# Establishment Shot Detection Using Qualitative Motion

Loong Fah Cheong, Yong Wang and Hee Lin Wang

Department of Electrical and Computer Engineering
National University of Singapore, Singapore
{ eleclf , engp1827, engp1622}nus.edu.sg

## Abstract

*In this paper, we developed a technique to detect a special kind of shot, namely establishment shot, which is used to introduce a new scene to the audience or remind the audience of a known environment. The detection of establishment shot aids in various middle-level issues such as analyzing the story units of a movie. An establishment shot is usually realized by using camera motions such as panning, tilting or lateral translation. As it possesses similar motion features to those of another kind of shot called object-tracking shot, the two can be easily confused. An object-tracking shot is important for indexing on its own; it keeps the target of interest in the field of view. In this paper, we developed a technique to differentiate these two kinds of shots. The technique developed does not require accurate egomotion estimation, and can handle arbitrary camera movement including zoom as well as multiple independent motions.*

## 1. Introduction

Motion is a rich information source that can potentially extend the limited capabilities of current solutions in identifying content. However, the use of motion in multimedia nowadays has been restricted to the extraction of a few simple types of camera motion, such as pan, tilt and zoom [1]. Complete recovery of the egomotion parameters is a notoriously ill-posed problem [2, 3, 4]. This, coupled with the view that the egomotion parameters must be computed before any of the motion-based competence can be accomplished, has seriously obscured the potential for using motion directly for multimedia applications.

Recent research in computational vision emphasizes the fact that, often, appropriate spatio-temporal representations, that are directly relevant to the tasks at hand, can be computed from the imagery without going through the ill-posed process of egomotion computation. Thus instead of one strict hierarchy, a variety of visual processes, computed in parallel and using motion features of varying amount of complexity, may be constructed. Our research efforts subscribe to this philosophy. We argue that through a variety of such motion competencies each having different computational requirement, a multi-faceted usage of motion information for video indexing can be effected.

In particular, we suggest that from the point of view of a producer, the video is purposive, in that it is trying to convey the story content in a script to the viewer. In order to tell a story, a number of tasks need to be routinely accomplished by the cameraman or the producer. Some of these tasks include the introduction to a scene, the unfolding of an event, tracking of an object and the directing of attention. Often, motion is needed to accomplish the aforementioned tasks. In this paper, we focus on recovering, via motion cues, what are known as establishment shots; some of the object tracking shots are also recovered as a by-product. These two types of shots often result in similar camera motion and this paper develops a technique to distinguish the two. The technique we developed is qualitative in the sense that accurate camera rotation and translation estimates are not needed; it is also general in that it can deal with arbitrary camera movement, including zoom component as well as multiple independent motions. From the shots thus extracted, indexes can be formed based on the intentions of these shots often resulting in high level indexes.

The rest of this paper is organized as follows. In section 2, we discuss the semantics and the motion attributes of both the establishment shot and the object tracking shot. In section 3, we introduce the method of global motion estimation. In section 4, we put forth a method of detecting establishment shot. Section 5 deals with the problem of differentiating object-tracking shots from establishment shots. Section 6 presents experiment results and we conclude this paper with section 7.

## 2. Establishment shot vs object tracking shot

In movie-making industry, it is known that it is best if every story unit starts with an *establishment shot* or a *re-establishment shot*. Establishment shots have the intention of bringing the audience to a new environment while re-establishment shots have the intention of reminding the audience of the known environment. Normally establishment shot has a longer duration to acquaint the audience with the new environment whereas re-

establishment shot tends to be shorter because the purpose is to remind the audience about this environment, which has been visited before. For the purpose of this paper, we content ourselves with lumping both establishment shot and re-establishment shot together in one category and call them establishment shot.

Establishment shot can be achieved using several ways. One way is to use the long shot if the scene allows itself to be captured with a wide-angle lens. Another way is to use panning or tilting shot, normally used if the environment involves scenery that cannot be covered by a static shot. What this paper is interested in is the establishment shot that is achieved using panning or tilting shot. Knowing the location of these establishment shots helps a lot in analyzing the story units of the movies and in providing a hierarchical structure to a video.

We turn now to another common technique in cinematography, the so-called *object-tracking shot*, where the camera is following an object of interest. The resulting camera movement is often similar to that of the establishment shot, although it does not have the intention of introducing the surrounding environment to the audience but rather to indicate the region where the director wants the audience to focus upon. Since the establishment shot is liable to be confused with the tracking shot, it is necessary to formulate some measures to disambiguate the two.

Before doing so, it is well to distinguish the slightly different meaning of the term "tracking" used here as compared to that used in traditional computational vision. Here, a tracking shot means that the director indeed has the intention of tracking the target. The fact that the target in question is more or less tracked does not necessarily mean that it is a tracking shot from the point of view of the director. For instance, there is a type of shot called the *intermittent pan* [5], which comprises of a sequence of rotation around the vertical axis that covers intermittent activity by various groups of people (see Fig. 1).

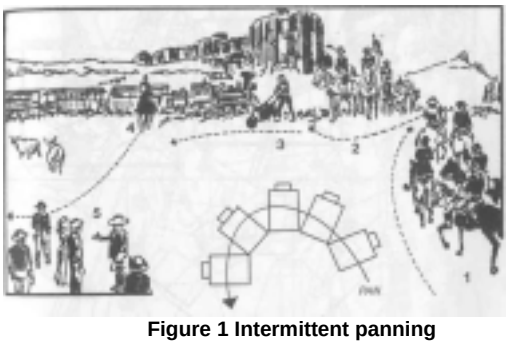

**Figure 1 Intermittent panning**

The camera begins by following a group of soldiers moving in double line (1). As they move away they meet an onlooking group turning to the left (2). The camera follows them. The soldiers then stop in front of a gardener (3) pushing a wheelbarrow. The camera follows him. The gardener stops as a man on horseback (4) passes in front of him. The camera moves with the rider. As he exits screen to the left, the camera stops and finally focus on a group of persons talking to each other (5). This type of shot can be considered as a form of establishment shot because the intermittent, overlapping actions give a sense of place as well as the activity going on in it, while at the end the central characters (5) are introduced naturally as part of the whole ensemble. The brief tracking episodes of characters 1, 2, 3 and 4 do not denote real interest in these characters as such (except in their being part of the whole ensemble) but are primarily used to lead up to the central characters in a smooth manner.

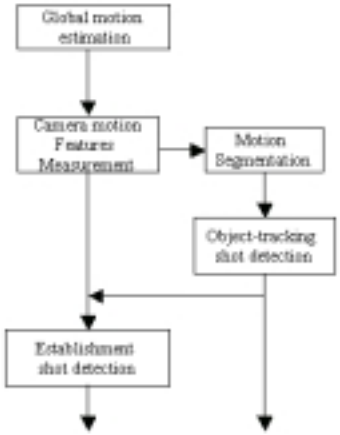


**Figure 2 Flow-chart of our algorithm**

Figure 2 summarizes the various blocks in our algorithm for establishment shot detection, which are presented in details in the next few sections.

## 3. Global motion estimation and outlier localization

The first step in our algorithm is to estimate the camera motion even with the presence of multiple motions. We assume that the dominant motion, which by definition covers half or more of the image, is only due to camera motion. Then an approach based on robust estimation of multiple motions by M.J. Black [6] is used to remove the flow fields at the motion boundaries of independently moving bodies. The approach first estimates a dominant motion in the scene and detects points that are not moving consistently with the dominant motion. These points are known as motion outliers. Adopting the affine motion model as in [6], the optical flow can be written as

$$u = a_0 + a_1 x + a_2 y$$
$$v = a_3 + a_4 x + a_5 y$$

where $\bar{a} = [a_0\ a_1\ a_2\ a_3\ a_4\ a_5]$ are the parameters of the affine motion model. A robust estimation method is then

used to find the best affine motion model through minimizing the error term

$$E = \sum_{(x,y)\in R} \rho[I_x u + I_y v + I_t]$$

for all the possible values of $\vec{a}$. In the above formula, $\rho$ is a robust function, ***R*** represents the whole image, the subscripts *x*, *y* and *t* denote the derivatives of the gray value *I* with respect to *x*, *y* and *t* respectively. Using the robust function, the effect of those pixels not conforming to the affine motion model can be reduced.

After obtaining the global affine motion model, the motion error energy $e(x, y)$ at each pixel is computed by

$$e(x, y) = | I_x u + I_y v + I_t |$$

If the error $e(x, y)$is larger than a constant threshold $TR_e$=10, this pixel is identified as a global motion outlier. Finally, we obtained a binary value mask representing the motion outliers. We refer to this mask as *energy map*.

## 4. Establishment Shot Detection

The input to the establishment shot detection module consists of the motion parameters obtained by the global motion estimation module. Our method of detection is based primarily on the observation that an establishment shot should exhibit motion with significant $a_0$ and $a_3$ component for a sustained period. From a computational point of view, this formulation has the advantage of bypassing the need to decouple the translational and the rotational terms ($a_0$ and $a_3$ lump them together). From a cinematic point of view, we also argue that these lumped quantities are the relevant categories since it seldom matters in practice whether say, a pan or a horizontal translation is used to effect a lateral camera movement.

The preceding observation can be formulated in terms of the following descriptors.

**Table. 1 Summary of establishment shot descriptors**

| | |
|---|---|
| $mag = \sqrt{a_0^2 + a_3^2}$ | Magnitude of lateral movement |
| *lcm* | Duration in which shot exhibits continual and large lateral movement |
| *len* | Total length of a shot in term of frames |
| *pct*= *lcm* × *10 / len* | Relative ratio of *lcm* against the total length of the shot |
| $dev = \sqrt{\frac{\sum_{n=1}^{N}(mag_n - avg)^2}{N-1}}$ | Standard deviation of the *mag* descriptor. |
| $smo = \sqrt{\frac{\sum_{n=2}^{N}(mag_n - mag_{n-1})^2}{N-1}}$ | Smoothness of the *mag* descriptor |

The *lcm* descriptor measures in a shot the longest period during which there is continual and large lateral movement. Its computation can be summarized by the following pseudocodes:

Let $lcm_i$ =1, if $mag > mag_{th}$ at frame i,
0, otherwise

```
for i :=1 to len do
if lcm_i = 1 then x++
else {
     if lcm < x then lcm := x
     x :=0}
```

To overcome problems caused by inaccuracy in motion estimation, which often results in the motion estimates momentarily dropping below the threshold and thus causing the system to underestimate the *lcm* value, we incorporate a hysteresis threshold test for the *mag* descriptor. That is, we introduce two thresholds for the *mag* value, $mag_{th1}$ and $mag_{th2}$, with $mag_{th2} > mag_{th1}$. The *mag* value is allowed to drop below $mag_{th1}$ for 10 frames without terminating the count for *lcm*. However, if the *mag* value drops below $mag_{th2}$ but above $mag_{th1}$, the drop is allowed for 20 frames.

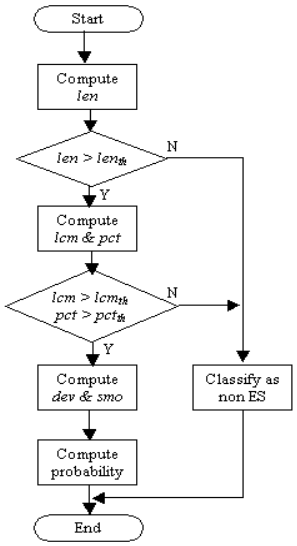


**Figure 3 Flow chart of establishment shot detection**

The flow chart in Fig. 3 describes the usage of all the descriptors and thresholds, where $xxx_{th}$ denotes the threshold value for the descriptor *xxx*.

## 5. Object-tracking shot detection

The method we put forth in section 4 cannot distinguish an establishment shot from an object tracking shot since these two kind of shots exhibit similar motion attributes. In this section, we introduce a scheme to detect those object-tracking shots which exhibit same motion features as the establishment shot and are thus liable to confusion. Our method has to first extract the target being tracked. It is quite obvious that the motion outlier detection scheme mentioned in section 3 alone will not

produce an accurate mask for the moving object due to the lack of texture in the interior of a region and therefore no reliable motion flow can be computed in this region. Our extraction method is based on the fusion of the motion segmentation result and a set of regions obtained from color segmentation. The main advantage is that they produce a better segmentation as a result. We already discussed the problem of global motion estimation in section 3. Here we discuss the other two components: color image segmentation and generating the final mask for the moving object.

### 5.1 Color image segmentation

In this paper, we used a recent color image segmentation technique, namely JSEG developed in [7], which has the capability to deal with both homogeneous texture regions and homogeneous color regions. JSEG defines a *J* value at each pixel of the original image, from which a *J* image was constructed .

This *J* value is based on the concept of between class variance and within class variance [8]. Assume *Z* is the set of *N* data points in the region ***R***, each data point has a class label *i* (*i*=1, 2, 3…*C*) where *C* is the total number of classes. For a color image, the *L*, *U*, *V* components of each pixel are quantized into several classes so that each pixel has a class label. Denote (*x*,*y*) by *z* and let $Z_i$ be the set of $N_i$ data points belonging to the $i^{\text{th}}$ class, the mean of the data points belonging to the $i^{\text{th}}$ class and the mean of all the data points are respectively:

$$m_i = \frac{1}{N_i}\sum_{z\in Z_i} z$$

$$m = \frac{1}{N}\sum_{z\in Z} z$$

Clearly, the within class variance and the total variance of the data set *Z* are given by respectively:

$$S_W = \sum_{i=1}^{C} S_i = \sum_{i=1}^{C}\sum_{z=Z_i} \| z - m_i \|^2$$

$$S_T = \frac{1}{N}\sum_{z\in Z} \| z - m \|^2$$

The *J* value of the region ***R*** is defined as

$$J = (S_T - S_W)/S_T$$

To calculate the corresponding *J* value at the pixel *z*=(*x*, *y*), a local circular window centered at this pixel is chosen as the region ***R*** referred to in the above discussion. The size of the local window determines the resolution of the region boundary detection. To locate region boundaries in different details, the radius of the local window we used ranged from 4 to 18 [7]. The *J* image thus obtained is then finally segmented into different regions to obtain a *region map* using region-growing method [9].

### 5.2 Generating the final mask

Motion segmentation provides us with an energy map, whose values indicate the locations of motion outliers. However, the motion outliers obtained are not enough to fully delineate a region. On the other hand, the color segmentation produces a number of closed regions but the object itself might be split into several regions. We now fuse the two results to obtain a better segmentation.

We collect from the energy map the motion error energy values of all the pixels at the region boundaries in the region map. We also propagate the motion error energy of all the pixels in the vicinity of the boundary pixels. After that, we identify those regions having big motion error energy on their boundaries.

The motion error energy at a boundary pixel is calculated in the following way. Let $e_{motion}(x_b, y_b)$ or $e_{motion}(z_b)$ be the motion error energy at the boundary pixel $z_b=(x_b, y_b)$, where the subscript *b* means the pixel is at a region boundary. Then identify those non-boundary pixels with nonzero motion error energy from the energy map. If there is such a pixel $z=(x, y)$ with nonzero motion error energy, we propagate the motion error energy of this pixel to all the boundary pixels $\{z_b=(x_b, y_b)\}$. The propagated value from *z* to $z_b$ is computed as follows:

$$\Delta e_{motion}(z, z_b) = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma}\exp\left[\frac{-d^2(z,z_b)}{2\sigma^2}\right] & \text{when } z_b \text{ is in the neighborhood of } z \\ 0 & \text{otherwise} \end{cases}$$

where, in the above equation, $d(z, z_b)$ is the Euclidean distance between *z* and $z_b$. The total motion error energy of the pixels $z_b$ at the region boundaries is

$$e_{motion}(z_b) = \sum_{z} \Delta e_{motion}(z, z_b)$$

There are two parameters: the size of the neighborhood and $\sigma$. We set the neighborhood as a $7\times7$ square area centered at the current pixel, and the second parameter as

We then compute the mean motion error energy $me(R_i)$ of *ith* segmented region in the region map:

$$me(R_i) = \frac{1}{N_i}\sum_{N_i} e(z_b)$$

where the summation is over the $N_i$ pixels on the boundary of region $\boldsymbol{R}_i$. Next, we identify those regions conforming to the following conditions as candidates of moving objects:

1. The area of the region $A(\boldsymbol{R}_i) \geq A_0$.
2. The mean motion error energy of the region $me(\boldsymbol{R}_i) \geq me_0$.

$A_0$ and $me_0$ are the thresholds of the region area and mean motion error energy respectively. Finally, merge those regions of area smaller than $A_0$ but of mean motion error energy greater than $me_0$ with the neighboring candidate moving objects if there are any.

### 5.3 Markov random field labeling

Since the background may be noisy or not perfectly conformed to the affine model assumption, the motion segmentation obtained previously can be further refined. We have therefore adapted the Markov Random Field (MRF) to refine the results of the motion segmentation obtained earlier. MRF processes are stochastic models having the ability to capture the spatial continuity inherent in images. They can be used to more accurately label the segmented regions as foreground objects or background. The energy functionals defined for the posteriori energy function are as proposed by Tsaig and Averbuch [10].

To impose temporal coherency on the labeling, Tsaig and Averbuch[10] introduced dynamic memory, which plays an important role in the energy functionals. This memory is a map defined over the image pixels, and functions approximately as a record of the resultant number of times a pixel belongs to foreground region. It is used in conjunction with an optical flow based mechanism to shift the memory values according to foreground movement.

However due to the approximating nature of this mechanism and the small size of the tracked object relative to the background, this scheme is not suitable for segmenting tracked objects in object tracking shots, especially over long sequences. This is due to minute background pixels being misclassified with foreground pixels over time, adversely affecting region classification.

We use a new dynamic memory system that defines a piece-wise linear memory function to assign memory values according to the function *mf*

$$mf(c_f - c_b) = \begin{cases} 2(c_f - c_b) + 10 & (c_f - c_b) > 10 \\ (c_f - c_b) & -10 \le (c_f - c_b) \le 10 \\ \textit{values capped at } [-10,20] & \end{cases}$$

where $c_f, c_b$ is the number of times it has been classified as foreground and background over a predefined number of recent frames. The memory values at both ends are capped to ensure more responsiveness, with the positive end having a higher cap to reflect the fact that there is more foreground than background. The higher gradient near the end of the positive *x* axis encourages a high $c_f - c_b$ to result in higher chances of being classified as foreground. The resultant labeling of regions as shown in Fig. 4 using the MRF refines the region segmentation to produce an accurate segmentation mask.

### 5.4 Tracking

We use just one criteria to distinguish an object tracking shot from an establishment shot, namely the independently moving object(s) must be present throughout most, if not all of the entire length of the shot. This is because the object tracking shot is by definition a director's device to focus the attention of the audience on an object(s), without which the shot will not serve the purpose. Conversely, the presence of the same independently moving object(s) throughout an establishment shot is extremely unlikely.

a. Independent object segmentation without MRF

b. Independent object segmentation with MRF

**Figure 4 Motion Segmentation Resutls**

We exploit certain characteristics unique to object tracking shots. As opposed to the traditional problem of tracking large numbers of arbitrarily moving points, we are corresponding a small number of measurements to a similarly small number of objects that may possibly be undergoing drastic changes in appearances that make them difficult to track. These drastic changes, or "tracking transitions", include sudden and prolonged occlusion, merging, splitting and uncovering. Due to the possibly severe nature of the tracking transitions of tracked objects, an approach that is specifically optimized for these situations using ad-hoc heuristics will be simpler, and work better.

For each single frame, the algorithm first uses a dilution morphological filter on the motion segmentation mask to filter off noise. This is followed by a greedy recursive region filling procedure using 8-connectivity. Regions below a certain size are eliminated, and remaining ones are labeled with a region number. Several blob attributes like the area, centroid, moments and color statistics of each of the 3 channels are extracted from every region. Finally, the area of overlap between the present and previous frame's segmentation mask for every possible pair of regions is computed.

Let an object blob be $O_{k,i}$, where *k* is the frame number where the object blob is observed and *i* is the index of the blob in its frame. The validation measure we use consists of both the centroid distance and area overlap between $O_{k,i}$ and $O_{k-1,j}$, where *i, j* are blob indices iterating through every valid blob in its frame. We find that area overlap is a very robust and accurate method to find one-to-one correspondence between blobs not undergoing transitions.

We next detect for merging and splitting transitions. However assuming that such transitions only involve two blobs, the area overlap measure can once again determine the correspondences that give rise to these transitions via

area checksums. We then detect occlusion and unocclusion transitions. If we hypothesize $O_{k-1,j}$ maps to $O_{k,i}$ via these transitions, then both their area change and overlap should be above a threshold. Finally, the following criteria should hold true:

$$SSD(OT_{k-1,j}, O_{k,i}) > SSD(O_{k-1,j}, O_{k,i})$$

where $OT_{k-1,j}$ is $O_{k-1,j}$ transposed by centroid differences between $O_{k-1,j}$, $O_{k,i}$ and SSD is the Sum of Squared Difference defined over the pixels in the overlap.

We use the number of recent frames over which a blob is undergoing transition as a measure of blob instability. This allows us to constantly save a blob's attributes while it is stable for later matching and identification purposes, where the Mahalanobois distance of blob attributes is used for matching. We also incorporated the ability to delete and initiate new blobs to identify objects that may be temporarily merged or occluded in order that an object tracking shot can be correctly classified.

## 6. Experimental Results

For the test, we used 50 video shots (totally 54 minutes) cut out manually from four VCDs (The Bodyguard, The Cliffhanger, The Predator, Walking with Dinosaur). In these clips, there are 16 establishment shots, 7 object-tracking shots and 27 other shots. After the first step of establishment detection, there are totally 22 shots detected as establishment shots. We called these shots as potential establishment shots in table 2. The true class labeling of these shots are listed in the last three columns in the same table.

**Table 2 Complete results of the experiments**

| | No. of clips | ES | TS | Other |
|---|---|---|---|---|
| Test Database | 50 | 16 | 7 | 27 |
| Potential ES | 22 | 14 | 6 | 2 |
| Potential TS | 6 | 1 | 5 | 0 |
| Correctly Classified | 43 | 13 | 5 | 25 |
| ES: establishment shot; TS: object-tracking shot | | | | |

As expected, most of the object-tracking shots were falsely detected as establishment shots. We then passed the output of the first step, i.e. a total of 22 potential establishment shots, to the module for object-tracking shot detection. Five of them were correctly detected as potential object-tracking shots, with one false alarm. Due to the lack of sustained independent motion in establishment shots, the tracking module had no difficulty in recognizing most true establishment shots save for one false alarm, which was due to violation of the affine model assumption. There was one remaining object tracking shots that was misidentified as potential establishment shots. This is missed by the tracking module because the tracked object was too small for the motion segmentation module to detect it. Due to the rarity of intermittent panning establishment shots, this category of shot was not tested. However the tracking mechanics involved is exactly equivalent to transitionless tracking, if not simpler: these shots are automatically disqualified the moment the first tracked object leaves the screen.

## 7. Conclusion

This work presents an approach to detect establishment shots, which is realized by analyzing qualitatively the lateral camera movement as described by the global affine motion model. Due to the similar motion attributes between establishment shots and object-tracking shots, we put forth a technique to differentiate the object-tracking shots from the establishment shots. The segmentation of tracked objects is based on motion and color segmentation, which is further refined by a MRF. The segmented output is fed into a tracking module to detect object tracking shots. The results significantly improve the precision rate of establishment shot detection by removing these object tracking shots. For better precision, we intend in our future work to devise an approach to detect close-up shots, which can cause errors by violating the assumption of the dominant motion being the background motion.

## 8. References


[1] A. Akutsu, et. al., "Video indexing using motion vectors," *Proc. SPIE: Visual Communications and Image Processing* 1818, pp. 1522-1530, 1992.

[2] L. Cheong, et. al., "Effects of errors in the viewing geometry on shape estimation," *Computer Vision and Image Understanding* 71(3), pp. 356-372, 1998.

[3] K. Daniilidis and M.E. Spetsakis, "Understanding noise sensitivity in structure from motion," *Visual Navigation: From Biological Systems to Unmanned Ground Vehicles*, Y. Aloimonos (Ed.), Lawrence Erlbaum Assoc., Pub. 1995.

[4] G.S. Young and R. Chellapa, "Statistical analysis of inherent ambiguities in recovering 3-D motion from a noisy flow field," *IEEE Trans. PAMI* 14, pp. 995-1013, 1992.

[5] Daniel Arijon, *Grammar of the film language*, Silman-James Press., 1991.

[6] M. J Black and P. Anandan, "The robust estimation of multiple motions: parametric and piecewise-smooth flow fields", *Computer Vision and Image Understanding Vol. 63*, 75-104, Jan 1996.

[7] Yining Deng and B. S. Manjunath, "Unsupervised segmentation of color-texture region in images and video", *IEEE Trans. PAMI* 23, pp. 800-810, 2001.

[8] R.O Duda and P.E. Hart, *Pattern Classification and Scene Analysis*, John Wiley & Sons, New York, 1970.

[9] S.C. Zhu and A. Yuille, "Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation", *IEEE Trans. PAMI*, vol. 18, no.9, p884-900.

[10] Y. Tsaig and A. Averbuch. "Automatic segmentation of moving objects in video sequences: a region labeling approach, *IEEE Trans. CSVT*, vol.12, no.7, 2002.