Pattern Recognition Letters xxx (2011) xxx–xxx

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Hierarchical spatio-temporal extraction of models for moving rigid parts

Nicole M. Artner [a,c,*], Adrian Ion [b,c], Walter G. Kropatsch [c]

[a] AIT – Austrian Institute of Technology, Donau-City-Straße 1, Vienna 1220, Austria
[b] University of Bonn, Institute for Numerical Simulation, Wegelerstraße 4, Bonn 53115, Germany
[c] Vienna University of Technology, Institute of Computer Graphics and Algorithms, Pattern Recognition and Image Processing Group (PRIP), Favoritenstraße 9/186-3, Vienna 1040, Austria

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Rigid parts
Articulated objects
Model extraction
Graph pyramid

ABSTRACT

This paper presents a method to extract a part-based model of an observed scene from a video sequence. Independent motion is a strong cue that two points belong to different "rigid" entities. Conversely, *things that move together throughout the whole video belong together* and define a "rigid" object or part. Successfully tracked features indicate trajectories of salient points in the scene. A triangulated graph connects the salient points and encodes their local neighborhood in the first frame. The length variation of the triangle edges is used to label them as *relevant* (on an object) or *separating* (connecting different objects). A following grouping process uses the motion of the triangles marked as relevant as a cue to identify the "rigid" parts of the foreground or the background. The choice of the motion-based grouping criterion depends on the type of motion: in the image plane or out of the image plane. The result is a hierarchical description (graph pyramid) of the scene, where each vertex in the top level of the pyramid represents a "rigid" part of the foreground or the background, and encloses to the salient features used to describe it. Promising experimental results show the potential of the approach.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

Visual tracking of articulated objects and their rigid parts is an important and still challenging task in computer vision (see for example the surveys Yilmaz et al., 2006; Gavrila, 1999; Moeslund et al., 2006; Aggarwal and Cai, 1999). Example applications are the analysis of human motion for action recognition, motion based diagnosis and identification, motion capture for 3D animation and human computer interfaces.

To be able to detect and associate instances of the object of interest in consecutive frames, tracking methods use a model of the *target* (the object to be tracked). This model is at the minimum a rectangle-shaped close-up of the object (called a template) or a color histogram, but can be as sophisticated as an online-trained classifier (Godec et al., 2010), or a hierarchical description of the objects' parts and their salient features (Artner et al., 2011).

Sources for the target model and the objects' position in the initial frame are: user input, various segmentation and/or object recognition methods (e.g. Felzenszwalb and Huttenlocher, 2005), or an initialization sequence (this work). In the latter case, the sequence is usually made specifically for this purpose, it emphasizes all relevant properties of the target object and does not pose a too high challenge to the visual aspect of the extraction of these properties (e.g. no unnecessary occlusion).

This paper presents a method to extract a model for the "rigid" parts of articulated objects using the motion information in an "initialization video". It follows the intuition that salient features on the same "rigid" parts will move together:

1. The input of the framework consists of point correspondences of salient features (e.g. corner points) over time (see Fig. 2). These correspondences result in a set of trajectories, which give information about the motion in a scene.
2. A triangulated graph is built based on the positions of the features in the first frame. It encodes the spatial relationships (edges) between the features (vertices) and its deformation over time is the basis for all processes and decisions (see Fig. 3).
3. The motion in the triangulation is analyzed by determining the motion of the triangles in or out of the image plane (see Section 3).
4. In a spatio-temporal filtering the features describing "rigid" parts of foreground and background are selected. The result is a labeling of each triangle in the graph as *relevant* or *separating* depending on the variation of the edge lengths (see Section 4.1 and Fig. 7).
5. All *relevant* triangles are the input for the hierarchical grouping process, which is realized by building a graph pyramid, where the base level is a graph encoding the adjacency of the triangles (see Section 2). The grouping depends on the similarity of the motion of the triangles over time (see Section 4.2 and Fig. 9).

* Corresponding author at: AIT – Austrian Institute of Technology, Donau-City-Straße 1, Vienna 1220, Austria.
E-mail addresses: artner@prip.tuwien.ac.at (N.M. Artner), ion@ins.uni-bonn.de (A. Ion), krw@prip.tuwien.ac.at (W.G. Kropatsch).

0167-8655/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2011.05.005

Please cite this article in press as: Artner, N.M., et al. Hierarchical spatio-temporal extraction of models for moving rigid parts. Pattern Recognition Lett. (2011), doi:10.1016/j.patrec.2011.05.005

6. The output of this process is a hierarchical description of the "rigid" parts in the scene, where in the optimal case each vertex in the top level of the pyramid represents one "rigid" part (see Fig. 9).

Like the methods in (Yan et al., 2008; Costeira and Kanade, 1998), the presented approach requires features that haven been tracked throughout the whole video sequence. We are interested in obtaining a model that describes the parts using the most salient features. The above requirement can be seen as a preprocessing step, which filters out the non-salient features.

1.1. Related work

The work in this paper is related to the concept of *video object segmentation* (VOS), where the task is to separate foreground from background in an image sequence. Notice however the difference: VOS methods try to group pixels as robustly as possible in a possibly highly cluttered scene, whereas in our case the emphasis is on extracting the relevant model properties (salient features and "rigid" parts) from a less cluttered scene. VOS methods can be divided into two categories (Celasun et al., 2001):

Two-frame motion/object segmentation: Altunbasak et al. (1998) a combination of pixel-based and region-based segmentation methods. Their goal is to obtain the best possible segmentation results on a variety of image sequences. Castagno et al. (1998) describe a system for interactive video segmentation. An important key feature of the system is the distinction between two levels of segmentation: regions and object segmentation. Chen et al. (2006) propose an approach to segment highly articulated objects by employing weak-prior random forests. The random forests are used to derive the prior probabilities of the object configuration for an input frame. Then these priors are applied to guide the grouping of over-segmented regions. The work of Alatan et al. (1998) presents the activities of the COST 211^{ter} group dedicated toward image and video sequence analysis and segmentation. This work is an important technological aspect for the success of emerging object-based MPEG-4 and MPEG-7 multimedia applications.

Multi-frame spatio-temporal segmentation/tracking: Celasun et al. (2001) present VOS based on 2D meshes. Tekalp et al. (1998) describe 2D mesh-based modeling of video objects as a compact representation of motion and shape for interactive video manipulation, compression, and indexing. Li et al. (2001) propose to use affine motion models to estimate the motion of homogeneous regions.

There are VOS methods explicitly dealing with the segmentation of articulated objects (e.g. Chen et al., 2006), but the result of these approaches is still only a separation of foreground and background.

Motion segmentation: In comparison to VOS, motion segmentation approaches work on the basis of trajectories of features and not on the pixel level.

Lauer and Schnörr (2010) present an approach to automatically segment multiple motions from tracked features points by spectral embedding and clustering of linear subspaces. Nordberg and Zografos (2010) also work on motion segmentation and propose an approach using the geometry of 6 points in 2D images to infer motion consistency with regard to rigid 3D motion. As in the work of Lauer and Schnörr (2010), they are able to segment the motion of an arbitrary number of moving objects. Nevertheless, articulated objects like humans are segmented as one object and not split into their moving rigid parts.

Automatic articulated model extraction: The most similar works to the presented approach lie in the field of automatic model extraction.

Yan et al. (2008) use factorization to analyze the trajectories of tracked features and cluster them into subspaces corresponding to parts. This method can cope with affine motion and can extract both articulation axes and joints, but their parts have to be fully rigid or defined by a linear combination of a subset of the features. Walther and Würtz (2009, 2008) a method to learn a 2D pictorial model of observed humans for pose estimation. Their approach is based on the 2D trajectories of features, which are grouped to limbs using spectral clustering. The extracted limbs are refined by employing multi-label image segmentation methods. They also extract body kinematics by finding joint connections between the segmented limbs. Drouin et al. (2008) propose an approach which incrementally identifies object parts in videos. The main contribution of their approach is the *Modeler*, which allows to track several candidates for the model of the foreground object in parallel. As the approach in (Walther and Würtz, 2009) their work is limited to movements in the 2D image plane. In comparison to our approach, this work requires the initialization of the foreground object.

1.2. Contributions

Our approach goes beyond the related works by following:

- Analysis of trajectories of features and their behavior on a higher abstraction level – in a triangulation. The related works group pixels or features independently without considering spatial proximity and relationships.
- A generic grouping framework, which allows the usage of different grouping criteria depending on the motion of the objects in the video. This flexibility allows to adjust and optimize the presented framework for any application.
- Build a graph pyramid based on the adjacency graph of the triangulation, guided by the motion of features. There is only a small number of works (e.g. Conte et al., 2005), where graph pyramids are employed on spatio-temporal data and to the best of our knowledge there is no work employing graph pyramids to extract models based on the motion of features in a triangulation.
- Besides providing a grouping of features into "rigid" parts like the related work, the presented approach additionally supplies a hierarchical description of the "rigid" parts, which can be used as a model in coarse-to-fine tracking scenarios.

This paper extends the work in (Artner et al., 2009) for out of plane motion and shows additional experimental results.

1.3. Overview

The paper is organized as follows: in Section 2 graph pyramids which are employed for the grouping process are briefly recalled. Section 3 explains how the motion of features in and out of the image plane is described and analyzed. Section 4 presents the generic framework, which allows to identify the "rigid" parts of articulated objects. Section 5 shows experiments and in Section 6 conclusions are given.

2. Recall: irregular graph pyramids

A *region adjacency graph* (RAG), encodes the adjacency of regions in a partition. A vertex is associated to each region, vertices of neighboring regions are connected by an edge. Classical RAGs do not contain any self-loops or parallel edges. An *extended region adjacency graph* (eRAG) is a RAG that contains the so-called *pseudo edges*, which are self-loops and parallel edges used to encode

Please cite this article in press as: Artner, N.M., et al. Hierarchical spatio-temporal extraction of models for moving rigid parts. Pattern Recognition Lett. (2011), doi:10.1016/j.patrec.2011.05.005

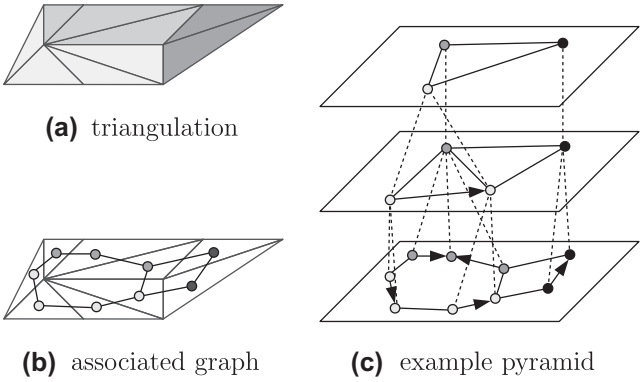

Fig. 1. Example graph pyramid for a triangulation. (a) Triangulation. (b) Associated adjacency graph, a vertex for each triangle, edges are added for triangles sharing a side. (c) Graph pyramid: contracted edges are marked with an arrow.

neighborhood relations to a cell completely enclosed by one or more other cells (Kropatsch, 1995). The *dual* graph of an eRAG G is called *boundary graph* and is denoted by $\overline{G}$ (G is said to be the *primal* graph of $\overline{G}$). The edges of $\overline{G}$ represent the boundaries of the regions encoded by G, and the vertices of $\overline{G}$ represent points where boundary segments meet. G and $\overline{G}$ are planar graphs. There is a one-to-one correspondence between the edges of G and the edges of $\overline{G}$, which also induces a one-to-one correspondence between the vertices of G and the 2D cells (denoted by *faces*[1]) of $\overline{G}$. The dual of $\overline{G}$ is again G. The following operations are equivalent: edge contraction in G with edge removal in $\overline{G}$, and edge removal in G with edge contraction in $\overline{G}$.

A (dual) *irregular graph pyramid* (Kropatsch, 1995; Kropatsch et al., 2005) is a stack of successively reduced planar graphs $P = \left\{\left(G_0, \overline{G}_0\right), \ldots, \left(G_n, \overline{G}_n\right)\right\}$. Each level $\left(G_k, \overline{G}_k\right)$, $0 < k \leqslant n$ is obtained by first contracting edges in G_{k-1} (removal in $\overline{G}_{k-1}$), if their end vertices have the same label (regions should be merged), and then removing edges in G_{k-1} (contraction in $\overline{G}_{k-1}$) to simplify the structure. The contracted and removed edges are said to be *contracted* or *removed* in $\left(G_{k-1}, \overline{G}_{k-1}\right)$. In each G_{k-1} and $\overline{G}_{k-1}$, contracted edges form trees called *contraction kernels*. One vertex of each contraction kernel is called a *surviving vertex* and is considered to have been "survived" to $\left(G_k, \overline{G}_k\right)$. The vertices of a contraction kernel in level $k-1$ form the *reduction window* $W(v)$ of the respective surviving vertex v in level k. The *receptive field* $F(v)$ of v is the (connected) set of vertices from level 0 that have been "merged" to v over levels $0, \ldots, k$.

For the sake of simplicity, the rest of the paper will only use the adjacency graph G, but for correctly encoding the topology, both G and $\overline{G}$ have to be maintained. Fig. 1 shows an example triangulation and pyramid.

3. Analysis of motion

The grouping of the triangles in the scene into "rigid" parts relies on the intuitive idea that features which move together belong to the same "rigid" part. This section explains how motion is described and analyzed in the presented framework. The description and analysis of motion is based on the trajectories of the independently tracked features (see Fig. 2). It is used later to filter the input for the hierarchical grouping process (see Section 4.1) and for the grouping itself (see Section 4.2).

[1] Not to be confused with the vertices of the dual of a RAG (sometimes also denoted by the term *faces*).

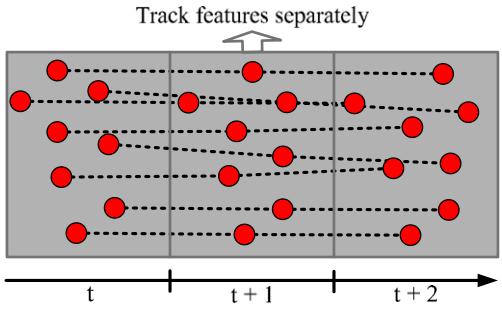

Fig. 2. Input of framework: trajectories of independently tracked features.

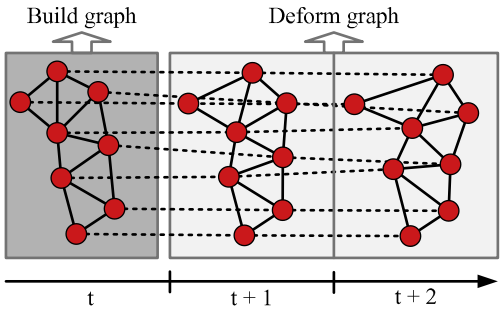

Fig. 3. Triangulated graph is built in first frame (dark box) and deformed over time (bright box).

The presented framework is generic with respect to the type of the motion representation and therefore to the criterion in the hierarchical grouping process.

In comparison to the related work, the presented framework analyzes the motion of features in a higher abstraction level – a triangulation. The vertices V of the triangulated graph G represent the tracked features (e.g. corner points with their positions). A Delaunay triangulation (Tuceryan and Chorzempa, 1991) is used in the first frame to insert the edges E, connect the features and represent their spatial relationships (see Fig. 3).

The advantage of analyzing the motion on triangles is the reduction of the ambiguity of motion in the 2D image plane (see Fig. 4 and Section 3.1) and the availability of necessary information for determining the affine transformation matrices for motion out of the image plane (see Section 3.2).

3.1. Motion in the image plane

If the movement of the objects in the scene is limited to the image plane, the motion of triangles can be described using their "translation invariant" *orientation variation* over time.

Definition 1. The *orientation variation* O_e *of an edge* e over time $t = 2, \ldots, n$, where n is the number of frames, is a 1D signal that encodes at each frame t the accumulated orientation change relative to the orientation at frame 1. More formally, for edge e and frame t,

$$O_e(t) = O_e(t-1) + \theta(t),$$

where $\theta(t)$ is the relative change in orientation (signed angle) of the edge e between frames t and $t-1$.

We assume that the rotation of an edge between two consecutive frames is less than 180° and therefore we do not deal with the problem of "circularity" (+180° = −180°).

If any of the two end points of an edge is fixed as the reference point and the other end point is turned around the reference point once this will give a value of 360° degrees and turning twice in the

Please cite this article in press as: Artner, N.M., et al. Hierarchical spatio-temporal extraction of models for moving rigid parts. Pattern Recognition Lett. (2011), doi:10.1016/j.patrec.2011.05.005

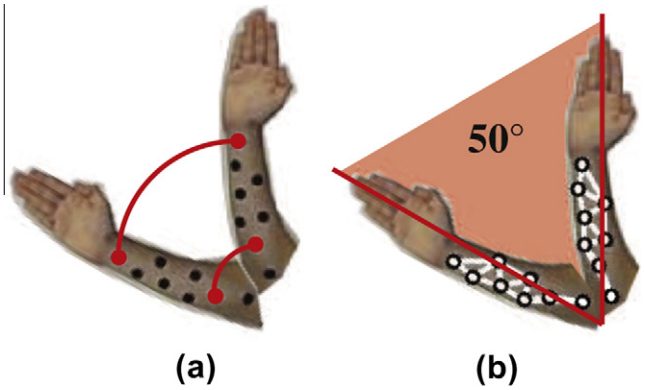

Fig. 4. Advantage of analyzing the motion of features in a triangulation in comparison to single trajectories/positions. (a) Trajectories of points on the same "rigid" part can differ a lot. (b) The orientation change of triangles on a "rigid" part are very similar.

same direction will give 720°, not 0°. The direction of rotation is encoded by the sign: counter clockwise (CCW) is positive, and clockwise (CW) is negative. Fig. 5 shows an example for the orientation variation of an edge: if turning 45° CCW, then again 45° CCW, and afterwards 90° CW, the computed variations will be 0°, 45° = 0° + 45°, 90° = 45° + 45°, 0° = 90°−90° (see Fig. 6).

Definition 2. The *orientation variation* O_r *of a triangle* r is the 1D signal obtained by taking the average of the 1D signals of the three edges of the triangle:

$$O_r(t) = \frac{1}{3}\sum_i O_{e_i}(t),$$

where e_i, $i = 0,1,2$ are the edges of the triangle.

Definition 3. The *similarity* X_O *between two in-plane orientation variation signals* O_1, O_2 is calculated as follows:

$$X_O(O_1, O_2) = \max_{t=2,\ldots,n}\{|O_1(t) - O_2(t)|\},$$

where $O_1(t)$ and $O_2(t)$ are the 1D signals corresponding to two triangles or two groups of triangles.

For an explanation of the 1D signal of a group of triangles refer to Section 4.2.

3.2. Motion out of the image plane

For objects moving out of the image plane the vertices of the triangles in the image are feature points corresponding to points projected from the 3D world to the image. The motion of the triangles is described using 2D affine transformation matrices.

An affine transformation can be described with the help of homogeneous coordinates by a matrix $M = 3 \times 3$:

$$P' = M * P, \tag{1}$$

where $P = 3 \times 1$, $P' = 3 \times 1$ are the homogeneous coordinates of the point(s) before and after applying the transformation. M can be written as:

$$M = \begin{bmatrix} A & B \\ 0\ 0 & 1 \end{bmatrix}, \tag{2}$$

where A describes the linear transformation (rotation, scaling or shear) and B is the translation (shift).

An affine transformation matrix M is uniquely determined by the correspondences of three non-collinear 2D points. The entries in the transformation matrix can be calculated by solving the resulting 6 linear equations (Klein, 1939).

As in the planar case (Section 3.1), we factor out the translation of the triangles when describing their motion, and focus on rotation as it provides a strong cue for (non-)rigid motion.

Hence, the presented approach uses the coordinates given by the three vectors $\vec{e_1} = v_1 - v_2$, $\vec{e_2} = v_2 - v_3$ and $\vec{e_3} = v_3 - v_1$ determined at two time instances, where $\{v_1, v_2, v_3\}$ are the three vertices of a triangle r.

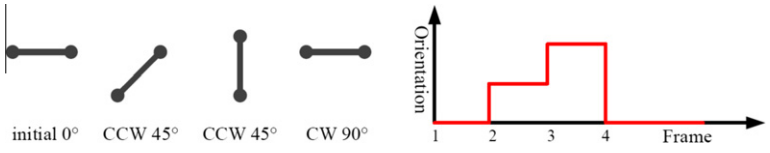

Fig. 5. Orientation variation of an edge as a 1D signal.

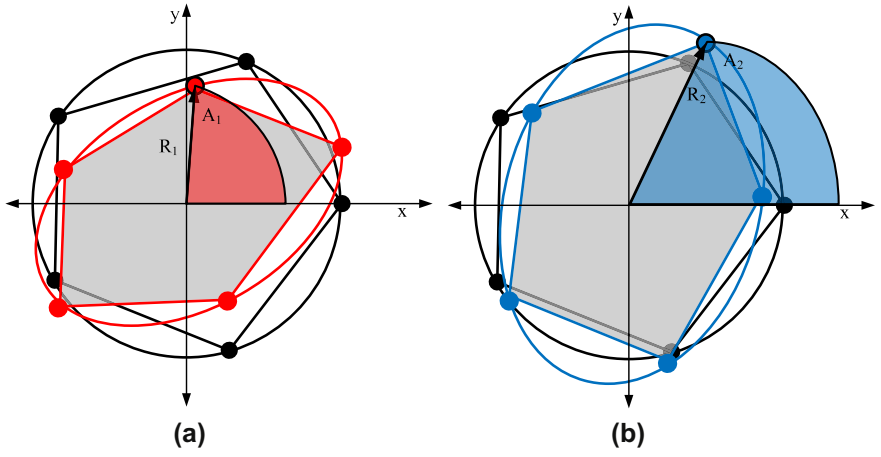

Fig. 6. Determining the similarity between two transformation matrices by applying them on a polygon inscribed in an unit circle (black polygon). Results of transformation matrices (a) T_1 and (b) T_2 are compared with the help of polar coordinates: radii (R_1,R_2) and angles (A_1,A_2).

Please cite this article in press as: Artner, N.M., et al. Hierarchical spatio-temporal extraction of models for moving rigid parts. Pattern Recognition Lett. (2011), doi:10.1016/j.patrec.2011.05.005

Definition 4. The (*affine*) *transformation* T_r *of a triangle* r over time $t = 2,\ldots,n$ is the signal that encodes for each frame t the (affine) transformation matrix that maps the vectors $\{\vec{e_1},\vec{e_2},\vec{e_3}\}$ of a triangle from the first frame to frame t:

$$P(t) = T_r(t) * P(1), \tag{3}$$

where $P(1)$ and $P(t)$ are 3×3 matrices having as rows $\{\vec{e_1},\vec{e_2},\vec{e_3}\}$ in homogeneous coordinates at time 1 and t.

The affine transformation matrix $T_r(t)$, which has the form in Eq. (2), is determined by solving a linear system of equations (Klein, 1939).

To determine a criterion for the hierarchical grouping process, a similarity measure for the transformation signals is needed. As there is no similarity measure to compare transformation matrices directly, we propose to apply the transformation matrices and compare the results (see Fig. 6).

The idea is to transform the points p_i, $i = 1,\ldots,k$ lying on an unit circle, which is centered at the origin (0,0), by the matrices T_1 and T_2 of two triangles and measure the similarity of the transformation matrices in the resulting positions $p_i^1 = T_1 * p_i$ and $p_i^2 = T_2 * p_i$.

Measuring the similarity by calculating the Euclidean distances between the two sets p_i^1 and p_i^2 has two disadvantages: (1) the Euclidean distance does not characterize well (linearly) in-plane rotation and (2) the observed effects of rotation in and out of the image plane on a triangle are defined on a different domain: rotation in the image plane maps linearly to angles, rotation out of the image plane is (non-linearly) observed as scaling.

Therefore, we propose to analyze the similarity using polar coordinates, which are better suited to describe in and out of plane rotation respectively by their radii and angles. Fig. 5 illustrates the concept by approximating the unit circle with a $k = 5$ sided regular polygon, where this approximation is also used for the experiments in Section 5.

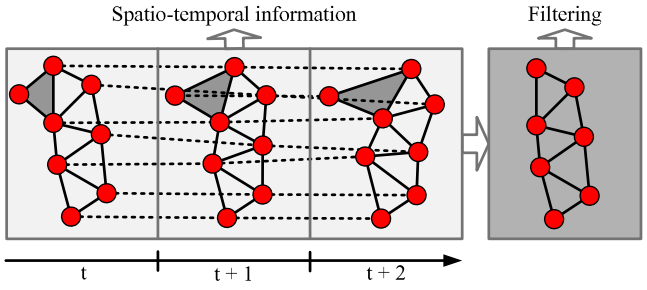

Fig. 7. Spatio-temporal filtering decides which triangles are input into the grouping process (dark box) based on the deformation of the graph over time (bright box).

Definition 5. The *similarity* X_T *of two transformation signals* T_1, T_2 is calculated by:

$$X_T(T_1,T_2) = \max_{t=2,\ldots,n}\{w \cdot d_A(T_1(t),T_2(t)) + (1-w)\cdot d_R(T_1(t),T_2(t))\},$$

where T_1, T_2 are the transformation matrices of two triangles or groups of triangles and d_A, d_R are the mean differences of the two sets p_i^1, p_i^2 regarding the angles and radii calculated for each frame t.

4. The generic grouping framework

In this section the proposed generic framework is presented, where the aim is to identify the "rigid" parts in a scene (e.g. decomposing a human into head, torso, and limbs).

The input of the framework is the spatio-temporal information about the tracked features, which includes the position of each feature over time (see Fig. 2), the motion of the triangles, and the resulting deformation of the triangulated graph (see Fig. 3).

First a spatio-temporal filtering selects the input for the hierarchical grouping process (see Fig. 7). Then each group of triangles belonging to a rigid part is generalized into a single vertex at the top level of an irregular graph pyramid (see Fig. 9).

4.1. Spatio-temporal filtering

The aim of the spatio-temporal filtering is to select *relevant* triangles for the input of the hierarchical grouping process. A triangle is *relevant* for the grouping process if the length of its edges

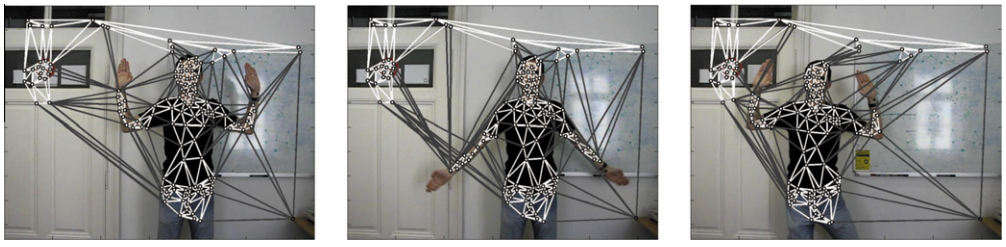

Fig. 8. Spatio-temporal filtering of triangles. Triangles on "rigid" parts are white and triangles connecting "rigid" parts are gray (see Section 4.1 for details on labeling).

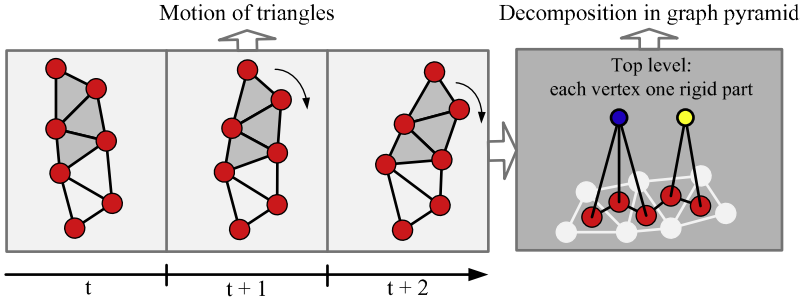

Fig. 9. Grouping process. Input: filtered triangles and their motion over time (bright box). The dark triangles rotate while the bright triangles stay still. Output: graph pyramid where each top vertex represents one "rigid" part of the scene (dark box).

Please cite this article in press as: Artner, N.M., et al. Hierarchical spatio-temporal extraction of models for moving rigid parts. Pattern Recognition Lett. (2011), doi:10.1016/j.patrec.2011.05.005

remains nearly stable over time, which indicates its affiliation to a "rigid" part.

The *edge length* of every edge $e = (v_1, v_2) \in E$ at time t is the Euclidean distance between the positions of the two vertices $e(t) = \|v_1(t) - v_2(t)\|$. Triangles which do not belong to a "rigid" part are outliers and mostly only stand out for a short period of time. Therefore, we propose to consider the extrema of an edge for the spatio-temporal filtering.

Definition 6. The *maximum variation of edge length* is the difference between the minimum and the maximum length of the edge in the video, more formally

$$l(e) = \max_{0 \leqslant t_1 < n} \{e(t_1)\} - \min_{0 \leqslant t_2 < n} \{e(t_2)\}.$$

A triangle is labeled as *relevant* if the edge length variations of all three edges e_1, e_2, e_3 are beneath a certain value i.e. $l(e_i) \leqslant \epsilon_r, 1 \leqslant i \leqslant 3$. Otherwise, the triangle is labeled as *separating*.

The value ϵ_r should be chosen s.t. triangles with a high deformation over time, which connect different "rigid" parts, are labeled as *separating* and triangles with points on the same "rigid" part will have edge length variations smaller than ϵ_r.

The result of the spatio-temporal filtering is a triangulation, where each triangle is labeled *relevant* or *separating* (see Figs. 7 and 8).

4.2. Hierarchical grouping process

The task of this process is to group the *relevant* triangles which survived the spatio-temporal filtering into groups of triangles, each describing one "rigid" part (see Fig. 9).

The grouping process is realized by building a irregular graph pyramid on the dual graph of the already existing triangulation. There are three reasons for the usage of an irregular graph pyramid: (1) using a hierarchy reduces the complexity of the grouping (global decisions become local ones), (2) the produced description can be used for a coarse-to-fine tracking approach and (3) in comparison to a regular graph pyramid the irregular one has the advantage that it is adaptive (shift and rotation invariant).

Algorithm 1: *BuildPyr*(T): Group triangles into "rigid" parts

Input: *relevant* triangles T (see Section 4.1)
1: $G_0 = (V_0, E_0)$
/*$V_0 = T$, *and* $(v,w) \in E_0 \Leftrightarrow$ *the corresponding triangles share an edge**/
2: $k = 0$
3: **repeat**
4: /*select edges to contract**/
$K = \emptyset$
$\forall v \in G_k$ **do** $K \leftarrow K \cup \arg\min_{(v,w) \in G_k} \{X(v,w)\}$
5: /*filter edges based on internal/external difference**/
$\forall (v,w) \in K$, **if** $X(v,w) > I'(v,w)$ **then**
$K \leftarrow K \backslash \{(v,w)\}$
6: **if** $K \neq \emptyset$ **then** break K into trees of radius 1
7: **if** $K \neq \emptyset$ **then** $G_{k+1} \leftarrow$ contract edges K in G_k and simplify
8: $k \leftarrow k + 1$
9: **until** $K = \emptyset$
Output: Graph pyramid $P = \{G_0, \ldots, G_{k-1}\}$.

Algorithm 1 creates a graph pyramid in which each vertex v of the top level identifies a detected "rigid" part, with its average motion description (orientation variation or transformation matrix over time) stored in $S(v)$. The receptive fields of these vertices identify the triangles that the respective part consists of.

Note that the pyramid is not built on the triangulated graph, but on its dual. In the base level G_0, one vertex is associated to each *relevant* triangle. Two vertices are connected by an edge if the respective triangles share a common edge. Edges to be contracted are selected from the edges proposed by the Minimum Spanning Tree algorithm by Boruvka (Nesetril et al., 2001) (Line 4).

The *external difference* $X(v,w)$ between two vertices (triangles) v and w depends on the representation for the motion of the triangles (see Section 3) and is computed using one of the two formulas:

$$\begin{aligned} X(v,w) &= X_O(S(v), S(w)), \\ X(v,w) &= X_T(S(v), S(w)), \end{aligned} \tag{4}$$

where $S(v)$ and $S(w)$ are the signals associated to v respectively w. For the vertices $v \in G_0$ of the base level, depending on the type of motion in the video, $S(v)$ is either the orientation variation signal O_r or the transformation signal T_t of the corresponding triangle r. For a vertex in a higher level it is computed as:

$$S(v) = \frac{\sum_{u \in W(v)} |F(u)| \cdot S(u)}{\sum_{w \in W(w)} |F(w)|}, \tag{5}$$

where $|F(v)|$ is the size of $F(v)$ and can be propagated up in the pyramid. Note that in the case of transformation matrices the computation is done element-wise (scalar multiplication, element-wise addition, division as scalar multiplication).

The *internal difference* of a vertex at level $k > 0$ is:

$$I(v) = \max(\max\{I(u)\}, \max\{X(w_i, w_j)\}), \tag{6}$$

where $u \in W(v)$ and $w_i, w_j \in W(v)$ s.t. w_i, w_j are connected by an edge. For the vertices in the base level $I(v) = 0$. The value $I'(v,w)$ is defined as:

$$I'(v,w) = \min\left(I(v) + \frac{\beta}{|F(v)|}, I(w) + \frac{\beta}{|F(w)|}\right), \tag{7}$$

where β is a parameter of the method that allows regions to start forming in the base level where $I(v) = 0$ for all vertices. The selected smallest edges (Line 4 of Algorithm 1) are accepted for contraction up to a weight of β. When going higher in the pyramid the size of the receptive fields increases and the contribution of β to the condition in Line 5 of Algorithm 1 rapidly decreases. As a result β sets an upper bound on the internal difference (deformation) of the produced parts.

Line 6 of Algorithm 1 keeps the contraction operations local (optimal for parallel processing) and avoids contracting the whole graph in a single level. It does this by excluding edges from K to obtain trees of radius 1 for the current contraction. The excluded edges will be selected again in the next level. In (Kropatsch et al., 2007) three methods, MIES, MIS, and D3P (used in our experiments) for breaking large contraction kernels are described, and it is also shown that their difference in the context of segmentation is minimal.

Note that the described grouping method is similar to the image segmentation method in (Haxhimusa and Kropatsch, 2004), which also builds a graph pyramid and uses concepts of internal/external contrast. Important differences are (presented approach vs. Haxhimusa and Kropatsch, 2004):

1. Edge weights are recomputed at every level to reflect the differences between the updated 'models' for the *whole regions* vs. always selecting a subset of the weights from the level below, which reflects the difference between vertices that are neighbors in the base level (contrast *along the boundary*).
2. The features are signals of orientation variation/transformation vs. color values.
3. The method starts from a (possibly disconnected) graph vs. from a neighborhood graph using 4 connectivity.

Please cite this article in press as: Artner, N.M., et al. Hierarchical spatio-temporal extraction of models for moving rigid parts. Pattern Recognition Lett. (2011), doi:10.1016/j.patrec.2011.05.005

The difference at 1 has the effect that a long chain of regions that differs by a constant small difference, will not be merged to create a single region (e.g. a smooth gradient over the whole image). This is very important when grouping movements with locally constant difference like for example the skin covered body.

5. Experiments

The experiments in this section are divided by the type of motion in the input videos: motion in (see Section 5.2) and out of the 2D image plane (see Section 5.3).

5.1. Parameters of the framework

In the following the parameters of the presented framework are recalled:

- ϵ_r This threshold decides if a triangle is labeled *relevant* or *separating*. It should be set considering the noise in the sequence, the tracking errors, and the local deformations in the parts (e.g. skin, cloth, material of man-made object).
- β Sets an upper bound for the internal difference in the grouping process in the graph pyramid, which decides if edges are contracted (triangles are grouped together). This parameter is important in the lower levels of the pyramid (especially in the first level) and its influence decreases in higher levels (see Eq. (7)).
- w Weights the influence of the differences in radii d_R and angles d_A on the similarity of the transformation matrices. It is only relevant for out of the plane motion.

5.2. Motion in the image plane

The videos human 1 (640 *times* 480, 860 frames) and human 2 (640 *times* 480, 1178 frames) are self-produced and show humans undergoing articulated motion in the image plane. In each video the Kanade–Lucas–Tomasi tracker (Birchfeld, 2008) is used to track corner points to supply the necessary motion information for the hierarchical grouping process.

Table 1
Parameters and results for sequences human 1 and 2 (see Section 5.2). "Ground truth" is the correct number of parts in the scene and "result" lists the outcome of the presented approach, where the numbers in brackets are (outliers/all triangles).

Sequence	ϵ_r	β	Ground truth	Result
Human 1	20	50°	7	7(0/180)
Human 2	15	40°	7	7(2/305)

Both video sequences, human 1 and 2, show a person undergoing articulated motion in the image plane. Table 1 lists the parameters used with the two videos and the results. In Fig. 10 the results of the spatio-temporal filtering are visualized. Fig. 11 shows the grouping result, where each color in the triangulation represents one "rigid" part. Figs. 12 and 13 present the grouping results in separate images.

The result for *experiment human* 1 is ideal, meaning that each "rigid" part and the background are one vertex in the top level of the graph pyramid. For *experiment human* 2 the right lower arm is represented by two top vertices. Additionally, one "rigid" part of the hair and one of the left upper arm with the background are produced. The reasons for this are: (1) the relative orientation change (angle) between two "rigid" parts is smaller than the local differences due to locally non-rigid deformation (e.g. skin) or the imprecisions of the tracker and (2) the labeling into *relevant* and *separating* has to allow certain variation (see Section 4.1). The torso is connected with the base of the chin in both sequences because during tracking the features at the base of the chin slide when the head is tilted and remain in the same position in the image, creating a *relevant* triangle.

5.3. Motion out of the image plane

The input data, videos and trajectories, for this experiments are from the benchmark used by Yan et al. (2008). All videos show articulated objects undergoing movements out of the 2D image plane (toy truck: 720×480, 31 frames; two cranes: 360×240, 61 frames; dancing: 720×480, 40 frames). Besides the different type of motion, these videos are significantly shorter than the self-produced sequences in Section 5.2. Therefore, there is less motion information and the presented approach is more prone to outliers. Table 2 summarizes the parameters used for the experiments.

Fig. 14 shows the *relevant* triangles, which are the input for the grouping process. Fig. 15 is an overview of the grouping results for all three videos, where the identified "rigid" parts are labeled with different colors.

The *experiment toy truck* successfully results in two "rigid" parts as in (Yan et al., 2008). Fig. 15(a) shows the labeled triangulation including the outliers (red triangles) and Fig. 16 the two main parts of the truck represented by their triangulation.

The presented approach correctly decomposes *experiment two cranes* in three main parts, where the crane on the left is one "rigid" part as it does not undergo articulated motion and the crane on the right is separated in two parts as its arm (boom) with the bucket moves separately (see Fig. 15(b) and Fig. 17).

Fig. 15(c) and (d), Figs. 18 and 19 present two results for *experiment human dancing*. Our approach is not able to decompose the left arm into two parts as the articulated movement is not distinc-

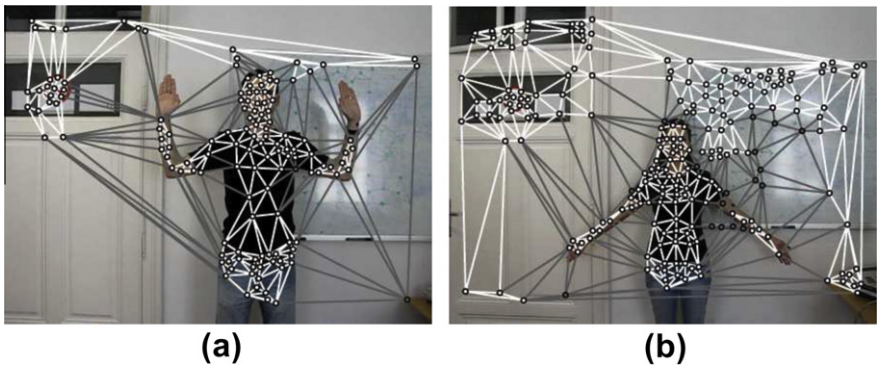

Fig. 10. Triangulation with labeling. White: relevant. Gray: separating. (a) *Human* 1. (b) *Human* 2.

Please cite this article in press as: Artner, N.M., et al. Hierarchical spatio-temporal extraction of models for moving rigid parts. Pattern Recognition Lett. (2011), doi:10.1016/j.patrec.2011.05.005

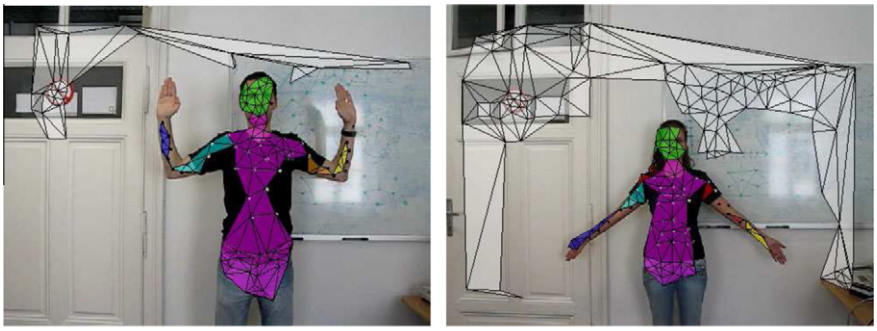

Fig. 11. Grouping result of *human* 1 (a) and *human* 2 (b). Each color represent one "rigid" part. The red triangles in (b) are outliers. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

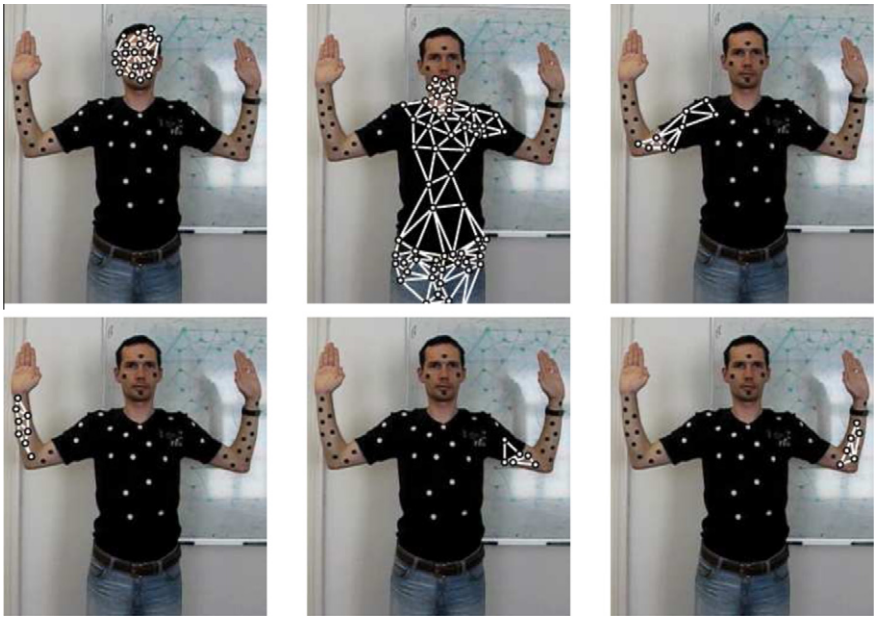

Fig. 12. Grouping result of *human* 1, where each image shows one identified part.

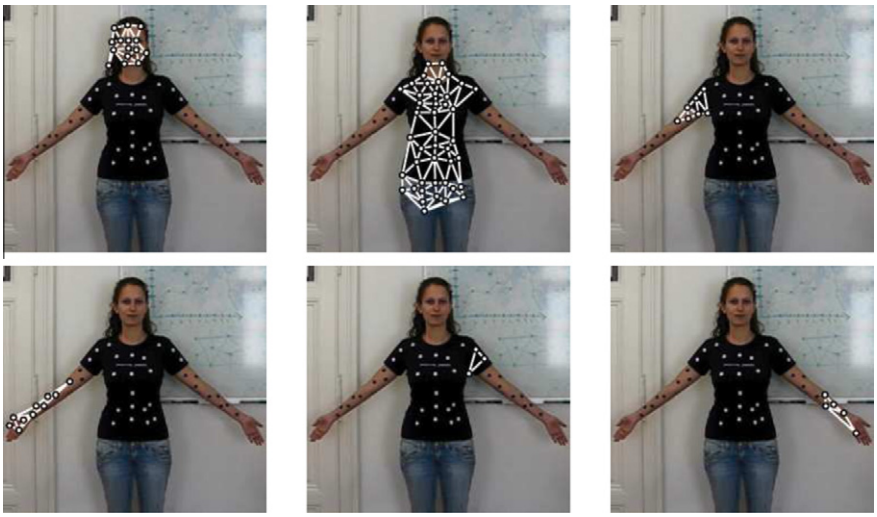

Fig. 13. Grouping result of *human* 2, where each image shows one identified part.

Please cite this article in press as: Artner, N.M., et al. Hierarchical spatio-temporal extraction of models for moving rigid parts. Pattern Recognition Lett. (2011), doi:10.1016/j.patrec.2011.05.005

Table 2
Parameters and results with videos out of the image plane (see Section 5.3). w is the weight of Eq. (5). "Ground truth" is the correct number of parts in the scene and "result" lists the outcome of the presented approach, where the numbers in brackets are (outliers/all triangles).

Sequence	ϵ_r	β	w	Ground truth	Result
Toy truck	20	1.3	1.0	2	2 (9/147)
Two cranes	20	0.4	0.9	3	3 (11/134)
Human dancing 1	20	0.4	0.6	6 (4)	4 (25/366)
Human dancing 2	20	0.5	1.0	6 (4)	5 (14/366)

tive enough. Yan et al. (2008) are able to separate the object in six parts. The best result with our approach is shown in Fig. 18 *human dancing* 1, where the object is separated in four parts Fig. 15(c)). Fig. 19 shows a result with five parts *human dancing* 2, but we prefer the first result considering the symmetry of the human body.

5.4. Discussion

The presented method extracts a model for the "rigid" parts of a scene in an unsupervised manner. It uses no prior knowledge (no

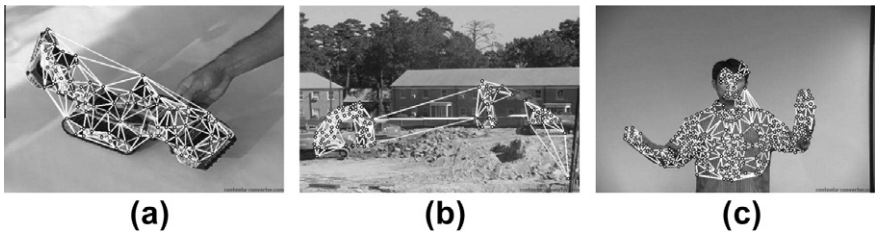

Fig. 14. Input for out of the plane experiments (triangles labeled *relevant*). (a) *Toy truck*. (b) *Two cranes*. (c) *Human dancing*.

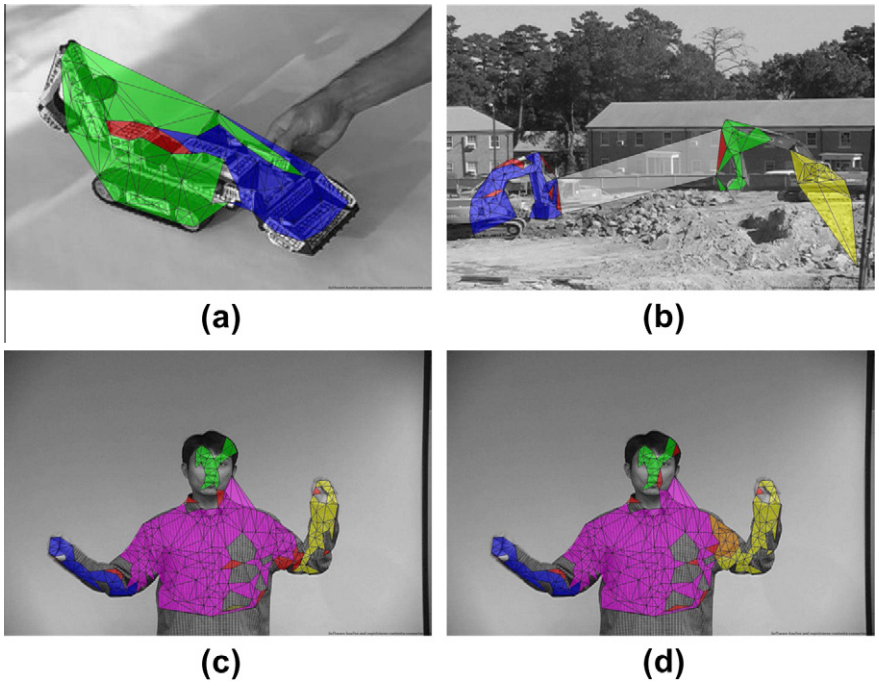

Fig. 15. Grouping results of experiments *toy truck* (a), *two cranes* (b), and *human dancing* 1 (c) and *human dancing* 2 (d). Each part is labeled with a color and all outliers are colored red. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

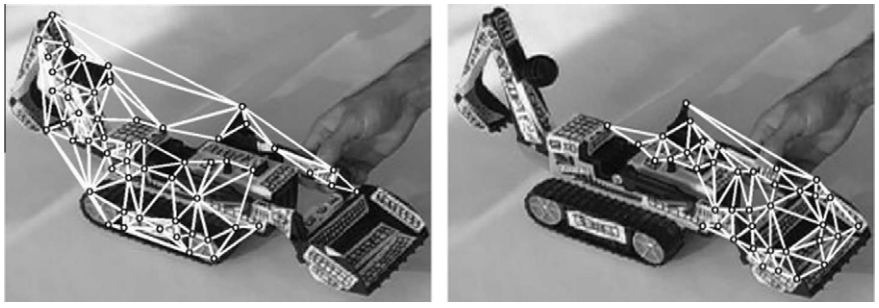

Fig. 16. The two main parts of the grouping process of *toy truck*.

Please cite this article in press as: Artner, N.M., et al. Hierarchical spatio-temporal extraction of models for moving rigid parts. Pattern Recognition Lett. (2011), doi:10.1016/j.patrec.2011.05.005

Fig. 17. Grouping of *two cranes* into three parts.

Fig. 18. Grouping of *human dancing* 1 in four parts.

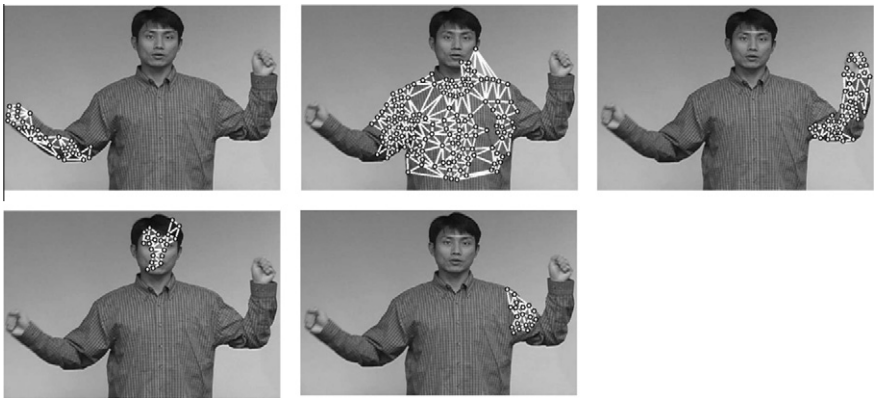

Fig. 19. Grouping of *human dancing* 2 in five parts.

training, no knowledge about the scene and its objects), it can deal with motion in and out of the 2D image plane and it can be applied to videos with any arbitrary articulated or rigid object (i.e. human, finger, animal, basket ball...).

The spatio-temporal filtering (Section 4.1) will correctly identify the triangles belonging to "rigid" parts, if the movement of the parts relative to each other (distance variation) is larger than the local distance variation between neighboring features of the same part.

The quality of the results depends on the quality of the input data, the trajectories of the features. If the trajectories are reliable and at least three features exist for each "rigid" part, the proposed framework is capable of decomposing the object. Only parts undergoing significant articulated movement different from the other parts in the object can be identified (e.g. if an arm never bends it will not be decomposed in two parts, but one part). The hierarchical grouping process is able to identify the "rigid" parts if the difference in the movement of triangles belonging to different parts is larger than the local differences due to locally non-rigid deformation (e.g. skin) or to imprecisions of the computed feature positions.

6. Conclusion

This paper presented a graph-based approach to extract a model of the "rigid" parts of articulated objects. The input consists of a video and the trajectories of corresponding salient features. A triangulated graph is used to represent the features and their spatial relationships over time. First a spatio-temporal filtering is performed which labels the triangles in the graph as *relevant* or *separating*. The *relevant* triangles are given as input to a hierarchical grouping process which identifies the "rigid" parts in the scene considering the motion of the triangles. Depending on the motion, a suitable representation is the orientation variation of the triangles or their transformation matrices. The hierarchical grouping is realized by building a graph pyramid, where the grouping criterion decides which triangles are grouped together and the vertices in the top level represent "rigid" parts. Promising experimental results show the potential of the approach.

Acknowledgments

This work was partially supported by the Austrian Science Fund under Grants P18716-N13 and S9103-N13. Adrian Ion was supported in part by the European Commission, under project MCEXT-025481.

References

Aggarwal, J.K., Cai, Q., 1999. Human motion analysis: A review. CVIU 73 (3), 428–440.

Alatan, A.A., Onural, L., Wollborn, M., Mech, R., Tuncel, E., Sikora, T., 1998. Image sequence analysis for emerging interactive multimedia services. Circuits Systems Video Technol. 8 (7), 802–813.

Altunbasak, Y., Eren, P.E., Tekalp, A.M., 1998. Region-based parametric motion segmentation using color information. Graphical Models Image Process. 60 (1), 13–23.

Artner, N.M., Ion, A., Kropatsch, W.G., 2009. Rigid part decomposition in a graph pyramid. In: Eduardo Bayro-Corrochano, J.O.E. (Ed.), The 14th Iberoamerican Congress on Pattern Recognition, LNCS, vol. 5856. Springer, pp. 758–765.

Artner, N.M., Ion, A., Kropatsch, W.G., 2011. Multi-scale 2d tracking of articulated objects using hierarchical spring systems. Pattern Recognition 44 (4), 800–810.

Birchfeld, S., 2008. Klt: An implementation of the kanade-lucas-tomasi feature tracker. http://www.ces.clemson.edu/stb/klt/ (04.08).

Please cite this article in press as: Artner, N.M., et al. Hierarchical spatio-temporal extraction of models for moving rigid parts. Pattern Recognition Lett. (2011), doi:10.1016/j.patrec.2011.05.005

Castagno, R., Ebrahimi, T., Kunt, M., 1998. Video segmentation based on multiple features for interactive multimedia applications. Circuits Systems Video Technol. 8 (5), 562–571. doi:10.1109/76.718503.

Celasun, I., Tekalp, A.M., Gokcetekin, M.H., Harmanci, D.M., 2001. 2-d mesh-based video object segmentation and tracking with occlusion resolution. Signal Process. Image Comm. 16 (10), 949–962.

Chen, H.-T., Liu, T.-L., Fuh, C.-S., 2006. Segmenting highly articulated video objects with weak-prior random forests. In: ECCV. Springer, Graz, Austria, pp. 373–385.

Conte, D., Foggia, P., Jolion, J.-M., Vento, M., 2005. Graph-Based Representations in Pattern Recognition. Springer, Berlin/ Heidelberg (Chapter: A Graph-Based, Multi-Resolution Algorithm for Tracking), pp. 193–202.

Costeira, J.P., Kanade, T., 1998. A multibody factorization method for independently moving objects. Internat. J. Comput. Vision 29 (3), 159–179.

Drouin, S., Hébert, P., Parizeau, M., 2008. Incremental discovery of object parts in video sequences. Comput. Vision Image Understanding 110, 60–74.

Felzenszwalb, P., Huttenlocher, D., 2005. Pictorial structures for object recognition. IJCV 61 (1), 55–79.

Gavrila, D.M., 1999. The visual analysis of human movement: A survey. CVIU 73 (1), 82–980.

Godec, M., Leistner, C., Saffari, A., Bischof, H., 2010. On-line random naive bayes for tracking. In: ICPR, IEEE, pp. 3545–3548.

Haxhimusa, Y., Kropatsch, W.G., 2004. Segmentation graph hierarchies. In: Fred, A.L.N., Caelli, T., Duin, R.P.W., Campilho, A.C., de Ridder, D. (Eds.), SSPR/SPR, Lecture Notes in Computer Science, vol. 3138. Springer, pp. 343–351.

Klein, E., 1939. Elementary Mathematics from an Advanced Standpoint: Geometry. MacMillan, New York.

Kropatsch, W.G., 1995. Building irregular pyramids by dual graph contraction. Vision, Image Signal Process. 142 (6), 366–374.

Kropatsch, W.G., Haxhimusa, Y., Pizlo, Z., Langs, G., 2005. Vision pyramids that do not grow too high. PRL 26 (3), 319–337.

Kropatsch, W.G., Haxhimusa, Y., Ion, A., 2007. Applied Graph Theory in Computer Vision and Pattern Recognition. Studies in Computational Intelligence, vol. 52. Springer (Chapter: Multiresolution Image Segmentations in Graph Pyramids), pp. 3–42.

Lauer, F., Schnör, C., 2010. Spectral clustering of linear subspaces for motion segmentation. In: ICCV, IEEE, pp. 678–685.

Li, H., Lin, W., Tye, B., Ong, E., Ko, C., 2001. Object segmentation with affine motion similarity measure. Multimedia Expo, 841–844.

Moeslund, T.B., Hilton, A., Krger, V., 2006. A survey of advances in vision-based human motion capture and analysis. CVIU 104 (2–3), 90–126.

Nesetril, J., Milková, E., Nesetrilová, H., 2001. Otakar boruvka on minimum spanning tree problem translation of both the 1926 papers, comments, history. Discrete Math. 233 (1–3), 3–36.

Nordberg, K., Zografos, V., 2010. Multibody motion segmentation using the geometry of 6 points in 2d images. In: ICPR. IEEE, Istanbul, pp. 1783–1787.

Tekalp, A., Van Beek, P., Toklu, C., Gunsel, B., 1998. Two-dimensional mesh-based visual-object representation for interactive synthetic/natural digital video. Proc. IEEE 86 (6), 1029–1051.

Tuceryan, M., Chorzempa, T., 1991. Relative sensitivity of a family of closest-point graphs in computer vision applications. Pattern Recognition 24 (5), 361–373.

Walther, T., Würtz, R.P., 2008. Learning to look at humans – what are the parts of a moving body? In: Articulated Motion and Deformable Objects, pp. 22–31.

Walther, T., Würtz, R.P., 2009. Unsupervised learning of human body parts from video footage. In: 2nd Workshop on Non-Rigid Shape Analysis and Deformable Image Alignment, pp. 336–343.

Yan, J., Pollefeys, M., 2008. A factorization-based approach for articulated nonrigid shape, motion and kinematic chain recovery from video. IEEE Trans. Pattern Anal. Machine Intell. 30 (5), 865–877.

Yilmaz, A., Javed, O., Shah, M., 2006. Object tracking: A survey. ACM Comput. Surv. 38 (4).

Please cite this article in press as: Artner, N.M., et al. Hierarchical spatio-temporal extraction of models for moving rigid parts. Pattern Recognition Lett. (2011), doi:10.1016/j.patrec.2011.05.005