International Journal of Computer Science and Engineering Research and Development (IJCSERD), ISSN 2248-9363 (Print), ISSN 2248-9371 (Online)
Volume 1, Number 1, April-June (2011)
pp. 15- 27 © PRJ Publication
http://www.prjpublication.com/IJCSERD.asp

IJCSERD

© PRJ PUBLICATION

ROBUST AND GENERIC APPROACHES FOR VIDEO SEGMENTATION

K. Mahesh
Associate Professor, Department of Computer Science and Engg
Alagappa University, Karaikudi,TN,India.
mahesh.alagappa@gmail.com

Dr.K. Kuppusamy
Associate Professor, Department of Computer Science and Engg
Alagappa University, Karaikudi,TN,India.
kkdisamy@yahoo.com

P. Pandi selvi
Department of Computer Science and Engineering
Alagappa University, Karaikudi,TN, India.
selvikrish.selvi@gmail.com

ABSTRACT

The survey describes an approach for object –oriented video segmentation based on motion coherence. Using a tracking process ,2-D motion patterns are identified with an ensemble clustering approach. Particles are clustered to obtain a pixel-wise segmentation in space and time domains. The limitation of the segmentation method concerning with complex 3-D spatial motion can be solved by using reality-based 3-D models. The reality-based 3-D models are produced with range-based, image-based or CAD modeling techniques. Each 3-D model can contain different levels of geometry and need therefore to be semantically segmented and organized in different ways.

Keywords: Ensemble clustering, motion segmentation, object-based video segmentation, point tracking, video coding, reality-based 3-D models.

1. INTRODUCTION

MOTION segmentation is an important preprocessing step in many computer vision and video processing tasks, such as surveillance, object tracking, video coding, information retrieval, and video analysis. These applications motivated the development of several 2-D motion segmentation techniques, where each frame of a video sequence is split into regions that move coherently. By 2-D motion, here we denote the motion of objects in a 3-D scene projected in the image plane of a camera. However, 2-D motion segmentation often leads to video over-segmentation. For example, a scene composed by static rigid objects, captured by a moving camera, can be over-segmented in several 2-D motion regions due to several reasons, such as depth discontinuities, occlusions, and the perspective projection effect. In some situations, a scene comprises several moving objects and it is necessary to identify each object as a coherent motion entity. In these cases, the segmentation process can be approached by considering the objects as moving in 3-D space , and this approach has motivated several works in 3-D motion segmentation [1]–[6]. Spatio-temporal motion segmentation [7]–[10] is a different approach, where different moving objects are segmented in volumes (called tunnels [10]) in the domain formed by the

spatial dimensions (e.g.,) and the temporal dimension , and these volumes are delimited by object motion boundaries (i.e., motion discontinuities).
The definition of object in a video segmentation framework is related to the concept of region homogeneity, and different applications require different region homogeneity criteria. In video coding, for example, segmentation is frequently used to explore the data redundancy in time [11]. In this context, an object region that retains its characteristics (e.g., color or texture) along the sequence can be considered homogeneous and redundant. Thus, even if the object region moves along the temporal sequence, the region representation remains the same, i.e., redundant, within the object motion boundaries.

In 3-D motion segmentation, the concept of object is related to actual objects existing in 3-D space that do not change their 3-D characteristics over time. The concept of object in spatio-temporal segmentation is different, since an object is represented by a spatio-temporal tunnel formed by a sequence of 2-D projections, each 2-D projection obtained at a time of an object in 3-D space. Two sets of parameters are often used to describe 3-D parametric motion, the set of global parameters representing the camera and/or object motion, and the set of local parameters representing the object attributes (e.g., shape, color, and texture). However, the estimation of a large number of parameters often is awkward, particularly in the presence of noise and outliers. An outlier can be, for example, a point trajectory incorrectly computed. On the other hand, when camera translations and depth variations are small compared to the distance of the camera to the scene objects, simpler 2-D motion models can become attractive [12]. In 2-D parametric motion segmentation, a small number of parameters is needed to describe the object motion, making the motion segmentation more robust to noise. Although the computer vision community has been consistently working towards improving 3-D motion segmentation [13]–[20], 2-D motion segmentation also has received attention, since it still has some open issues and it is suitable for some important video processing tasks like video coding, where a simple representation is important, and in general the semantic aspects of the scene are less relevant [11].

In the context of motion estimation, the literature can be divided in two classes of methods: direct methods [12] and feature-based methods [21]. Motion segmentation methods can also be divided according to these two paradigms. Direct methods recover the unknown parameters directly from measurable image quantities at each pixel in the image, solving two problems simultaneously: 1) the motion of the camera and/or objects of the scene, and 2) the correspondence of every pixel. This is in contrast with the feature-based methods, which first extract a sparse set of distinct features from each image separately, and then recover and analyze their correspondences in order to determine motion. Feature-based methods minimize an error measure that is based on distances between a few corresponding features, while direct methods minimize a global error measure that is based on direct image information collected from all pixels in the image. For this reason, direct methods are sometimes called dense methods in the literature.
However, we should note that, according to the feature-based philosophy, motion can be estimated using sparse features in a first step, and in a second step the motion should guide the dense correspondence for the nonfeature pixels. Thus, in this work we refer to any motion estimation / segmentation method that yields correspondence / classification for each pixel as "dense", even if the motion estimation/segmentation core is guided only by a sparse set of features. It is important to observe that with direct methods the pixel correspondence / classification is performed directly with the measurable image quantities at each pixel, while in feature-based methods this is done indirectly, based on independent feature measurements in a set of sparse pixels.

An important property of the direct methods is that they can sucessfully estimate global motion even in the presence of multiple motions and/or outliers [12]. Moreover, normal flows can only be combined across regions of the image that have some simple parametric form (such as an affine or quadratic [21]), and motion estimate errors can accumulate when frames are distant apart and the data may not fit the model very well.

On the other hand, feature-based methods initially ignore areas of low information, resulting in a problem with fewer parameters to be estimated, with good convergence even for long sequences. Further, there is a wide of choice of algorithms to estimate parameters for more complex models (e.g., epipolar or trifocal geometry) from point or line features. Nevertheless, in these methods, feature correspondences are computed independently, being more susceptible to outliers. Variational frameworks have been widely used in the context of direct motion segmentation [7]–[10], [22]–[24]. In order to improve the quality of object segmentation, shape priors were integrated in variational methods by Cremer *et al.* [22]–[24]. However, prior information about the objects in a sequence often is not available. Therefore, Mitiche *et al.* [7] proposed to segment moving objects by detecting the tunnel delimited by motion discontinuities in the spatio-temporal domain. Feghali and Mitiche further extended this idea to also handle moving cameras [8], and later Sekkati and Mitiche proposed a 3-D direct motion segmentation method with a similar approach in [14] and [25]. They formulated the problem as a Bayesian motion partitioning problem, and approached the corresponding Euler-Lagrange equations as a level-set problem. Cremers and Soatto [9] proposed a multiphase level-set method to segment a video using spatio-temporal surfaces (tunnels) that separate regions with piecewise constant motion.

A limitation of segmentation methods based on motion discontinuities(motion boundaries) is that they tend to fail in frames where these boundaries are not evident, or do not exist. For example, a static object in a static background that moves only in the last few frames of the sequence can not be correctly segmented at the beginning of the sequence, since there were no motion boundaries and the object was not moving with respect to the background. Ristivojevic and Konrad [10] also proposed a spatio-temporal segmentation method based on the level-set approach, where they defined the concepts of occlusion volumes (i.e., background regions that become occluded) and exposed volumes (background regions that become visible). The authors suggested that potentially the concepts of occlusion and exposed volumes can be applied in the next-generation of video compression methods. As occlusion and exposed areas are difficult to predict, new efficient compression techniques could be developed by estimating occlusion and exposed areas *a priori.* However, the occlusion volume concept have limitations as proposed, since it does not consider the occlusion of a moving object by the background, or by another moving object. A characteristic shared by most variational methods is that they rely on motion models defined *a priori.* If the data do not fit these models well, the methods tend to fail—except when special conditions can be assumed, like static background, number of objects known *a priori,* etc.

Feature-based methods for motion segmentation usually consist of two independent stages: 1) feature selection and/or correspondence and 2) motion parameter estimation [21]. The second stage often is performed through factorization methods [2], [5], [15], [26], although some simpler clustering strategy can be used [27], [28]. In factorization methods, motion and shape information are treated separately by applying constraints to the scene projection on the image formation plane, as well as on the object shape and motion. Several methods have been proposed for sparse feature selection and/or correspondence, and among the most popular are the Harris Corner Detector [29], KLT [30], and SIFT [31]. As mentioned before, a weak point of these sparse feature-based methods is that feature correspondences are computed independently. Thus, they are very sensitive to outliers, making them susceptible to errors in motion parameter estimation / segmentation. Also, homogeneous regions of a frame may present none or few features, making the motion estimation/segmentation difficult (or even impossible) in large areas of the video frames.

Eventually, we can obtain consistent object segmentation by combining several partial informations about an object. The idea of merging object segmentation information from several parts of a sequence was proposed by Geldon *et al.* [32], as a probabilistic multiple hypothesis tracking (PMHT) approach. The authors propose to track an object over the whole image sequence, by combining partial object segmentations previously computed in different parts of the sequence. This is done by modeling the motion and geometry of the objects, and these models are combined assuming smooth trajectories, and are used to eliminate ambiguities caused by occlusions and incorrect detections. However, the object motion and/or geometry modeling accuracy depends on how well the models fit the data, besides the trajectory constraints preclude the application of this approach in videos with discontinuous object trajectories, which is common in sequences obtained with hand-held cameras, for example.

Here we present a new approach for video object segmentation where objects are defined as nonoverlapping regions (at pixel level) in the spatio-temporal domain. These regions are expected to retain their spatial and photometric characteristics in time. Our approach combines the advantages of dense and feature-based methods, as described next. Initially, correspondences in time of sparse points (i.e., particles) are computed, so that long-range motion patterns can be identified. However, instead of computing point correspondences independently (as done in many feature-based methods), neighboring particles are treated as they were linked, reducing the chance of occurring outliers and avoiding the aperture problem [33].1 Moreover, the density of sampled points (i.e., particles) is adaptive, and denser particle distributions are used in regions where precision is more important (for example, in motion boundaries), saving computation without neglecting homogeneous regions.

To compute particle correspondences in a video sequence, we use the approach proposed by Sand and Teller [34], which relies on particles that are located with sub-pixel precision. After the particle correspondences are computed, particles are clustered in each frame of the sequence. The individual particle clusterings at frame level, are then further grouped in larger sets of particles associated to different frames, according to an ensemble clustering strategy. Finally, a dense video frame representation (i.e., a pixel-wise representation) of the final clustering is obtained.And in the case of 3-D complex modeling in which it has the drawback of over segmentation it can be rectified with the help of Reality-Based 3-D Modeling.

The continuous development of data capture methodologies, multiresolution 3D representations and the improvement of existing ones are contributing significantly to the documentation, conservation and presentation of information and to the growth of research in the field of video processing. This is also driven by the increasing requests and needs for digital documentation at various sites and at different scales and resolutions for successive applications like conservation, restoration, visualization, education, data sharing, 3D GIS, etc. 3D surveying and modeling of scenes or objects should be intended as the entire procedure that starts with the data acquisition, geometric and radiometric data processing, 3D information generation and digital model visualization. A technique is intended as a scientific procedure (e.g. image processing) to accomplish a specific task while a methodology is a combination of techniques and activities joined to achieve a particular task in a better way. Reality-based surveying techniques (e.g. photogrammetry, laser scanning, etc.) [44] employ hardware and software to metrically survey the reality as it is, documenting in 3D the actual visible situation of a site by means of images [45], range-data [46,47], Reality-Based 3D Modeling and Segmentation of the aforementioned techniques [48,49,50]. Non-real approaches are instead based on computer graphics software or procedural modeling [51,52]and they allow the generation of 3D data without any metric survey as input or knowledge of the site.

1.1 Range-Based 3D Reconstruction

Optical range sensors like pulsed (TOF), phase-shift or triangulation-based laser scanners and stripe projection systems have received in the last years a great attention, also from non-experts, for 3D documentation and modeling purposes. These active sensors deliver directly ranges (i.e. distances thus 3D information in form of unstructured point clouds) and are getting quite common in the heritage field, despite their high costs, weight and the usual lack of good texture. During the surveying, the instrument should be placed in different locations or the object needs to be moved in a way that the instrument can see it under different viewpoints. Successively, the 3D raw data needs errors and outliers removal, noise reduction and the registration into a unique reference system to produce a single point cloud of the surveyed scene or object. The registration is generally done in two steps: (i) manual or automatic raw alignment using targets or the data itself and (ii) final global alignment based on Iterative Closest Points (ICP) or Least Squares method procedures. After the global alignment, redundant points should be removed before a surface model is produced and textured. Generally range-based 3D models are very rich of geometric details and contain a large number of polygonal elements, producing problems for further (automated) segmentation procedures.

1.2 Image-Based 3D Reconstruction

Image data require a mathematical formulation to transform the two-dimensional image measurements into three - dimensional information. Generally at least two images are required and 3D data can be derived using perspective or projective geometry formulations. Image-based modeling techniques (mainly photogrammetry and computer vision) are generally preferred in cases of lost objects, monuments or architectures with regular geometric shapes, small objects with free-form shape, low-budget project, mapping applications, deformation analyses, etc. Photogrammetry is the primary technique for the processing of image data. Photogrammetry, starting from some measured image correspondences, is able to deliver at any scale of application metric, accurate and detailed 3D information with estimates of precision and reliability of the unknown parameters. The dense 3D reconstruction step can instead be performed in a fully automated mode with satisfactory results [54,55,56]. But the complete automation in image-based modeling is still an open research's topic, in particular in case of complex heritage scenes and man-made objects [57] although the latest researches reported quite promising results [58].

1.3 CAD-Based 3D Reconstruction

This is the traditional approach and remains the most common method in particular for architectural structures, constituted by simple geometries. These kinds of digital models are generally created using drawings or predefined primitives with 2D orthogonal projections to interactively build volumes. In addition, each volume can be either considered as part of adjacent ones or considered separated from the others by non-visible contact surfaces. Using CAD packages, the information can be arranged in separate regions, each containing different type of particles, which help the successive segmentation phase. The segmentation, organization and naming of CAD models and their related sub-components generally require the user’s intervention to define the geometry and location of subdivision surfaces, as well as to recognize rules derived e.g. from classical orders.

The proposed method is general in the sense that it does not rely on motion models, does not impose trajectory constraints and segments multiple objects of arbitrary shapes, without knowing the number of objects a priori. Instead of motion boundaries, the segmentation is guided by the consistent motion behavior of sample points of the frames. This strategy allows to extract longer tunnels in the spatio-temporal domain. Besides, it does not need any special treatment for changes in topology or new objects that arise along the sequence. The proposed method potentially has the ability to generate occluded and exposed volumes, since motion patterns are discovered and associated to each moving region, and voxels of the spatio-temporal volumes are classified as belonging to object volumes, occluded volumes or exposed volumes. The proposed approach generates a simple scene representation, adequate for object video coding, and also delivers a more redundant and temporally persistent partition of the scene than direct video segmentation methods and motion prediction strategies.

2. METHOD OVERVIEW

The structure of the proposed coherent motion segmentation approach can be divided in three main parts.
2.1) Estimation of Particle Trajectories
2.2) Segmentation of Particle Trajectories
2.3) Dense Segmentation Extraction

2.1 Estimation of Particle Trajectories

It concerns the selection and tracking of a set of points of the scene (namely, particles). This stage takes as input the original video frames, and returns as output a set of particles and their respective trajectories. During the estimation of particle trajectories, the particles whose correspondent point locations in the scene suffer occlusion are eliminated, and new particles are created in regions that become newly visible along the video sequence.

2.2 Segmentation of Particle Trajectories

It deals with the segmentation of particle trajectories, so that particles moving coherently are grouped together. This stage takes as input the particles trajectories computed in the first stage, and returns labels for all the particles as outputs, representing the motion segmentation of frame regions according to the particle trajectories. The segmentation of particle trajectories can be divided in four steps.

• Clustering of 2-frame motion vectors:
In this step, clusterings of particles are performed with displacement motion vectors taken from pairs of frames. Only neighboring frames are considered (1, 2, and 3 unit time distances), and clusterings are computed in an independent way. For each pair of frames, the input to this step is the position of particles in each frame, and the output is a set of particle clusterings and their labels, valid for each pair of frames considered.

• Ensemble clustering of particles:
Here, all the clusterings computed in the previous step are processed simultaneously to produce a unique division of the full set of particles in sub-sets of particles in coherent motion, called meta-clusters; several sets of clustering labels are taken as input to this step, and a unique set of segmentation labels (several particles in coherent motion share the same segmentation label) are returned as output.

• Meta-clustering validation:
In this step, particles that were segmented in the previous step are compared to meta-cluster prototypes in terms of motion and spatial position to detect incorrectly labeled particles and, when this occurs, particles are re-labeled. A set of segmentation labels is taken as input, and a corrected set of segmentation labels is returned as output.

• Spatial filtering:
In this step, outliers are eliminated and groups of adjacent particles that are not significant. The particle labels are analyzed spatially, and links between particles are created to define spatial adjacency in each frame. Small groups of adjacent particle labels that are not significant are then re-assigned. This step takes as input a set of particle labels, and returns as output a filtered set of particle labels.

2.3 Dense Segmentation Extraction

The proposed motion segmentation method is the dense segmentation extraction. This stage takes as input the original video frames, the segmentation labels returned by the second stage, as well as the particle positions returned by the first stage, and returns as the output the corresponding segmentation labels for each pixel of each frame of the video sequence. This is equivalent to the segmentation of a spatio-temporal volume in several tunnels. The dense segmentation extraction is done by creating implicit functions for each particle, based on motion and spatial position. This representation of motion segmentation through tunnels can be employed to obtain efficient motion predictions for video coding applications. All the stages of the proposed approach are processed sequentially.

Every stage is performed for the entire video before going to the next stage. Thus, the proposed motion segmentation method can not be used in online applications, without video partitioning.

3. 3D Model Segmentation

The segmentation of a polygonal model consists in the decomposition of the 3D geometry into sub-elements which have generally uniform properties. The semantic segmentation should be ideally performed fully automatically to imitate the human visual perception and the decision intents. But in most of the applications (Cultural Heritage, 3D city models, etc.) the user intervention is still mandatory to achieve more accurate results. Following [58], the main reasons that limit the automatic reconstruction of semantic models are related to:

• the definition of a target model which restricts object configurations to sensible building structures and their components, but which is still flexible enough to cover (nearly) all existing buildings in reality;
• the geometric and radiometric complexity of the input data and reconstructed 3D models;
• data errors and inaccuracies, uncertainty or ambiguities in the automatic interpretation and segmentation;
• the reduction of the search space during the interpretation process.

The interpretation and segmentation of a 3D model allow to generate a topologically and semantically correct model with structured boundary. The 3D geometry and the related semantic information have to be structured coherently in order to provide a convenient basis for simulations, urban data analyses and mining, facility management, thematic inquiries, archaeological analyses, policies planning, etc. The CityGML,conceived as target data format, fulfills all these requirements and became the standard approach for 3D city models [59].

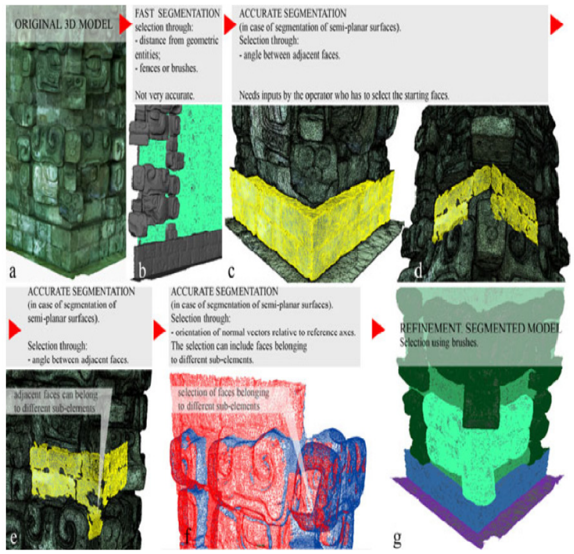

Figure 2. Example of automated segmentation of complex and detailed polygonal model: a fast but inaccurate segmentation can be improved with geometric constraints and manual refinements to separate the narrative elements

In the literature, the most effective automated segmentation algorithms are based on 3D volumetric approaches, primitive fitting or geometric segmentation methods. While the former two approaches segment meshes by identifying polygons that correspond to relevant feature of the 3D shape, the latter segments the mesh according to the local geometrical properties of 3D surface.A comparative study between some segmentation algorithms and related applications have been presented in [60], where only visual results were presented, without any quantitative evaluation of the algorithm’s effectiveness. In [61] a more complete and to-date review of mesh segmentation techniques is presented. A fully automatic protocol for the quantitative evaluation of 3D mesh segmentation algorithms aimed at reaching an objective evaluation of their effects are shown in [62, 63]. In particular, they provided 3D mesh segmentation benchmark in order to help researchers to develop or improve automatic mesh segmentation algorithms. As automation cannot provide for satisfactory results in all the possible dataset, [64] developed a methodology which combines different automatic segmentation algorithms with an interactive interface to adjust and correct the segmented polygonal models.

The methodology specified here also follow these concepts and uses a combination of automated and interactive segmentation tools according to the 3D model and its complexity (Figure.2). Furthermore the segmentation is generally performed according to rules or specifications differs for each project. Therefore the user intervention is generally not neglected in order to derive correct subdivisions of the polygonal models.

The segmentation procedure performs:

• an automatic geometric separation of the different mesh portions using surface geometric information and texture attributes;
• a manual intervention to adjust the boundaries of the segmented elements
• an assisted annotation of the sub-elements that constitute the segmented 3D model.

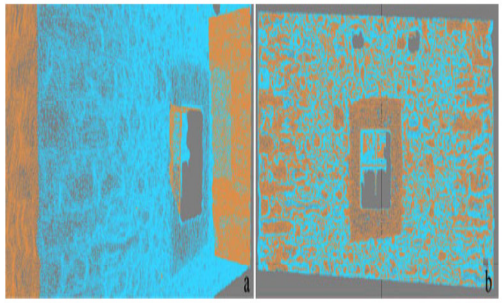

Figure 3. Segmentation of a stone wall of a castle (Figure 2-f) with small surface irregularities. a) result derived aggregating the main surface orientations; b) result of the segmentation following planar adjacent faces.

The geometric segmentation requires recognizing the transition between the different geometric elements of a 3D model. Automatic procedures to select and group faces of 3D models are available in common modeling packages (Maya, 3DS Max, Rhino, Meshlab, etc.). Faces can be separated and grouped using constraints such as inclination of adjacent faces, lighting or shading values. The surface normals are generally a good indicator to separate different sub-elements , when semi-planar faces need to be separated from reliefs (Figure 3 and Figure 4). The detection of lighting or texture transitions can be instead eased applying filters or using edge detector algorithms. This can be quite useful in flat areas with very low geometric discontinuities where the texture information allow to extract, classify and segment figures or relevant features for further uses (Figure 5).For the correct hierarchical organization and visualization of the segmented subelements, a precise identification of the transition borders in the segmented meshes is required. For complex geometric models constituted by detailed and dense meshes, the manual intervention is generally required. Another aspect that has to be considered during the geometric segmentation of complex and fully 3D models is related to the possibility of subdividing only visible surfaces or to build complete volumes of sub-element models, modeling also non- visible closure or transition surfaces.

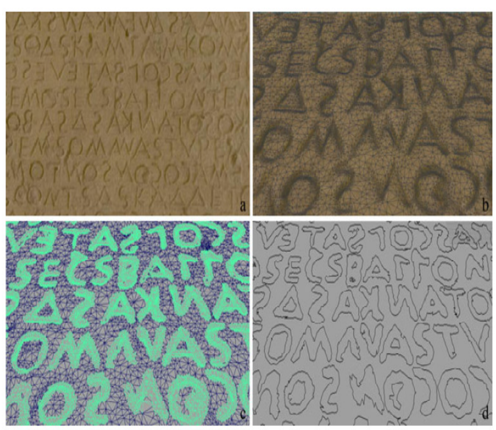

Figure 4 a) Image of the law code in Gortyna (Figure 2-e) with symbols of ca 3-4 mm depth; b)close view of the 3D textured polygonal model; c) automatic identification of the letters using geometric constraints; d) final segmentation and vectorialization of the letters

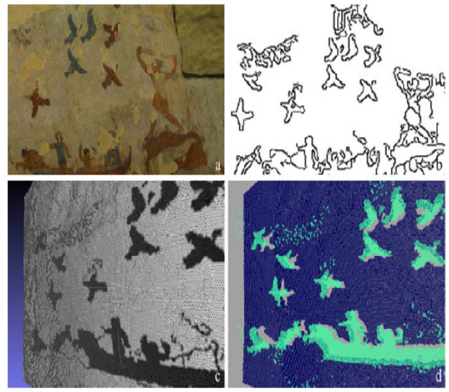

Figure 5 a) The 3D model of an underground frescoed Etruscan tomb, b) filters to detect edges on the texture information of the 3D model; c,d) ease of lighting transitions and final segmentation of the polygonal model

The semantic segmentation of a geometric 3D model is followed by the assignation to each sub-element of characteristics and information which need to be represented, organized and managed using advanced repository of geometric and appearance components to allow visualization and interaction with the digital models as well as database queries.

4. CONCLUSION

A study has been made in this paper for the identification of coherent motion in adaptively sampled videos. This technique provides a new way of linking low-level information in videos to high-level concepts that can be employed directly in video coding. This approach can be useful in many other image processing and computer vision tasks, including object tracking, information retrieval and video analysis.

The proposed particle segmentation method uses ensemble clustering to combine particle clusters obtained for adjacent frames, allowing the identification of long-range motion patterns, which we represent as spatio-temporal volumes called tunnels. The identification of long-range motion patterns is crucial to take full advantage of temporal redundancy in segmentation- based video coding.

Another limitation of the proposed segmentation method concerns the type of motions that are better handled with this approach. Since we generate clusters based on similarity of the 2-D projected motion vectors, sequences with pronounced perspective effects and/or with complex 3-D spatial motion tend to be over-segmented. In order to overcome this drawback Reality-Based 3-D modeling has been proposed here which enable us to semantically segment complex reality-based 3-D models.

REFERENCES

[1] T. Boult and L. G. Brown, "Factorization-based segmentation of motions," in Proc. IEEE Workshop Visual Motion, 1991, pp. 179–186.

[2] Costeira and Kanade, "A multibody factorization method for independently moving objects," Int. J. Comput. Vis., vol. 29, pp. 159–179, 1998.

[3] K. Kanatani, "Motion segmentation by subspace separation and model selection," in Proc. 8th IEEE Int. Conf. Computer Vision, 2001, vol. 2, pp. 586–591.

[4] A. Gruber and Y.Weiss, "Multibody factorization with uncertainty and missing data using the em algorithm," in Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition, 2004, vol. 1, pp. I-707–I-714.

[5] Yan and Pollefeys, “A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate,” presented at the ECCV, 2006.
[6] R. Vidal and R. Hartley, “Motion segmentation with missing data using powerfactorization and gpca,” in Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition, 2004, vol. 2, pp. II-310–II-316.
[7] A. Mitiche, R. El-Feghali, and A.-R. Mansouri, “Motion tracking as spatio-temporal motion boundary detection,” Robot. Autonom. Syst., vol. 43, no. 1, pp. 39–50, 2003.
[8] R. Feghali and A. Mitiche, “Spatiotemporal motion boundary detection and motion boundary velocity estimation for tracking moving objects with a moving camera: a level sets pdes approach with concurrent camera motion compensation,” IEEE Trans. Image Process., vol. 13, no. 11, pp. 1473–1490, Nov. 2004.
[9] D. Cremers and S. Soatto, “Variational space-time motion segmentation,” in Proc. 9th IEEE Int. Conf. Computer Vision, 2003, vol. 2, pp. 886–893.
[10] M. Ristivojevic and J. Konrad, “Space-time image sequence analysis: Object tunnels and occlusion volumes,” IEEE Trans. Image Process., vol. 15, pp. 364–376, 2006.
[11] L. Torres, M. Kunt, and F. Pereira, “Second generation video coding schemes and their role in mpeg-4,” in Proc. MPEG-4 Eur. Conf. Multimedia
Applications, Services and Techniques, 1996, pp. 799–824.
[12] M. Irani and P. Anandan, “About direct methods,” in Proc. Int. Workshop on Vision Algorithms: Theory and Practice, 2000, pp. 267–277.
[13] R. Tron and R. Vidal, “A benchmark for the comparison of 3-d motion segmentation algorithms,” in Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2007, pp. 1–8.
[14] H. Sekkati and A. Mitiche, “Concurrent 3-d motion segmentation and 3-D interpretation of temporal sequences of monocular images,” IEEE Trans. Image Process., vol. 15, no. 3, pp. 641–653, Mar. 2006.
[15] R. Lublinerman, O. Camps, and M. Sznaier, “Dynamics based robust motion segmentation,” in Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition, 2006, vol. 1, pp. 1176–1184.
[16] R. Vidal, S. Sastry, and Y. Ma, “Generalized principal component analysis (gpca),” IEEE Trans. Pattern Anal. Mach. Intell., pp. 1945–1959, 2005.
[17] H. Sekkati and A. Mitiche, “Joint optical flow estimation, segmentation,
and 3D interpretation with level sets,” Comput. Vis. Image Understand., pp. 89–100, Aug. 2006.
[18] A. Mitiche and H. Sekkati, “Optical flow3Dsegmentation and interpretation: A variational method with active curve evolution and level sets,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 28, no. 11, pp. 1818–1829, Nov. 2006.
[19] K. Schindler, U. James, and H. Wang, “Perspective n-view multibody structure-and-motion through model selection,” in Proc. 9th Eur. Conf. Computer Vision, 2006, pp. 606–619.
[20] T. Li, D. Singaraju, R. Vidal, and V. Kallem, “Projective factorization of multiple rigid-body motions,” in Proc. IEEE Conf. Computer Vision and Pattern Recognition, 2007, pp. 1–6.
[21] P. H. S. Torr and A. Zisserman, “Feature based methods for structure and motion estimation,” in Proc. Int. Workshop on Vision Algorithms: Theory and Practice, 2000, pp. 278–294.
[22] D. Cremers, T. Kohlberger, and C. Schnörr, A. Heyden, Ed. et al., “Nonlinear shape statistics in mumford-shah based segmentation,” in Proc. Eur. Conf. Computer Vision, Copenhagen, Denmark, May 2002, vol. 2351, pp. 93–108.
[23] D. Cremers, S. J. Osher, and S. Soatto, “Kernel density estimation and intrinsic alignment for shape priors in level set segmentation,” Int. J. Comput. Vis., vol. 69, no. 3, pp. 335–351, Sep. 2006.
[24] D. Cremers, “Dynamical statistical shape priors for level set based tracking,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 28, no. 8, pp. 1262–1273, Aug. 2006.

[25] H. Sekkati and A. Mitiche, "Joint dense 3D interpretation and multiple motion segmentation of temporal image sequences: A variational framework with active curve evolution and level sets," Image Process., vol. 1, pp. 553–556, 2004.
[26] N. Ichimura, "A robust and efficient motion segmentation based on orthogonal projection matrix of shape space," in Proc. IEEE Conf. Computer
Vision and Pattern Recognition, 2000, vol. 2, pp. 446–452.
[27] Y. Li, D. Hatzinakos, and A. Venetsanopoulos, "A multi-frame, regionfeature based technique for motion segmentation," in Proc. Int. Conf.
Image Processing, 1999, vol. 1, pp. 11–15.
[28] S. Smith and J. Brady, "Asset-2: Real-time motion segmentation and shape tracking," IEEE Trans. Pattern Anal. Mach. Intell., vol. 17, no. 8, pp. 814–820, Aug. 1995.
[29] C. Harris and M. Stephens, "A combined corner and edge detector," in Proc. 4th Alvey Vision Conf., 1988, pp. 147–151.
[30] J. Shi, J. Shi, C. Tomasi, and C. Tomasi, "Good features to track," in Proc. IEEE Computer Society Computer Vision and Pattern Recognition, 1994, pp. 593–600.
[31] Lowe, "Distinctive image features from scale-invariant keypoints," Int
J. Comput. Vis., pp. 91–110, Nov. 2004.
[32] M. Gelgon, P. Bouthemy, and J.-P. Le Cadre, "Recovery of the trajectories of multiple moving objects in an image sequence with a PMHT approach," Int. J. Comput. Vis., vol. 23, no. 1, pp. 19–31, 2005.
[33] B. Horn, Robot Vision (MIT Electrical Engineering and Computer Science). Cambridge, MA: The MIT Press, 1986.
[34] P. Sand and S. Teller, "Particle video: Long-range motion estimation using point trajectories," Comput. Vis. Pattern Recognit., vol. 2, pp. 2195–2202, 2006.
[35] J. Wang, G. Eude, Q. Liu, and H. Lu, "Moving object segmentation using graph cuts," in Proc. ICSP 7th Int. Conf. Signal Processing, 2004, vol. 1, pp. 777–780.
[36] H. Xu, M. Kabuka, and A. Younis, "Automatic moving object extraction for content-based applications," IEEE Trans. Circuits Syst. Video
Technol., vol. 14, no. 6, pp. 796–812, Jun. 2004.
[37] P. Viswanath and K. Jayasurya, "A fast and efficient ensemble clustering method," in Proc. 18th Int. Conf. Pattern Recognition, 2006, vol. 2, pp. 720–723.
[38] A. Strehl and J. Ghosh, "Cluster ensembles—A knowledge reuse framework for combining multiple partitions," J. Mach. Learn. Res, vol. 3, pp. 583–617, 2003.
[39] A. Fred and A. Jain, "Combining multiple clusterings using evidence accumulation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, pp. 835–850, 2005.
[40] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, pp. 603–619, 2002.
[41] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," ACM Comput. Surv., vol. 31, pp. 264–323, 1999.
[42] T. Meier and K. Ngan, "Automatic segmentation of moving objects for video object plane generation," IEEE Trans. Circuits Syst. Video Technol., vol. 8, no. 5, pp. 525–538, 1998.
[43] Y. Nie and K.-K. Ma, "Adaptive rood pattern search for fast blockmatching motion estimation," IEEE Trans. Image Process., vol. 11, pp.
1442–1449, 2002.
[44] Gruen, A.: Reality-based generation of virtual environments for digital earth. International Journal of Digital Earth 1(1) (2008)
[45] Remondino, F., El-Hakim, S.: Image-based 3d modelling: a review. The Photogrammetric Record 21(115), 269–291 (2006)
[46] Blais, F.: A review of 20 years of range sensors development. Journal of Electronic Imaging 13(1), 231–240 (2004)

[47] Vosselman, G., Maas, H.-G.: Airborne and terrestrial laser scanning, p. 320. CRC Press, Boca Raton (2010)

[48] Yin, X., Wonka, P., Razdan, A.: Generating 3d building models from architectural drawings: A survey. IEEE Computer Graphics and pplications 29(1), 20–30 (2009)

[49] Stamos, I., Liu, L., Chen, C., Woldberg, G., Yu, G., Zokai, S.: Integrating automated range registration with multiview geometry for photorealistic modelling of large-scale scenes. International Journal of Computer Vision 78(2-3), 237–260 (2008)

[50] Guidi, G., Remondino, F., Russo, M., Menna, F., Rizzi, A., Ercoli, S.: A multi-resolution methodology for the 3d modeling of large and complex archaeological areas. International Journal of Architectural Computing 7(1), 40–55 (2009)

[51] Remondino, F., El-Hakim, S., Girardi, S., Rizzi, A., Benedetti, S., Gonzo, L.: 3D virtual reconstruction and visualization of complex architectures - The 3d-arch project. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 38(5/W10), on CD-ROM (2009)

[52] Mueller, P., Wonka, P., Haegler, S., Ulmer, A., Van Gool, L.: Procedural modeling of buildings. ACM Transactions on Graphics 25(3), 614–623 (2006)

[53] Agarwal, S., Snavely, N., Simon, I., Seitz, S., Szelinski, R.: Building Rome in a day. In: Proc. ICCV 2009, Kyoto, Japan (2009)

[54] Remondino, F., El-Hakim, S., Gruen, A., Zhang, L.: Development and performance analysis of image matching for detailed surface reconstruction of heritage objects. IEEE Signal Processing Magazine 25(4), 55–65 (2008)

[55] Hirschmueller, H.: Stereo processing by semi-global matching and mutual information. IEEE Transactions on Pattern Analysis and Machine Intelligence 30(2), 328–341 (2008)

[56] Hiep, V.H., Keriven, R., Labatut, P., Pons, J.P.: Towards high-resolution large-scale multiview stereo. In: Proc. CVPR 2009, Kyoto, Japan (2009)

[57] Patias, P., Grussenmeyer, P., Hanke, K.: Applications in cultural heritage documentation. Advances in Photogrammetry, Remote Sensing and Spatial Information Sciences. In: 2008 ISPRS Congress Book, vol. 7, pp. 363–384 (2008)

[58] Barazzetti, L., Remondino, F., Scaioni, M.: Automation in 3D reconstruction: results on different kinds of close-range blocks. In: ISPRS Commission V Symposium Int. Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Newcastle upon Tyne, UK, vol. 38(5) (2010)

[59] Nagel, C., Stadler, A., Kolbe, T.H.: Conceptual Requirements for the Automatic Reconstruction of Building Information Models from Uninterpreted 3D Models. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 38(3-4/C3) (2009) 124 A.M. Manferdini and F. Remondino

[60] Attene, M., Katz, S., Mortara, M., Patané, G., Spagnuolo, M., Tal, A.: Mesh segmentation - a comparative study. In: Proc. IEEE International Conference on Shape Modeling and Applications 2006, p. 12. IEEE Computer Society, Washington (2006)

[61] Shamir, A.: A survey on mesh segmentation techniques. Computer Graphics Forum 27(6), 1539–1556 (2008)

[62] Benhabiles, H., Vandeborre, J.-P., Lavoué, G., Daoudi, M.: A framework for the objective evaluation of segmentation algorithms using a ground-truth of human segmented 3Dmodels. In: Proc. IEEE Intern. Conference on Shape Modeling and Applications 2009, p. 8. IEEE Computer Society, Washington (2009)

[63] Chen, X., Golovinskiy, A., Funkhouser, T.: A Benchmark for 3D Mesh Segmentation. Proc. ACM Transactions on Graphics 28(3), 12 (2009)

[64] Robbiano, F., Attene, M., Spagnuolo, M., Falcidieno, B.: Part-based annotation of virtual 3d shapes. In: Proc. of the International Conference on Cyberworlds, pp. 427–436. IEEE Computer Society, Washington (2007)

[65]Image Fast Template Matching Algorithm Based on Projection and Sequential Similarity Detecting Liyong Ma, Yude Sun, Naizhang Feng, Zheng Liu School of Information Science and Engineering Harbin Institute of Technology at Weihai Weihai, China maly@hitwh.edu.cn , sun-yd@163.com, fengnz@hitwh.edu.cn, withitren@yahoo.com.cn (2010)