and perspectives, a descriptor is determined that characterizes the local scene appearance. This descriptor is used to determine a match to descriptors of all other video fragments. In order to be computational efficient, the descriptor needs to be compact, while summarizing characteristics for preferable several frames [15]. Changes of perspective are crucial for changes of the local image scene and correcting for ego-motion, as by using the similarity measure between warped frames, would eliminate crucial scene changes. On the other hand, the ego-motion does not depend on the scene, and would not provide a useful criterion itself. Instead, the scene is modeled using global appearance changes. For this, image descriptors are distributed over the field of view and their values are modeled using a sliding window approach

$$H_t = (1-\alpha)D_t + \alpha H_{t-1} \tag{5}$$

Where, H_t is the local model value, D_t is the local image descriptor, such as a color histogram vector and α is the decay constant. The first model is stored as "key model" after a fix time (say ~1s), thereafter only models are stored if the similarity to the last key model is below a certain threshold. These key models can be used for sparse matching between scene candidates.

Fig. 5 shows an example of scene modeling for a vehicle based camera system. The field of view is divided in 6x6 regions and for each region a gray value histogram is extracted with 16 bins. Each bin value is modeled, as for the equation above, and models are matched using the histogram intersection. The resulting similarity matrix is established for each 10th frame only. The result is shown in the left panel. Red colored regions indicate a large similarity, as shown along the diagonal. This is due to homogeneity of the scene and the relative slow adaptation rate (α=0.95, 10fps). At frame 70 the vehicle repeats the same round for a while (loop closure) as visible by a high similarity value along a line about parallel to the diagonal. Additional to the loop closure, sporadic hot spots with high similarity can be found. The left panels shows the frames associated with such hot spots. Although the associated building (each left-right pair) are not the same, the associated images are very similar at first glance.

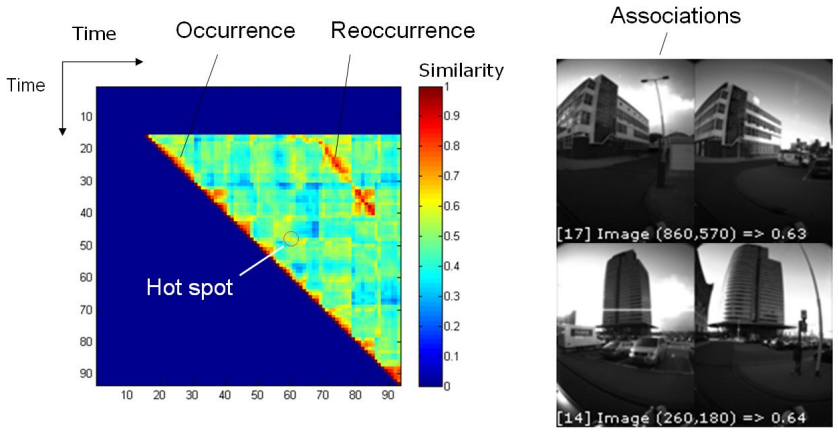

Fig. 7 Loop closure detection and scene association for a vehicle based camera.

2.3 3D Reconstruction

Finally, a 3D reconstruction technique is applied in order to illustrate processing steps following the key-shot selection. First, the motion between subsequent key-frames is determined in following steps [10]:

(1) Stereo matching

(2) Key-frame matching

(3) Robust motion estimation

Step (1) matches feature points between left and right image of the stereo pair. The disparity is determined for each pair of corresponding points, and the right image point coordinates {x,y,d} are converted to a point in 3D, {X,Y,Z} using camera calibration parameters, where lower case {x,y} denotes the image position and d denotes the disparity between left and right image in pixels. Step (2) determines the ego-motion between subsequent key-frames determining (i) correspondences of feature points for both right images, and (ii) solving the essential matrix as described by eq. 2. The matching scheme is visualized in Fig. 2. Since, from step (2) the essential matrix is known, the camera projections {P, P'} can be derived that relate image points $\{\vec{x}, \vec{x}'\}$ to corresponding point in the 3D world, $\vec{X} = P^{-1}\vec{x} = P'^{-1}\vec{x}'$. However, the Euclidian motion can directly be determined using the 3D points as follows from step (1), using following

$$\vec{X}' = R\vec{X} + t \tag{6}$$

Where, R is the rotation matrix and t is the translation vector, and $\{\vec{X}, \vec{X}'\}$ are the 3D point positions as follows from the stereo correspondence for the subsequent key-frames. Using eq. 5, determination of the essential matrix in step (2) is redundant for solving the geometry, however it strongly helps to pre-select the set of corresponding points $\{\vec{X}, \vec{X}'\}$. First, step (2) uses Ransac to get a reasonable set of inliers for solving the essential matrix. Second, step (3) provides robust motion estimation by using 3D positions, as follows from step (1) but only for inliers as follow from step (2).

After determination of the ego-motion, using a sparse set of 3D points of all key-frames, a 3D point cloud is determined based on dense disparity matching. Following steps are performed:

(4) Dense stereo matching

(5) Point cloud mapping

(6) Surface reconstruction

For each key frame, the disparity map is determined using a real-time stereo matching algorithm [13]. For each valid entry in the disparity map, a 3D point is calculated, forming a point cloud in the coordinate system of the current key-frame. Point clouds of subsequent key-frames are mapped to each other using eq. 5 and the ensemble of point clouds is

used for surface reconstruction. For longer sequences, especially if using dense key-frame sampling and high resolution images, the point cloud ensemble becomes huge (tens of millions) and impractical to handle. In order to control the point cloud complexity, a voxelization is used that, at least locally, reduces the point cloud complexity without loss of detail.

The voxelization is done in 3 steps. (i) A cuboid is defined in front of the current camera's view, including most of the image field of view and a range that coincide with most disparity values of the depth map, and subdivided in voxels. (ii) Each point of the point cloud is assigned to one voxel and the points within one voxel are merged. (iii) For a next key-frame, the remaining point cloud is transformed, using eq.5, and additional points are merged for all voxels within the cuboid of the actual frame. Fig. 7 shows accumulation of surface points after processing few key-frames of about the same object. The upper panel indicates the point cloud in three dimensions (points are colored with their original intensity value). The red box indicates the cuboid, in which points are merged (units in mm). The lower panels show the left and right image of the actual key-frame, respectively.

In order to reduce merging errors, points within one voxel are clustered according to color and surface normal (as derived from the disparity image), and several points are propagated within one voxel in case of ambiguities. Finally, Poisson surface reconstruction is applied to the entire point cloud, for purpose of visualization [14].

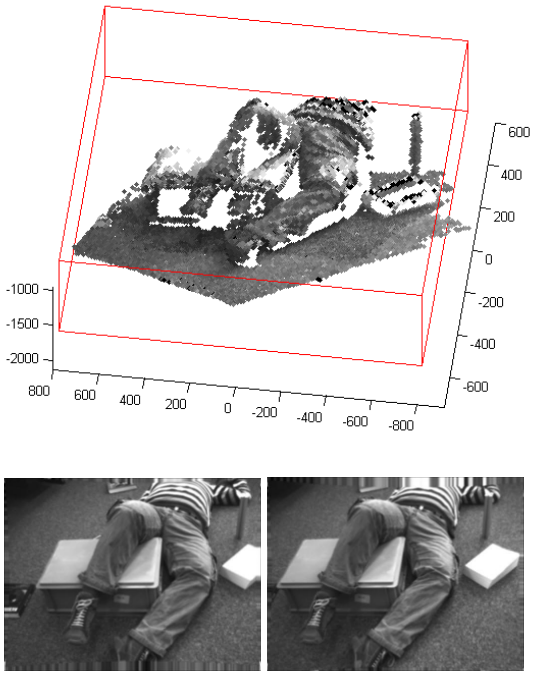

Fig. 8 Reduction of point cloud complexity by voxelization.

3. CRIME SCENE ANALYSIS

The proposed method is applied optical recordings of crime scenes as used for police training. In order to illustrate the effects on real-life scenes, a re-enactment was recorded with typical crime scene conditions. The resulting sequence was segmented manually, resulting in a set of 25 segments of dominated objects. Fig. 9 shows the shots for different segments as identified by human observation. Segments differ in length, depending on the intention of recording. Some details become particular attention, like the stick in upper left and lower right image, face and the badge. Since the camera comes closer for these objects, and the segments contains many frame, these shots specially suits 3D-reconstruction. Intermediate segments are mainly used to establish a spatial arrangement between different objects of interest. Since the distance to the object is relatively small in all cases, this can be done by full 3D-reconstruction. If objects of interests are further apart, this relation may only be established coarsely or only based on 2D analysis (as for the second part of the camera trajectory shown in Fig. 1).

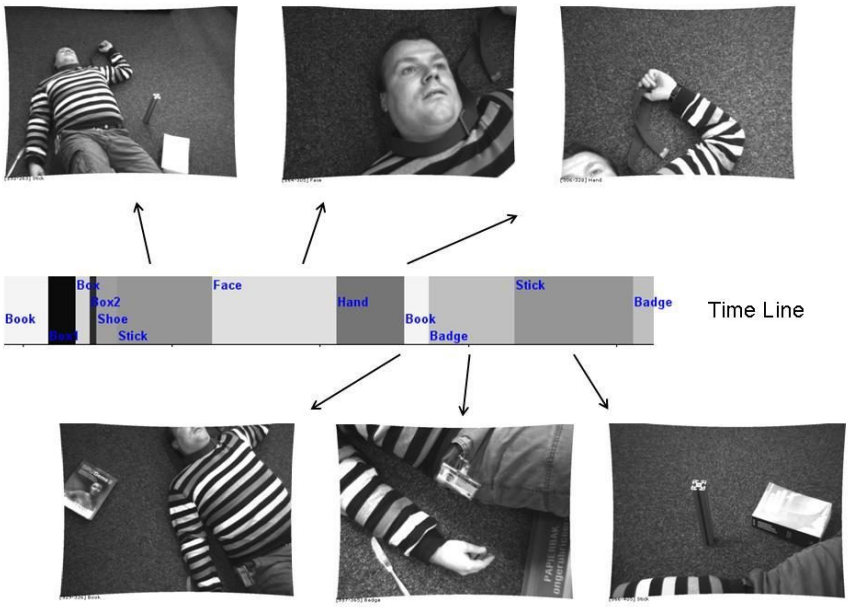

Fig. 9 Manual annotation results for a re-enacted crime scene.

3.1 Key-frame selection

The key-frames were selected for the first 430 frames of the re-enacted crime scene recording. Images were downscaled to 512x385pixels, using frames of the left camera only. Totally, 15 segments were identified, including 19 key-shots and 21 additional key-frames. Table 1 shows a comparison with the manual annotated sequence, and a key-frame selection by a computer vision expert. Although the number of segments and key-frames is smaller, the selection included a segment for each major shot in the scene. Few transitional segments were omitted by the algorithm (duration 3-5 frames), but selected during manual annotation. The number of selected key-frames is smaller than that of the expert annotation. This is partly caused by the missing of the short segments, partly, because manual reconstruction only uses a subset of the key-frames for creating a 3D-point cloud. If all key-frames of the manual segmentation would be used, the point cloud becomes extremely large and difficult to handle for outliers. This second selection procedure is based on feedback of the reconstruction results. For automatic selection, this user feedback is not given and all automatic selected key-frames are used in order to provide a complete reconstruction result. The point cloud complexity is controlled using the voxelization as discussed above.

Comparing the full list of manually selected key frames (as used for motion determination), the number of key-frames largely coincides for the larger segments. Automatic selection puts little more emphasis for frames at which the distance to the object changes (zooming) and in-place camera rotations. Fig. 10 shows a series of key-shots as used for 3D-reconstruction. Although the perspective and/or the distance to the object differ between subsequent key-frames, central positioned objects re-occur in the subsequent frames. This enables determination of 3D ego-motion by key-point matching

Table 1. Segmention result.

	Automatic selection	Manual selection
#Segments	15	25
#Key Frames	40	79
#Reconstructions	1	1

Fig. 10 Automatically selected key-frames.

3.2 3D-reconstruction

This section illustrates further processing steps for 3D-reconstruction. Since the main contribution of this paper is the pre-processing, effects on post-processing, outlier reduction, surface reconstruction and 3D rendering were left out of the analysis. Results are presented as 3D-point cloud, for which each point consists of a dot, colored with the original image colors. Following steps were performed to all 40 key-frames provided for the sequence as given by table 1: ego-motion determination from SIFT features, dense disparity calculation and point cloud voxelization. Totally 1M points were visualized by 3D projection, as shown by Fig. 11. Apart from some occasional outliers, and incomplete surfaces (book, box) the ego-motion was determined properly, providing one plane for all pixels of the ground plane.

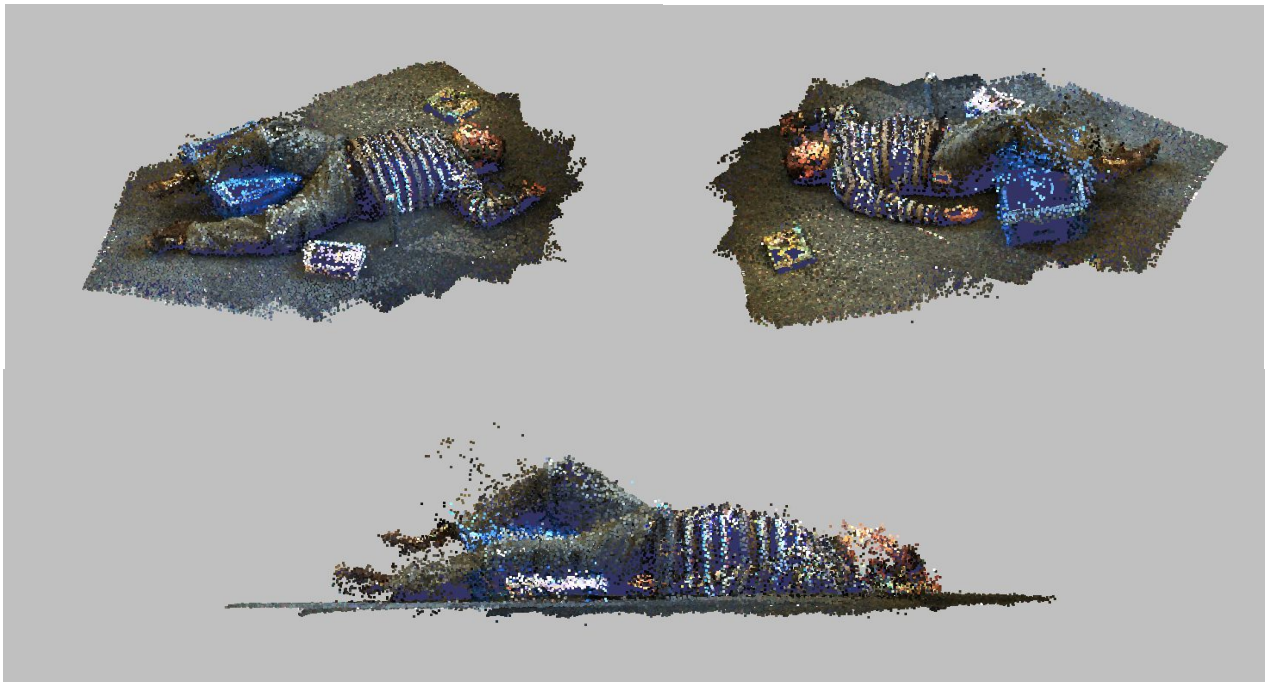

Fig. 11 Left: point cloud ensemble, colored with image value. Right: 3D-surface reconstruction

CONCLUSIONS

Automatic detection of key-shots is an important step for unsupervised 2D and 3D video processing. A method is proposed that provides a key-shot selection for recording with a hand-held camera, as can be used for user feedback and 3D scene reconstruction. The underlying technique is based on a measure for the local image similarity, under constraint of planar motion for subsequently recorded frames. Based on this measure, key frames are selected, representing video segments with largely consistent contents and small changing of appearance. The key-shot selection is used to provide on-site feedback, indicating repetitive recording to the user and 3D reconstruction. The similarity measure provides a dynamic scene model, as can be used for association and loop closure detection. Selected key-frames are compared to manual key-frame selection for a re-enacted crime scene. Results show that most important segments are presented by key-frames, enabling a proper determination of the full 3D ego-motion.

A 3D reconstruction is illustrated from the point cloud, as result from dense disparity matching and ego-motion of all the key-frames. Although automatic selection uses more input frames than for manual 3D reconstruction, the number of points can be controlled, even if 3D points of all key-frames are superposed. Further steps include smoothing and outlier reduction of the point cloud, enabling unsupervised surface reconstruction, from hand-held camera recordings.

REFERENCES

[1] Agrawal, M., K. Konolige, K. Bolles, R.C., Localization and Mapping for Autonomous Navigation in Outdoor Terrains : A Stereo Vision Approach, Applications of Computer Vision, 2007. WACV '07.

[2] Davison A.J., Real-Time Simultaneous Localisation and Mapping with a Single Camera, ICCV 2003.

[3] Newman et al, Outdoor SLAM using Visual Appearance and Laser Ranging, ICRA 2006.

[4] Hansen et al., Scale Invariant Feature Matching with Wide Angle Images, Int.Conf. on Intelligent Robots and Systems 2007.

[5] Thormaehlen et al. Keyframe Selection for Camera Motion and Structure Estimation from Multiple Views, ECCV 2004.

[6] Repko, Pollefeys, 3D Models from Extended Uncalibrated Video Sequences: Addressing Key-frame Selection and Projective Drift, Int. Conf. on 3-D Digital Imaging and Modeling 2005

[7] Adam, Rivlin, Shimshoni, Robust Fragments-based Tracking using the Integral Histogram, CVPR 2006.

[8] Liu et al. Shot reconstruction degree: a novel criterion for key frame selection, Pattern Recognition Letters, Volume 25, Issue 12, September 2004, Pages 1451-1457.

[9] Hartley, Zisserman *Multiple view geometry in computer vision*, Cambridge University Press, 2000.

[10] W. van der Mark et al. "3D Scene Reconstruction with a Handheld Stereo Camera", Proc. Cognitive Systems with interactive Sensors, COGIS 2007.

[11] X. Song, G. Fan, Key-frame extraction for object-based video segmentation, ICASSP 2005.

[12] J. Calic, E. Izquerdo, Efficient key-frame extraction and video analysis, Information Technology: Coding and Computing, 2002,

[13] W. Mark, D. Gavrila, Real-Time Dense Stereo for Intelligent Vehicles, Trans. Intel. Transporatation Systems, Vol. 7/1, 2006

[14] M. Kazhdan, M. Bolitho, and H. Hoppe, Poisson Surface Reconstruction, Symposium on Geometry Processing 2006.

[15] L.A. Clemente et al. Mapping Large Loops with a Single and-Held Camera. RSS 2007.