Topological Reconstruction of Occluded Objects in Video Sequences

Vincent Agnus and Christian Ronse

Laboratoire des Sciences de l'Image, de l'Informatique et de la Télédétection
(UMR 7005 CNRS–ULP),
67400 Illkirch, France
{`agnus,ronse`}`@dpt-info.u-strasbg.fr`
`http://lsiit.u-strasbg.fr/`

Abstract. In [1,2] we have introduced a new approach for the spatio-temporal segmentation of image sequences. Here a 2D+t sequence is considered as a 3D image, and 2D objects moving in time (or following a given motion model) are segmented as 3D objects with the use of connected morphological filters, and are represented as spatio-temporal flat zones. However when an object undergoes occlusion by another in the sequence, their 3D trajectories intersect, and the spatio-temporal segmentation will fuse the two objects into a single flat zone. In this paper we introduce a method for separating occluded objects in spatio-temporal segmentation. It is based on a study of the changes of topology of the temporal sections of a flat zone. A topologically constrained watershed algorithm allows to separate the objects involved in the occlusion.

1 Introduction

The goal of segmentation is to partition an image into regions having the same attributes (color, contrast, texture, etc.). In the case of 2D+t image sequences, attributes for segmentation are generally contrast and/or motion. In [1,2] we introduced a new morphological approach enabling the segmentation of 2D+t regions following a given (local) spatio-temporal translation model. The method also provides a natural framework for tracking objects and, using an extension of the method, for classifying motions. Other morphological approaches [4,5] produce a 2D spatial segmentation and try to preserve space-time consistency by propagating markers in the future and applying a new spatial segmentation. (see [1,2] for a detailed discussion of other methods, morphological or based on optical flow). Our 2D+t morphological approach avoids the computation of the optical flow and the control of marker propagation in the future. It naturally handles the case of new object emerging in the sequence or objects leaving the scene. However our method encounters a problem: the occlusion between objects produces a unique flat zone covering many different objects. In this paper we present a new method of topological reconstruction in order to split these flat zones and seperate the different objects.

A. Braquelaire, J.-O. Lachaud, and A. Vialard (Eds.): DGCI 2002, LNCS 2301, pp. 155–164, 2002.

In Section 2 we recall briefly the spatio-temporal segmentation method of [1,2]. Section 3 introduces our topological reconstruction approach for dealing with occlusions. First we explain how occlusions lead to the fusion of the flat zones corresponding to the different objects. Then we outline the decomposition of spatio-temporal objects into *connected blocks*, which fall into three classes: *single-objects*, *junctions* and *disjunctions*. Single-objects become markers for a modified 3D watershed algorithm, which takes into account the topological relations between 2D time sections, in order to separate the different objects involved in an occlusion. Finally we discuss the oversegmentation and the ways to reduce it. Experimental results are presented in Section 4, where we give the conlusion.

2 Segmentation Using Motion Information

Here we summarize our spatio-temporal segmentation method given in [1,2]. Considering a sequence as a single 3-dimensional image, an erosion in the temporal direction modifies the original sequence only at motion boundaries. We can use this information to find the moving objects. From these boundaries we reconstruct geodesically [7] the original signal, and this produces flat zones (maximal connected regions where grey level is constant) corresponding to moving objects. With this method we can avoid choosing the structuring element size to create flat zones, which was required in [6]. This reconstruction is performed both for dark and light objects in the scene, using duality in mathematical morphology. From these two reconstructed images, we allocate a region marker for each flat zone, keeping the one which is farthest from the original signal. This processing can produce false detections in some cases. They are removed by our motion filters.

We now describe briefly the construction of flat zones for moving objects. Objects undergoing a typical motion are reconstructed from their motion boundaries. These are computed by taking the difference between the original sequence S and its erosion in the temporal direction. The shape of the temporal structuring element allows to detect a given motion model. This difference sequence is filtered by grey-level thresholding and area opening on connected components, to obtain a binary sequence S_{mb} which represents only significative motion borders. We notice that these connected components are inside light moving objects. From these components, a new sequence S_m is built, for which we give in each S_{mb} connected component the minimum of its grey-levels from the original sequence. The geodesic reconstruction of the orginal sequence by dilation of the S_m sequence produces flat zones at moving object locations. These large flat zones are used to determine moving objects. The generation of flat zones is applied for both bright and dark moving objects, using the principle of morphological duality. An election stage is performed in order to keep the most significant flat zones among bright and dark objects. This new sequence is filtered out by studying the shape of flat zones in space-time. We remove those that do not undergo a given motion model. The remaining ones receive a unique label to differentiate the objects in the scene (sequence S_o named *object*). The flat zones produced belong here to space-time, so we have an object tracking across time, and a boundary localization at each time.

However our method from [1,2] does not deal with occlusions: occluded objects are fused with occluding ones when they are both lighter (or darker) than the background. We solve this problem in the next section.

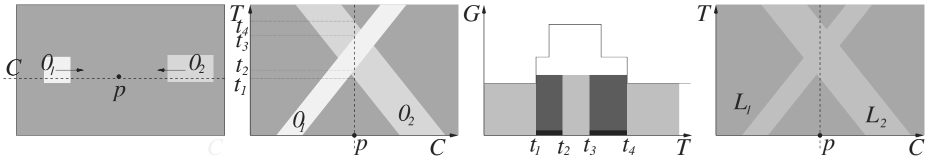
3 Topological Reconstruction

In some cases, occlusions cause the creation of the same flat zone for different objects in the scene. This drawback comes from our construction method. We name these flat zones *multi-objects*. The real objects (named *single-objects*) belonging to the same flat zone can be found by studying the topological structure of the multi-objects in space-time.

In a first part we describe when the creation of multi-objects occurs, and in a second part we give our method of topological reconstruction. It can determine a single-object in each branch of a multi-object. Branches which don't contain a single-object are filled with the labels of the other single-objects. This operation is performed by using a spatio-temporal watershed algorithm with both grey-level and topological changes as hierarchical propagation priorities.

3.1 Generation of Multi-objects

The sequence S_m, from which flat zones are reconstructed, is computed by keeping the minima of the grey-level of the orginal sequence in each connected component of the filtered difference sequence S_d. So a flat zone recovers only the darkest object at occlusion places. Fig.1 shows the flat zones generated by our method in the case of two crossing bright objects. The sequence S_m, shown as two dark rectangles in Fig.1.*c*, is computed by keeping the minima of each connected component of the motion border. So only one flat zone is constructed for the two bright objects. This flat zone L_2 is a multi-object.



a) *b*) *c*) *d*)

Fig. 1. Generation of one flat zone for several objects: *a*) grey-level sequence, *b*) spatio-temporal representation along cross section C, *c*) grey level at point p, *d*) flat zones produced

To differentiate the objects in this flat zone, we suppose that objects cannot be ubiquitous, so the multi-object shape allows us to determine the number of single-objects within it. We apply our topological reconstruction method, see Fig.2. From the sequence object S_o we decompose each spatio-temporal flat zone into regions where topology is preserved in some sense (deepened below). The study of adjacency between these regions allows to classify them as representing an object or an occlusion place. The latter are filled with the labels from regions classified as single-object. So we can allocate a label for each different object contained in a unique flat zone from the sequence object.

In a first part we describe the framework of the detection of objects in flat zones, this step is named *connected block decomposition*. We give also an algorithm to build them. The propagation of object labels at occlusions places is described in Section 3.4, where we use a modified version of the Salembier's region growing algorithm [6]. To overcome the oversegmentation due to the generation of labels we define a region adjacency graph contraction to reduce the number of regions.

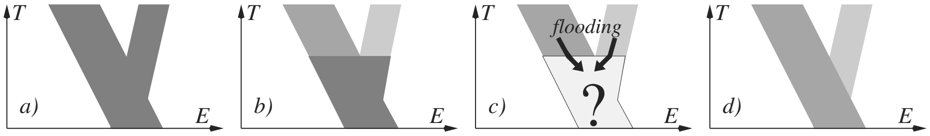


Fig. 2. Topological reconstruction step: a) the sequence object S_o, b) topological invariant region decomposition, c) removal of occlusions zones, d) relabeling using watershed

3.2 Connectivity Block Decomposition

The study of topological changes (i.e., of neigbourhoods of spatial connected components in a multi-objet) can give information about the number of single-objects in the multi-object. For example, consider a multi-object with a "Y" shape (Fig.2); the temporal dimension is along the vertical axis. We can determine that this object contains at least 2 different objects. Indeed the two upper branches point to the presence of two different objects. The base of the "Y" is the place of the occlusion between these objects. We now develop a categorization of multi-objects branches. For this we decompose the multi-object into *connected blocks*. They are connected components which don't contain a local topological change in the multi-object. A distinction is made within these blocks: those which contain a single-object and the other ones. The blocks of a same object which have more than two adjacent connected blocks in the past or future are considered as blocks that do not contain a single-object. It is the place where the objects move away or approach. We give a more formal definition of connected blocks and a method to build them.

Definition 1. *The temporal section R_t of region $R \subset E \times T$ at time t is the set $R_t = R \cap (E \times \{t\})$.*

Definition 2. *A subset X of region $R \subset E \times T$ is called a* connected block *of R iff it is a maximal subset of R such that: a) $X \neq \emptyset$ and X connected, b) for all times, the temporal sections X_t are connected (spatially, because they belong to E) and c) a temporal section X_{t-1} or X_{t+1} of X which is adjacent to X_t, may not be adjacent to $R_t \setminus X_t$.*

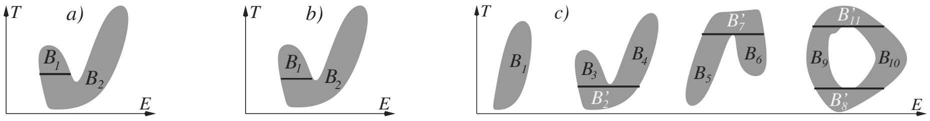


Fig. 3. Examples of connected blocks on spatio-temporal objects: *a*) and *b*) two counter-examples, *c*) connected blocks for many configurations, where bright letters represent single-objects

There is some similarity between our decomposition and *Reeb Graphs* [3]. Given a smooth function f defined on a smooth surface M, the *Reeb Graph* of f associates to every connected component of a level set $f^{-1}(t)$ a point of height t. Smoothly evolving level curves give then rise to arcs in the graph and critical points of the surface produce junction nodes in the graph. In the case of image sequences, the surface M would consist of the spatiotemporal contours of the sequence, and the function f would be time. Then the Reeb Graph would code the evolution of object spatial contours at different times. Our representation is simpler, in that we only consider connected components of objects, while we ignore inner contours and holes.

3.3 Connected Block Decomposition Algorithm

To obtain this decomposition, we build a label sequence named S_{topo} from the begin to the end of the sequence, in a recursive scheme. We label the spatial connected components along time using the labels of connected components already labeled in the past. We use three arrays of labels:

- $ARRNSO$ which contains the labels whose connected component descended from a junction (2 spatial connected components in past) or fathering a disjunction (2 in future). This array permits us to know the connected blocks to remove, i.e. the labels not representing a single-object.
- $ARRP[L_t]$ which contains the set of spatial connected component labels at time $t-1$ which are contiguous to the connected component at time t labeled L_t.
- $ARRF[L_{t-1}]$ which contain the set of spatial connected component labels at time t which are contiguous to the connected component at time $t-1$ labeled L_{t-1}.

We have: $L \in ARRP[L']$ iff $L' \in ARRF[L]$. The labeling is built as follows:
At $t = 0$, a spatial labeling is performed in the time section $t = 0$. To go to step $t + 1$: a temporay labeling of spatial connected components of S_o in time section $t + 1$ is carried out. The new labels are partially absorbed by the labels of previous time sections already processed. We keep the new label if a birth or topological change occurs. For this we use the array $ARRP[]$ and $ARRF[]$. For each new label L' (at time $t + 1$), we have to deal with several cases:

- if $ARRP[L']$ is empty, a new object appears at time $t+1$; it is a birth. This new label is preserved for the next time section.
- if $ARRP[L']$ contains at least two labels: it is a junction. The label L' is inserted in $ARRNSO$ because this label indicates a topological change. This label represents the birth of a new connected block.
- if $ARRP[L'] = \{L\}$, we have to deal with two cases:
  - if $ARRF[L] = \{L'\}$, two connected components are neighbours in time. They belong to the same branch of the multi-object (i.e in the same connected block). The new label L' take the value of the label L.
  - if $ARRF[L]$ contains at least two labels, there is a disjunction at time t. The label L' is preserved due to the topological change and the creation of a new connected block and the label L is inserted in $ARRNSO$.

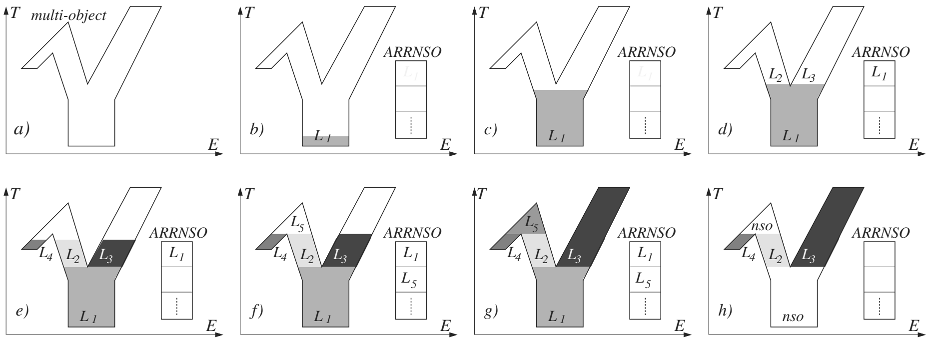


Fig. 4. Construction of S_{topo} and S_{so}: *a*) sequence object S_o, *b*) birth of a new connected block, *c*) contruction of the connected block, *d*) topological change (disjunction, two new connected blocks), *e*) a new connected block born, *f*) topological change (junction), *g*) sequence S_{topo}, *h*) sequence S_{so} (connected blocks of S_{topo} containing only single-object)

Fig.4 represents the application of this algorithm in a general scheme and Fig.5 illustrates the local count of neighbouring spatial connected components between two succesive time sections.

Now we have a marker only for each single-object in each multi-object. We start from the object sequence S_o, it is decomposed into connected blocks (sequence S_{topo}) as described above. Connected blocks neighbour in past to a junc-

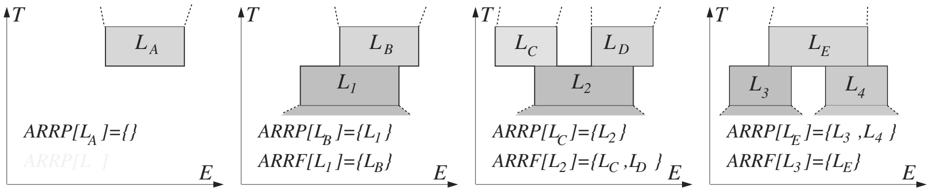


Fig. 5. Connected component count: birth, same connected component, disjunction, junction

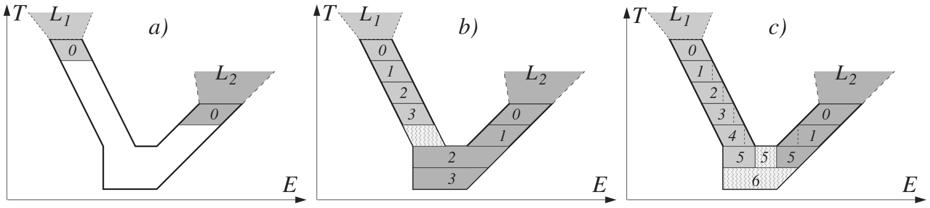


Fig. 6. Single vs. Intensity and topological hierarchy: *a*) start of the flooding of a multi-object with two labels, *b*) grey-level propagation, *c*) topological and grey level propagation

tion or in future to a disjunction are removed, they are not single-objects (sequence S_{so}). Single-object labels are propagated in their respective multi-object to fill the occlusions section. The propagation is based on an extension of watershed using grey-level and topological information, described below.

3.4 Topological Watershed Algorithm

The occlusion regions have been removed during our topological labeling. To reconstruct the segmentation at these places, we use a modified version of the watershed region growing defined by Salembier [6]. We use grey-level information of the source sequence to allow the tracking of objects across time, and the topological change in multi-objects to synchronize the propagation stage. We must use a double hierarchy (grey-level and topology) to ensure the good propogation of labels in space-time. Indeed, consider Fig.6, we have a multi-object with two labels L_1 and L_2 of single-objects. The number in each spatial connected component corresponds to the time step in the flooding process. If we use only grey-level information, the L_2 label reaches the disjunction before L_1, so at this time section the space flooding is performed whitout taking into account the label L_1. To avoid this drawback, before flooding a time section we synchronize the labels: the labels are propagated normally if and only if they belong to the same connected block. Labels which cross the border of a new connected block are temporarily stopped. When all labels are stopped (i.e they all belong to a connected block border) the flooding process restarts normally. This approach allows the labels to wait for others before flooding in space at a time section.

To implement this propagation scheme, the use of ordered queues (OQ) is natural to manage this double hierarchy. We use two OQs: OQ_G and OQ_T. The first one is used to flood the time sections and the other one is used to store labels which are stopped. The hierarchy within these OQs is the similarity between grey-levels of neighbouring pixels. When OQ_G is empty (i.e., all pixels are stopped) and all occlusion places are not flooded, we swap these two OQs, so that the flooding process can continue with labels synchronized. For instance in Fig.6 the labels are progated using OQ_G, and when L_2 reaches the disjunction time section (step 1), it is stopped and stored in OQ_T. The label L_1 progresses via OQ_G until it reaches the border between connected blocks (step 4). It is stopped and stored in OQ_T. The queue OQ_G is now empty: all labels belong to a border of topological change. We swap the two OQs and the flooding stage restarts (steps 5 and 6).

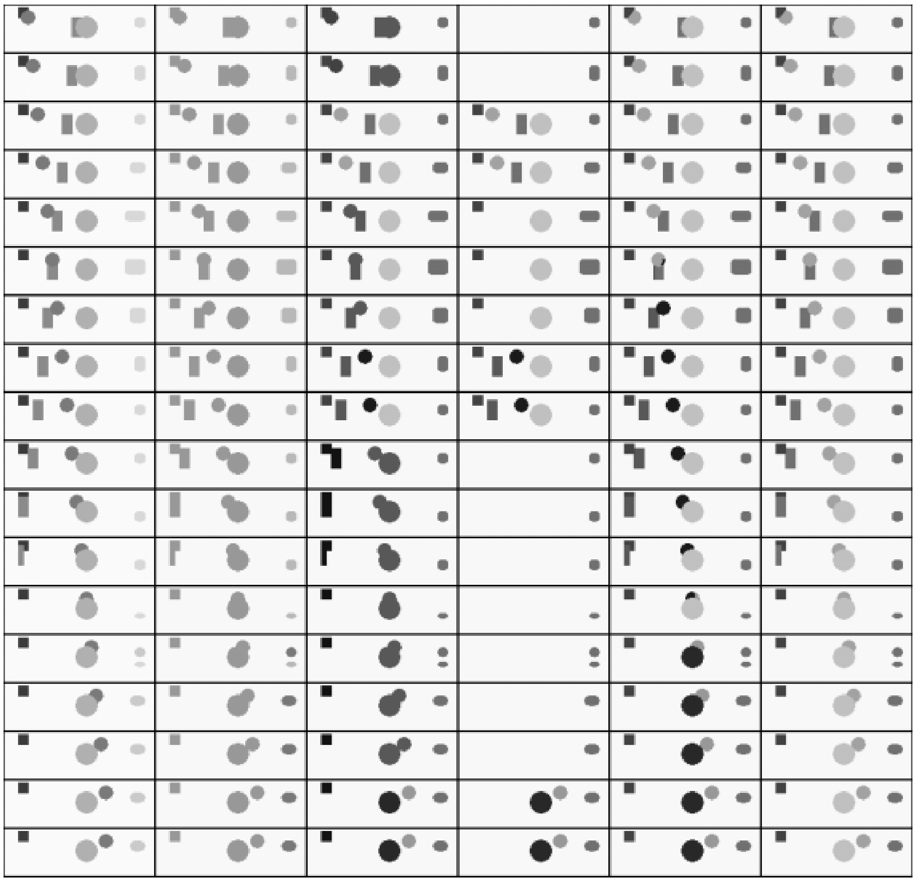
We have described how to find objects in a multi-object and how to recover their shape in space-time, but some drawbacks remain. The search for an object, in some cases, generates an over-labeling. We describe in the next section when these cases occur, and how to deal with them.

3.5 Reduction of Oversegmentation

The construction of connected blocks strongly depends on the topology of the multi-object. In two cases, too many connected blocks are constructed. The first one is due to topological changes arising from small noisy objects, the second one depends on our connected block decomposition. For instance, when two objects cross each other, their trajectories form together a “X” shape, our block decomposition gives four labels B_1 to B_4 and an occlusion place C, corresponding to the four branches and the crossing in the “X” shape. We must keep all B labels because we don’t know if they represent the same object before and after the occlusion, indeed some objects can hide other ones which can appear after the occlusion.

To avoid the influence of topological changes due to small deformations, we pre-filter the object sequence S_o by removing with an area opening all small spatial connected components in each time section. Block connected decomposition is made on the filtered sequence and the watershed labeling stage is processed on the object sequence using markers obtained with the filtered object sequence.

To overcome the over-labeling inherent to some multi-object shapes in space-time, we have to use more global information; we use the contraction of a region adjacency graph. We define a model $\mathcal{M}(R)$ for each region R. It contains information about it. We use mean grey-level $\mathcal{M}.\bar{g}$, presence time (birth $\mathcal{M}.t_s$ and death $\mathcal{M}.t_e$), and multi-object membership $\mathcal{M}.object$ and size (i.e., the volume in space-time $\mathcal{M}.N$) to characterise them. An order $\mathcal{O}(R_1, R_2)$ is computed to determine the priority of merging regions. The region merging stage runs as follow: the two regions of the pair with the lowest merging order are merged together, the model $\mathcal{M}$ of the union is reestimated, and the merging order is recomputed with the neighbouring regions.



a) *b*) *c*) *d*) *e*) *f*)

Fig. 7. Application example: column *a*) Original grey-level sequence, *b*) object sequence, *c*) connected blocks, *d*) single-objects, *e*) topological watershed, *f*) regions merged

When regions R_1 and R_2 merge, the new model $\mathcal{M}(R_1 \cup R_2)$ is:
— $\mathcal{M}.\overline{g} \leftarrow (N_1\overline{g_1} + N_2\overline{g_2})/(N_1 + N_2)$, $\mathcal{M}.N \leftarrow N_1 + N_2$,
— $\mathcal{M}.t_s \leftarrow \min\big(\mathcal{M}(R_1).t_s, \mathcal{M}(R_1).t_s\big)$ and $\mathcal{M}.t_e \leftarrow \max\big(\mathcal{M}(R_1).t_e, \mathcal{M}(R_1).t_e\big)$,
— $\mathcal{M}.object \leftarrow \mathcal{M}(R_1).object$ (label of S_o),
where $N_i = \mathcal{M}(R_i).N$ and $g_i = \mathcal{M}(R_i).\overline{g}$. The merging order is defined as follow:

$$\mathcal{O}(R_1, R_2) = \big[N_1(\overline{g_1} - \overline{g}_{R_1 \cup R_2})^2 + N_2(\overline{g_1} - \overline{g}_{R_1 \cup R_2})^2\big] \times \left[\begin{array}{l} \big(N_1 \mathit{\Delta}_t(R_1 \cup R_2)/\mathit{\Delta}_t(R_1) - N_1 - N_2\big)^2 \\ +\big(N_2 \mathit{\Delta}_t(R_1 \cup R_2)/\mathit{\Delta}_t(R_2) - N_1 - N_2\big)^2 \end{array}\right],$$

where $\mathit{\Delta}_t(R_i) = \mathcal{M}(R_i).t_e - \mathcal{M}(R_i).t_s + 1$ is the region presence time. This term measures the error when two regions merge. It is the product of two factors. The first one expresses the grey level difference between regions, and the second one compares the region spatial mean size.

In some case, we don't allow fusion (in putting the merging order to infinity). Regions which belong to different multi-objects are not merged. In the same way, regions which share a same presence time cannot be merged because they are not ubiquitous.

4 Results and Conclusion

Fig.7 show the step of our topological reconstruction. From the original sequence Fig.7.*a*, flat zones are produced by our approach[1,2] (Fig.7.*b*). It contains three multi-objects. Fig.7.*c* represents the decomposition into connected blocks, and single-objects blocks are shown in Fig.7.*d*. The labels are propagated in their respective multi-objects (Fig.7.*e*) using our double hierarchical watershed algorithm. The over labeling is reduced with our adjacency region graph contraction method (Fig.7.*f*). An animated version of Fig.7, as well as other pratical applications (in particular, results on standard motion sequences) can be found at URL `http://arthur.u-strasbg.fr/~agnus/DGCI2002/` .

The flat zone generation and topological reconstruction are obtained with low-level morphological operators, that are well suited for fast implementations, thanks to optimal implementations which access each pixel only once. This contrasts with optical flow based methods, which require repeated iterations.

References

1. V. Agnus, C. Ronse, and F. Heitz. Segmentation spatio-temporelle de séquences d'images. In "*12ème Congrès Francophone Reconnaissance des Formes et Intelligence Artificielle*", volume 1, pp. 619–627, Paris, France, Feb. 2000.
2. V. Agnus, C. Ronse, and F. Heitz. Spatio-temporal segmentation using 3d morphological tools. In *15th International Conference on Pattern Recognition*, volume 3, pp. 885–888, Barcelona, Spain, Sep. 2000.
3. S. Biasotti, B. Falcidieno, and M. Spagnuolo. Extended Reeb Graphs for Surface Understanding and Description In *9th Discrete Geometry for Computer Imagery Conference*, LNCS, Springer Verlag, pp. 185-197, Uppsala, 2000.
4. B. Marcotegui and F. Meyer. Morphological segmentation of images sequences. *Mathematical Morphology and its Applications to Image Processing*, J. Serra & P. Soille Eds., pp. 101–108, 1994.
5. F. Marqués. Temporal stability in sequence segmentation using the watershed algorithm. *Mathematical Morphology and its Applications to Image Processing*, P. Maragos & R. W. Schafer & M. A. Butt Eds., pp. 321–328, 1996.
6. P. Salembier, P. Brigger, J. R. Casas, and M. Pardás. Morphological operators for image and video compression. *IEEE Trans. on Image Proc.*, 5(6): pp. 881–898, June 1996.
7. L. Vincent. Morphological grayscale reconstruction in image analysis: applications and efficient algorithms. *IEEE Trans. on Image Proc.*, vol. 2, pp. 176–201, Apr. 1993.