INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

# *A Three-layer MRF model for Object Motion Detection in Airborne Images*

Csaba Benedek — Tamás Szirányi — Zoltan Kato — Josiane Zerubia

**N° 6208**

June 2007

Thème COM

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--6208--FR+ENG

# A Three-layer MRF model for Object Motion Detection in Airborne Images

Csaba Benedek*† , Tamás Szirányi†* , Zoltan Kato‡ , Josiane Zerubia§

Thème COM — Systèmes communicants
Projets ARIANA

Rapport de recherche n° 6208 — June 2007 — 35 pages

**Abstract:** In this report, we give a probabilistic model for automatic change detection on airborne images taken with moving cameras. To ensure robustness, we adopt an unsupervised coarse matching instead of a precise image registration. The challenge of the proposed model is to eliminate the registration errors, noise and the parallax artifacts caused by the static objects having considerable height (buildings, trees, walls etc.) from the difference image. We describe the background membership of a given image point through two different features, and introduce a novel three-layer Markov Random Field (MRF) model to ensure connected homogenous regions in the segmented image.

**Key-words:** aerial images, change detection, camera motion, MRF

* Pázmány Péter Catholic University, Department of Information Technology, Budapest, Hungary.
† Distributed Events Analysis Research Group of the Computer and Automation Research Institute, Budapest, Hungary
‡ University of Szeged, Institute of Informatics, Szeged, Hungary
§ Ariana (joint research group INRIA/CNRS/UNSA), Sophia-Antipolis, France

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex (France)
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

# Un modèle de champ de Markov pour la détection de mouvement sur des images aériennes

**Résumé :** Dans ce rapport, nous proposons un modèle stochastique pour la détection automatique de changements sur des images aériennes prises à l'aide de cameras mobiles. Afin d'assurer la robustesse de la méthode, nous adoptons une technique de mise en correspondance grossière non-supervisée au lieu d'une méthode précise de recalage d'images. Le défi du modèle proposé est de pouvoir éliminer les erreurs de recalage, le bruit et les artefacts dus au paralaxe causés par des objets statiques qui ont une certaine hauteur (bâtiments, arbres, murs, etc.), ceci à partir de l'image des différences. L'appartenance d'un point de l'image au fond est décrite grâce a' deux attributs différents et nous introduisons un champ de Markov original à 3 niveaux afin d'obtenir des régions homoge'nes connectées dans l'image segmentée.

**Mots-clés :** images aériennes, détection de changements, camera mobile, champ de Markov

# Contents

# 1 Introduction

Change detection is an important early vision task in several computer vision applications. Shape, size, number and position parameters of the moving objects can be derived from the change-mask and used, for example for people or vehicle detection, tracking and activity analysis. Object motion detection is also a key issue in aerial surveillance and exploitation [25]. Here, the task is more difficult to obtain, since the images to be compared are taken at different camera positions.

The present paper addresses the problem of detecting the accurate silhouettes of moving objects, or at least, object-groups in image pairs taken by moving airborne vehicles in consecutive moments. The shots are focused on urban roads. We consider the presence of static objects in the scene, like small buildings, trees and walls. The time difference between the corresponding images is approximately 1 second, meanwhile the moving objects change their position significantly.

The task needs an efficient combination of image registration for camera motion compensation and frame differencing. For registration, feature correspondence is widely used, where we look for corresponding pixels or other primitives such as edges, corners, contours, shape etc. in the images which we compare [2][5][31][44][45]. However, this procedure may fail at occlusion boundaries and within regions where the chosen primitives or features cannot be reliably detected. We find methods focusing on the reduction of errors at object boundaries caused by occlusion [10][11], but these approaches work with slightly different images used in stereo vision. In [41], a motion-based method is presented for automatic registration of images in multi-camera systems, to enable the synthesis of wide-baseline composite views. However, the latter method needs video flows recorded by static cameras.

In our application, the camera may move continuously and rapidly causing significant global offset and rotation between the consecutive frames. Thus, we must expect that feature matching presents correct pixel correspondences only for sparsely distributed feature points instead of matching the two frames completely. A possible way to handle this problem is searching for a global 2D transform between the images. Two main approaches are available. Pixel correspondence based techniques estimate the optimal coordinate transform (e.g. homography) which maps the extracted feature points of the first image to the corresponding pixels identified by the feature tracker module in the second frame [45]. In global correlation methods, the goal is to find the parameters of a similarity [36] or affine transform [29] for which the correlation between the original first and transformed second image is maximal. For computational purposes, these methods work in the Fourier domain.

Although there are sophisticated ways to enhance the accuracy of the 2D mappings [23], these approaches cause significant parallax errors [45] at locations of static scene objects with considerable height (see Fig 1). To overcome this problem plane+parallax [14] models have been frequently used. However, [14] emphasizes that performance of these modells is very sensitive to find the accurate epipoles, which may fail if, besides camera motion, many independent object displacements are present in the scene. [39] uses shape constancy constraints together with global motion estimation for very low altitude aerial videos captured from sparsely cultural scenes. More specifically, the '3Dness' of the scene is sparsely distributed containing a few moving objects, while the algorithm needs at least three frames from a video sequence. On the other hand, in scenarios being investigated in the current paper, both the 3D static objects and the object motions are densely distributed, but the frames are captured from higher altitude, thus the parallax distortions usually cause errors of a few pixels. We do not expect that a video sequence is available, thus we may have only two images to compare. Hence, [32] cannot be used here, since it exploits a prediction for the camera motion based on previously processed frames, while [34] needs also processing long videos.

For the above reasons, we introduce a two stage algorithm which consists of a coarse (but robust) image registration for camera motion compensation, and an error-eliminating step. From this point of view, it is similar to [6], where the authors assume that errors mainly appear near sharp edges. Therefore, at locations where the magnitude of the gradient is large in both images, they consider that the differences of the corresponding pixel-values are caused with higher probability by registration errors than by object displacements. However, this method is less effective, if there are several small objects (containing several edges) in the scene, because the post processing may also remove some real objects, but it leaves errors in smoothed textured areas (e.g. group of trees, corresponding test results are shown in Section 6).

Another important issue is related to feature selection. Scalar valued features may be weak to model complex classes appropriately, therefore integration of multiple observations has been intensively examined recently [1][12][16]-[20],[22][26][27][38][43]. According to one of the most straightforward approaches, an $n$ dimensional feature vector is constructed from the observations [20] and for each class, the distribution of the features should be approximated by an $n$ dimensional multinomial density function. A practical problem may appear here: although the feature vector's one dimensional marginal distributions can be often modelled well with well-known densities (e.g. Gaussian, Beta, uniform, or a finite mixture of them), the joint distribution may be hard to express. For example, there is a difficult case, if the first feature-dimension can be modelled by a Gaussian term and the second one follows a uniform distribution. Moreover, efficient methods for probability calculation and parameter estimation are only available for certain distributions. The correspondence between the feature components may be also difficult to model, or, at least, increases the number of free parameters (e.g. the Gaussian correlation matrix must be non-diagonal).

For the above reasons, multi-layer models have became popular nowadays [16]-[19]. In this case, individual layers are assigned to the different feature components (or to a group of components). Each layer's segmentation is directly influenced by its corresponding measurement component(s) and indirectly by features of the other layers. The inter-layer connections may achieve data interaction [16][17][19] (the inter-layer interactions also use the features' datas and the segmentation labels directly) or label fusion [15][26] (the interactions use only the labels in the different layers). Usually, the right choice between these two approaches depends on the domain which we model. We show later that regarding the problem, which we investigate in this paper, the label fusion is more a natural model.

In this report, we use a Bayesian approach to tackle the above change detection problem.

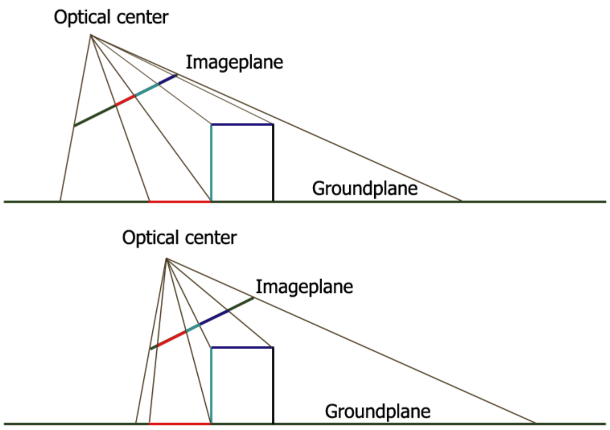


Figure 1: Illustration of the parallax effect, if a rectangular high object appears on the ground plane. We mark different sections with different colors on the ground and on the object, and plot their projection on the image plane with the same color. We can observe that the length ratio of the corresponding sections is significantly different.

We derive features describing the background membership of a given image point in two independent ways, and develop a three-layer Bayesian labeling model to integrate the effects of the different features. We use a similar model structure to [16]-[19], which has two layers corresponding to the different observations, and a third one presenting the final foreground-background segmentation result. However, there are two essential differences: while in [16]-[19], the segmentation classes in the combined layer were constructed as the cross product of the classes at the observation layers, we use the same classes in each layer: foreground and background. On the other hand, we define the inter-layer connections also differently: in [16]-[19], the observation layers were directly connected with the segmentation layer via doubleton cliques, while we define connections between all three layers via cliques of site-triples.

## 2 Image registration

In this section, we define the formal image model. Hereafter, we introduce briefly two approaches on coarse image registration. Finally, we compare the methods on the images of our datasets, and we choose the most appropriate one to be the preprocessing step of our Bayesian labeling model.

## 2.1 Image model

Denote by $X_1$ and $X_2$ the two consecutive frames of the image sequence above the same pixel lattice $S$. The gray value of a given pixel $s \in S$ is $x_1(s)$ in the first image and $x_2(s)$ in the second one. A pixel is defined by a two dimensional vector containing its x-y coordinates: $s = [s_x, s_y]^T$, $s_x = 1...M$, $s_y = 1...N$. We define a 4-neighborhood system on the lattice:

$$\forall s \in S: \quad \Phi_s = \{r \in S \ : \ ||s - r||_{L1} = 1\}, \tag{1}$$

where we determine the distance between two pixels by the Manhattan (L1) distance.
Formally, the segmentation procedure is a labeling process: a label is assigned to each pixel $s \in S$ from the label-set: $L = \{\text{fg}, \text{bg}\}$, corresponding to the two classes: foreground (fg) and background (bg). A pixel belongs to foreground if it is part of an object displacement.

## 2.2 FFT-Correlation based similarity transform (FCS)

Reddy and Chatterji [36] proposed an automatic and robust method for registering images, which are related via a similarity transform (translation, rotation and scaling). In this approach, the goal is to find the parameters of the similarity transform $\mathcal{T}$ for which the correlation between $X_1$ and $X_2^\dagger = \mathcal{T}(X_2)$ is maximal.
The method is based on the Fourier shift theorem. In the first step, we assume that $X_1$ and $X_2$ images differ only in displacement, namely there exists an offset vector $o^*$, for which $x_1(s) = x_2(s + o^*) : \forall s, s + o^* \in S$. Let us denote with $X_2^o$ the image we get by shifting $X_2$ with offset $o$. In this case, $o^* = \text{argmax}_o C_r(o)$, where $C_r$ is the correlation map: $C_r(o) = \text{Corr}\{X_1, X_2^o\}$. $C_r$ can be determined efficiently in the Fourier domain. Let $F_1$ and $F_2$ be the Fourier transforms of the images $X_1$ and $X_2$. We define the Cross Power Spectrum (CPS) by:

$$\text{CPS}(\eta, \xi) = \frac{F_1(\eta, \xi) \cdot \overline{F}_2(\eta, \xi)}{|F_1(\eta, \xi) \cdot \overline{F}_2(\eta, \xi)|} = e^{j2\pi(o_x\eta + o_y\xi)},$$

where $\overline{F}_2$ means the complex conjugate of $F_2$. Finally, the inverse Fourier transform of the CPS is equal with the correlation map $C_r$[36].
The Fourier shift theorem also offers a way to determine the angle of the rotation. Assume that $X_2$ is a translated and rotated replica of $X_1$, where the translation vector is $o$ and the angle of rotation is $\phi$. It can be shown that considering $|F_1|$ and $|F_2|$ as images, $|F_2|$ is the purely rotated replica of $|F_1|$ with angle $\phi$. On the other hand, rotation in the Cartesian coordinate system is equivalent to a translational displacement in the polar representation [36], which can be calculated similarly to the determination of $o^*$.
The scaling factor of the optimal similarity transform may be retrieved in an analogous way [36].
In summary, we can determine the optimal similarity transform $\mathcal{T}$ between the two images based on [36], and derive the (coarsely) registered second image, $X_2^\dagger$. In the following, $x_2^\dagger(s)$ will denote the gray value of pixel $s$ in $X_2^\dagger$.

### 2.3 Pixel-correspondence based homography matching (PCH)

This approach consist of two consecutive steps. First, corresponding pixels are collected in the images, thereafter, the optimal coordinate transform is estimated between the elements of the extracted point pairs [45]. Therefore, only the first step is influenced directly by the observed image data, and the method may fail if the feature tracker produces poor result. On the other hand, we can obtain a more general transformation in this way than with FCS.

In our implementation, we search for pixel correspondences for sharp corner pixels with the pyramidal Lucas-Kanade feature tracker [4][28]. The set of the resulting point pairs contains several outliers, which are filtered out by the RANSAC algorithm [9], while the optimal homography is estimated so that the back-projection error is minimized [13].

### 2.4 Experimental comparison of FCS and PCH

The FCS and PCH algorithms have been tested on our test image pairs. Obviously, both gives only a coarse registration, which is inaccurate and is disturbed by parallax artifacts. In fact, FCS is less effective if the projective distortion between the images is significant. The weak point of PCH appears if the object motion is dense, thus a lot of point pairs may be in moving objects, and the automatic outlier filtering may fail, or at least, the homography estimation becomes inaccurate.

In our test database, the latter artifacts are more significant, since the corners of the several moving cars present dominant features for the Lucas-Kanade tracker. Consequently, if $C^*$ is the number of all the detected corner pixels and $C^o$ is the number of corner pixels on moving objects; while $P^*$, $P^o$ denote the number of all pixels and pixels corresponding to object displacement, respectively, $\frac{C^o}{C^*} \gg \frac{P^o}{P^*}$ may hold and the FCS method becomes much more robust.

Some corresponding results are presented in Fig. 2. We can observe that using FCS, the error-appearances are limited to the static objects boundaries, while regarding two out of the four frames, the PCH registration is highly erroneous. We note that the Bayesian post processing, which will be proposed later in this report, can remove the FCS errors, but it is unable to deal with the large PCH gaps.

For the above mentioned reasons, we will use the FCS method for preliminary registration in the following part of this report, however, in other test scenes it can be replaced with PCH in a straightforward way.

## 3 Feature selection

In this section, we introduce the feature selection using an airborne photo pair.[1] Taking a probabilistic approach, first we extract features, and then consider the class labels to be

[1]We have also observed similar tendencies regarding the other test images, provided by the ALFA project.


Figure 2: Qualitative illustration of the coarse registration results presented by the FFT-Correlation based similarity transform (FCS), and the pixel-correspondence based homography matching (PCH). In col 3 and 4, we find the thresholded difference of the registered images. Both results are quite noisy, but using FCS, the errors are limited to the static object boundaries, while regarding P#25 and P#52 the PCH registration is erroneous. Our Bayesian post processing is able to remove the FCS errors, but it cannot deal with the demonstrated PCH gaps.

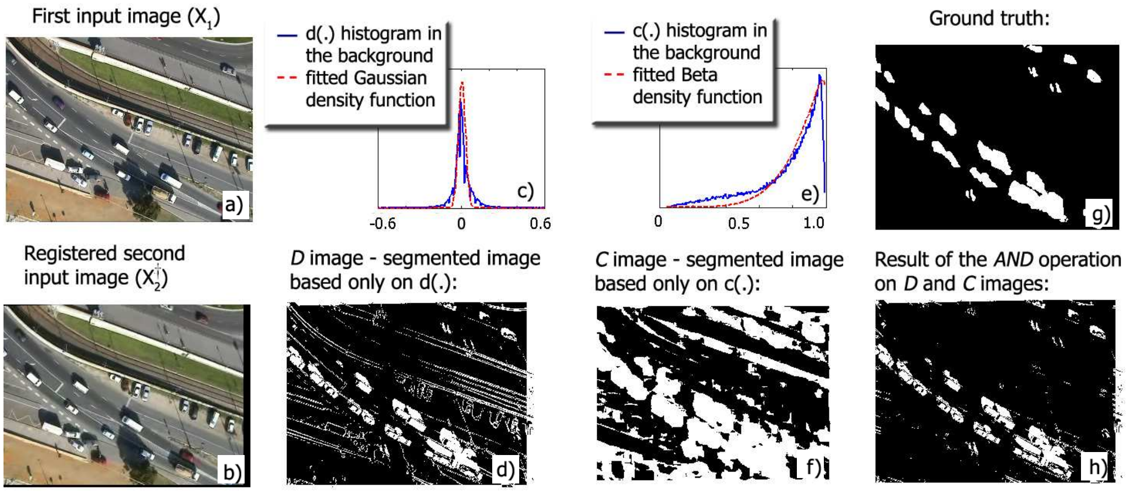


Figure 3: Feature selection. Notations are in the text of Section 3.

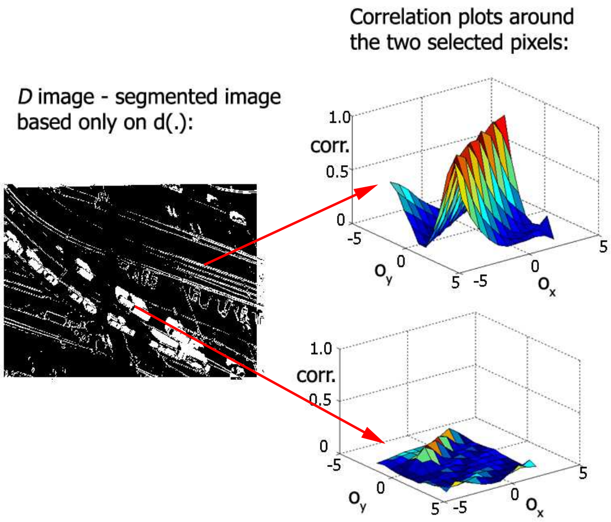


Figure 4: Plot of the correlation values over the search window around two given pixels. The upper pixel corresponds to a parallax error in the background, while the lower pixel is part of a real object displacement.

random processes generating the features according to different distributions.

### 3.1 Definition and illustration of the features

The first feature is the gray level difference of the corresponding pixels in the registered images:

$$d(s) = x_2^{\dagger}(s) - x_1(s).$$

We validate this feature through experiments (Fig. 3c): if we plot the histogram of $d(s)$ values corresponding to manually marked background points, then we can observe that a Gaussian approximation is reasonable:

$$P(d(s)|\mathrm{bg}) = N(d(s), \mu, \sigma) =$$

$$= \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(d(s)-\mu)^2}{2\sigma^2}\right). \tag{2}$$

On the other hand, any $d(s)$ value may occur in the foreground, hence the foreground class is modeled by a uniform density:

$$P(d(s)|\mathrm{fg}) = \begin{cases} \frac{1}{b_d - a_d}, & \text{if} \quad d(s) \in [a_d, b_d] \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Next, we demonstrate the limitations of this feature. After supervised estimation of the distribution parameters, we derive the $D$ image in Fig. 3d as the maximum likelihood estimate: the label of $s$ is

$$\mathrm{argmax}_{\psi \in \{\mathrm{fg},\mathrm{bg}\}} P(d(s)|\psi).$$

We can observe here that the registration and parallax errors cannot be filtered out using only $d(.)$, since their $d(s)$ values appear as outliers with respect to the previously defined Gaussian distribution.
From another point of view, assuming the presence of errors of a few pixels, we can usually find an $o_s = [o_x, o_y]$ offset vector, for which the rectangular neighborhood of $s$ in $X_1$ and the same shaped neighborhood of $s + o_s$ in $X_2^{\dagger}$ is strongly correlated. Correlation of two image parts $A = \{a_1, a_2, \dots a_n\}$ and $B = \{b_1, b_2, \dots b_n\}$, where $(a_i, b_i)$ are the values of the corresponding pixels, $\overline{a}$ and $\overline{b}$ being the mean values in the images, is computed by:

$$\mathrm{Corr}(A, B) = \frac{\sum_{i=1}^{n} (a_i - \overline{a})(b_i - \overline{b})}{\sqrt{\sum_{i=1}^{n} (a_i - \overline{a})^2 \sum_{i=1}^{n} (b_i - \overline{b})^2}}. \tag{4}$$

In Fig 4, we plot the correlation values over the search window of the offset $o_s$ around two given pixels (marked with the beginning of the arrows in Fig 4). The upper pixel corresponds to a parallax error in the background, while the lower one is part of a real object displacement. The correlation plot exhibits a high peak only in the upper case. We use $c(s)$, the maxima in the local correlation function around pixel $s$ as second feature. By examining the histogram of $c(s)$ values in the background (Fig 3e), we find that it can be approximated with a beta density function:

$$P(c(s)|\text{bg}) = B(c(s), \alpha, \beta), \tag{5}$$

where

$$B(c, \alpha, \beta) = \begin{cases} \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} c^{\alpha-1}(1-c)^{\beta-1}, & \text{if} \quad c \in (0,1) \\ 0 & \text{otherwise} \end{cases}$$

$$\Gamma(\alpha) = \int_0^\infty t^{\alpha-1} e^{-t} dt.$$

As for the foreground class we will use a uniform probability $P(c(s)|\text{fg})$ with $a_c$ and $b_c$ parameters. We see in Fig. 3f ($C$ image) that the $c(.)$ descriptor causes also poor result in itself. Even so, if we consider $D$ and $C$ as a Boolean lattice, where 'true' corresponds to the foreground label, the logical AND operation on $D$ and $C$ improves the results significantly (Fig. 3h). We note that this classification is still quite noisy, although in the segmented image, we expect connected regions representing the motion silhouettes. Morphological postprocessing of the regions may extend the connectivity, but assuming the presence of various shaped objects or object groups, it is hardly possible to define appropriate morphological rules. Since the work of Geman and Geman [8], Markov Random Fields (MRFs) offer a powerful tool to ensure contextual classification. However, our case is particular: we have two weak features, which present two different (poor) segmentations, while the final foreground-background clustering depends directly on the labels of the weak segmentations. To decrease noise, we must prescribe, that both the weak and the final segmentations must be 'smooth'. For the above reasons, we introduce a robust segmentation model in Section 4.

## 3.2 Justification of the feature selection

Based on the experiments of the previous section, the gray level difference and the local correlation seem to be complementary features which describe together the background class efficiently. This observation can be empirically explained as follows:

1. If the gray-level difference $d(s)$ votes for background at $s$, the correct segmentation class of $s$ is usually background (except in cases of background-colored object points).

2. If the gray level difference $d(s)$ votes for foreground at $s$ we may have two possibilities:

   - $s$ is a real foreground object pixel,
   - $s$ is the location of a registration/parallax error. This artifact occurs mainly in textured 'background' areas and near to the region boundaries. On the other hand, if the background is homogenous in the neighborhood of $s$, the pixel values in a few pixel distance are similar, so $d(s)$ difference is close to the $\mu$ value expected in the background (see eq. 2).

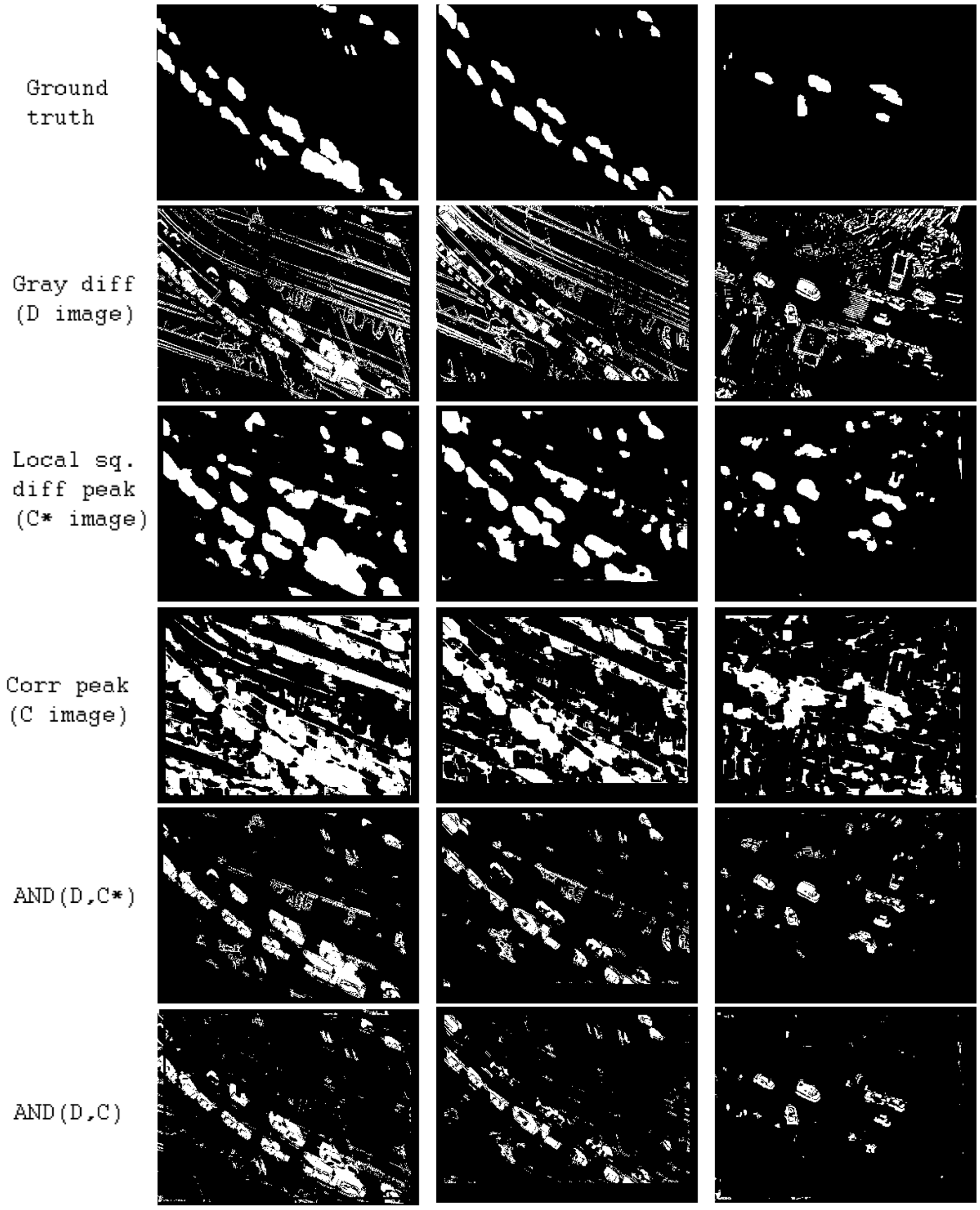


Figure 5: Qualitative comparison of the 'sum of local squared differences' ($C^*$) and the 'normalized cross correlation' ($C$) similarity measures with our label fusion model. In itself, the segmentation $C^*$ is significantly better than $C$, but after fusion with $D$, the normalized cross correlation outperforms the squared difference.

3. If the correlation-peak-feature $c(s)$ votes for background at $s$, the correct segmentation class is usually background.

4. If the correlation-peak-feature $c(s)$ votes for foreground at $s$ we may have two possibilities:

   - $s$ is a real foreground object point,
   - the normalized correlation is erroneously low around $s$. This artifact occurs mainly in homogenous 'background' areas: if the variance of the pixel values in the rectangular correlation window is low, eq. 4 becomes quite sensitive to noise.

Therefore, we can summarize that the $d(.)$ and $c(.)$ features may cause quite a lot of false positive foreground points, however, the rate of false negative detection[2] is low in both cases: they appear only at location of background-colored object parts, and they can be partially eliminated by the smoothness constraints introduced via a MRF [8]. Moreover, examining $d(s)$ results usually in a false positive decision if the neighborhood of $s$ is textured, but in that case the decision based on $c(s)$ is usually correct. Similarly, if $c(s)$ votes erroneously, we can usually trust in the hint of $d(s)$. This observation is confirmed by the experimental results of Section 3.1 and supports our decision structure: the class of $s$ is usually background, if and only if at least one of the $d(s)$ or $c(s)$ features votes for background.

We make two further comments regarding the feature selection. First, the proposed segmentation scheme is a label fusion (like [15][26]) of two 'weak' segmentations, instead of observation fusion ([22][20]) of the features $d(.)$ and $c(.)$. Hence, the final segmentation labels depend on the observations indirectly via the 'weak' segmentation labels.

Secondly, the limitation of the $c(.)$ descriptor is caused by the denominator term in the normalized correlation expression (eq. 4). Here, we offer as alternative descriptor a non-normalized similarity factor, namely, the simple squared difference. For $A = \{a_1, a_2, \dots a_n\}$ and $B = \{b_1, b_2, \dots b_n\}$:

$$\text{Sqdiff}(A, B) = \sum_{i=1}^{n} (a_i - b_i)^2, \tag{6}$$

and denote by $c^*(s)$ the minimal Sqdiff value around $s$, while $C^*$ is the segmented image based on $c^*(.)$. We show some comparative experimental results for features $C$ and $C^*$ in Fig. 5. We can observe that in itself, $C^*$ has significantly better quality than $C$, but $c(.)$ is a better complementary feature of $d(.)$, and the $D - C$ joint segmentation is better than the clustering based on $D - C^*$.

# 4 Multi-layer segmentation model

In the proposed approach, we construct a Markov Random Field (MRF) model on a graph $\mathcal{G}$ whose structure is shown in Fig. 6. In the previous section, we segmented the images in two

[2] Number of pixels corresponding to real object displacements but classified as background.

independent ways, and derived the final result by a label fusion using the two segmentations. Therefore, we arrange the sites of $\mathcal{G}$ into three layers $S^d$, $S^c$ and $S^*$, each layer has the same size as the image lattice $S$. We assign to each pixel $s \in S$ a unique site in each layer: e.g. $s^d$ is the site corresponding to pixel $s$ on the layer $S^d$. We denote $s^c \in S^c$ and $s^* \in S^*$ similarly.

We introduce a labeling process, which assigns a label $\omega(.)$ to all sites of $\mathcal{G}$ from the label-set: $L = \{\text{fg, bg}\}$. The labeling of $S^d/S^c$ corresponds to the segmentation based on the $d(.)/c(.)$ feature, respectively; while the labels at the $S^*$ layer present the final change mask. A global labeling of $\mathcal{G}$ is

$$\underline{\omega} = \left\{\omega(s^i) | s \in S, i \in \{d, c, *\}\right\}.$$

In our model, the labeling of an arbitrary site depends directly on the labels of its neighbors (MRF property). For this reason, we must define the neighborhoods (i.e. the connections) in $\mathcal{G}$ (see Fig. 6). To ensure the smoothness of the segmentations, we put connections within each layer between site pairs corresponding to neighboring pixels of the image lattice $S$.[3] On the other hand, the sites at different layers corresponding to the same pixel must interact in order to produce the fusion of the two different segmentations labels in the $S^*$ layer. Hence, we introduce 'inter-layer' connections between sites $s^i$ and $s^j$: $\forall s \in S$; $i, j \in \{d, c, *\}$, $i \neq j$. Therefore, the graph has doubleton 'intra-layer' cliques (their set is $\mathcal{C}_2$) which contain pairs of sites, and 'inter-layer' cliques ($\mathcal{C}_3$) consisting of site-triples. We also use singleton cliques ($\mathcal{C}_1$), which are one-element sets containing the individual sites: they will link the model and the local observations. Hence, the set of cliques is $\mathcal{C} = \mathcal{C}_1 \cup \mathcal{C}_2 \cup \mathcal{C}_3$.

Denote the observation process by

$$\mathcal{F} = \{f(s) | s \in S\},$$

where $f(s) = [d(s), c(s)]$.

Our goal is to find the optimal labeling $\widehat{\underline{\omega}}$, which maximizes the a posterior probability $P(\underline{\omega}|\mathcal{F})$ that is a maximum a posteriori estimate (MAP) [8]:

$$\widehat{\underline{\omega}} = \text{argmax}_{\underline{\omega} \in \Omega} P(\underline{\omega}|\mathcal{F}).$$

where $\Omega$ denotes the set of all the possible global labelings. Based on the Hammersley-Clifford Theorem [8] the a posterior probability of a given labeling follows a Gibbs distribution:

$$P(\underline{\omega}|\mathcal{F}) = \frac{1}{Z} \exp\left(-\sum_{C \in \mathcal{C}} V_C(\underline{\omega}_C)\right),$$

where $V_C$ is the *clique potential* of $C \in \mathcal{C}$, which is 'low' if $\underline{\omega}_C$ (the label- subconfiguration corresponding to $C$) is semantically correct, 'high', if not. $Z$ is a normalizing constant, which does not depend on $\underline{\omega}$.

In the following part of this section, we define the clique potentials. We refer to a given clique as the set of its sites (in fact, each clique is a subgraph of $\mathcal{G}$), e.g. we denote the

[3] We use first order neighborhoods in $S$, where each pixel has 4 neighbors.

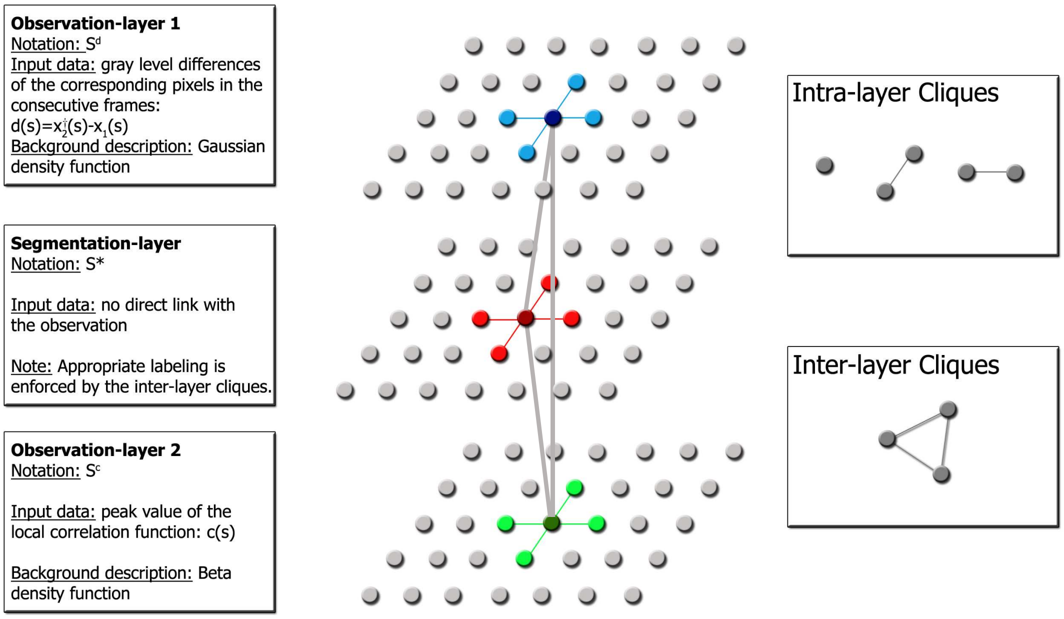


Figure 6: Summary of the proposed three layer MRF model

doubleton clique containing sites $s^d$ and $r^d$ with $\{s^d, r^d\}$.

The observations affect the model through the singleton potentials. As we stated previously, the labels in the $S^d$ and $S^c$ layers are directly influenced by the $d(.)$ and $c(.)$ values, respectively, $\forall s \in S$:

$$V_{\{s^d\}}\left(\omega(s^d)\right) = -\log P(d(s)|\omega(s^d)), \tag{7}$$

$$V_{\{s^c\}}\left(\omega(s^c)\right) = -\log P(c(s)|\omega(s^c)), \tag{8}$$

where the probabilities that the given foreground or background classes generate the $d(s)$ or $c(s)$ observation, were already defined in Section 3 by eq. 2, 3 and 5.
On the other hand, the labels at $S^*$ have no direct links with these measurements:

$$V_{\{s^*\}}\left(\omega(s^*)\right) = 0. \tag{9}$$

In order to get a smooth segmentation in each layer, the potential of an intra-layer clique $C_2 = \{s^i, r^i\} \in \mathcal{C}_2$, $i \in \{d, c, *\}$ has the following form [35]:

$$V_{C_2} = \theta\left(\omega(s^i), \omega(r^i)\right) = \begin{cases} -\delta^i & \text{if} \quad \omega(s^i) = \omega(r^i) \\ +\delta^i & \text{if} \quad \omega(s^i) \neq \omega(r^i) \end{cases} \tag{10}$$

with a constant $\delta^i > 0$.
As we concluded from the experiments in Section 3, a pixel is likely generated by the background process, if and only if in the $S^d$ and $S^c$ layers, at least one corresponding site has the label 'bg'. We introduce the following indicator function:

$$I_{\text{bg}} : S^d \cup S^c \cup S^* \to \{0,\ 1\},$$

where

$$I_{\text{bg}}(q) = \begin{cases} 1 & \text{if} \quad \omega(q) = \text{bg} \\ 0 & \text{if} \quad \omega(q) \neq \text{bg}. \end{cases}$$

With this notation the potential of an inter-layer clique $C_3 = \{s^d, s^c, s^*\}$ is:

$$V_{C_3}(\underline{\omega}_{C_3}) = \zeta(\omega(s^d), \omega(s^c), \omega(s^*)) = \begin{cases} -\rho & \text{if} \quad I_{\text{bg}}(s^*) = \max\left(I_{\text{bg}}(s^d), I_{\text{bg}}(s^c)\right) \\ +\rho & \text{otherwise.} \end{cases} \qquad (11)$$

with $\rho > 0$.
Therefore, the optimal MAP labeling $\widehat{\underline{\omega}}$, which maximizes $P(\widehat{\underline{\omega}}|\mathcal{F})$ (hence minimizes $-\log P(\widehat{\underline{\omega}}|\mathcal{F})$) can be calculated as:

$$\widehat{\underline{\omega}} = \text{argmin}_{\underline{\omega}\in\Omega} - \sum_{s\in S} \log P(d(s)|\omega(s^d)) - \sum_{s\in S} \log P(c(s)|\omega(s^c))$$

$$+ \sum_{C_2\in\mathcal{C}_2} V_{C_2}\left(\underline{\omega}_{C_2}\right) + \sum_{C_3\in\mathcal{C}_3} V_{C_3}\left(\underline{\omega}_{C_3}\right). \qquad (12)$$

The above energy minimization is performed with simulated annealing. (See Appendix B for details.) The final segmentation is taken as the labeling of the $S^*$ layer.

# 5 Parameter settings

In the following we define a possible grouping of the free parameters in the process: the first group is related to the correlation calculation and the second one to the potential functions.

## 5.1 Parameters related to the correlation window

The correlation window defined in Section 3 should not be significantly larger than the expected objects to ensure low correlation between an image part which contains an object and one from the same 'empty' area. We use a $9 \times 9$ pixel window in our experiments for images of size $320 \times 240$.
The *maximal offset* of the search window determines maximal parallax error, which can be compensated by the method. We note that in homogenous background, object motions with less than the offset parameter can be falsely detected as parallax errors. Therefore, at the given resolution, we use $\pm 3$ pixels for the maximal offset, and detect the moving objects whose displacement is larger.

### 5.2 Parameters of the potential functions

The singleton potentials are values of conditional density functions as it was defined in Section 3 by eq. 2, 3 and 5.
The Gaussian mean parameter ($\mu$) corresponds to the average gray value difference between the images caused by quick changes in the lighting conditions or in the camera white balance, the deviation ($\sigma$) depends on the noise. These parameters can be estimated by creating a histogram for D difference image, and estimating the parameters of the area close to the main peak of this histogram.
The Beta distribution parameters and the uniform values are determined from one image to another one by trial and error. We use $\alpha = 4.5$, $\beta = 1$ and $a_c = 0, b_c = 1$ for all image pairs (with the assumption that the gray values of the images are normalized between 0 and 1), while the optimal value of $a_d$ and $b_d$ shows significant differences in the different image sets. Using the '$2\sigma$-rule' proved to be a good initial approximation, namely $\frac{1}{b_d - a_d} = N(\mu + 2\sigma, \mu, \sigma)$. Here, following the Chebyshev inequality [7]:

$$P(|d(s) - \mu| > 2\sigma \mid \omega(s) = \mathrm{bg}) < \frac{1}{4}.$$

The parameters of the intra-layer potential functions, $\delta^d$, $\delta^c$ and $\delta^*$ influence the size of the connected blobs in the segmented images. Higher $\delta^i$ ($i \in \{d, c, *\}$) values result in more compact foreground regions, however, fine details of the silhouettes may be distorted that way. We have used in each layer $\delta^i = 0.7$ for test images with relatively small objects (e.g. 'balloon1' and 'Budapest' sets, introduced in Section 6.1), while $\delta^i = 1.0$ have been proved to be appropriate regarding images captured from lower altitude ('balloon2').
Parameter $\rho$ of the inter-layer potentials determines the strength of the relationship between the segmentation of the different layers. We have used $\rho = \delta^*$: this choice gives the same importance to the intra-layer smoothness and the inter-layer label fusion constraints.

## 6 Results

In this section, we validate our method via image pairs from different test sets. We compare the results of the three layer model with three reference methods first qualitatively, then using different quantitative measures. Thereafter, we test the significance of the inter layer connections in the joint segmentation model. Finally, we comment on the complexity of the algorithm.

### 6.1 Test sets

The evaluations are conducted using manually generated ground truth masks regarding different aerial images. We use three test sets which contain 83 (=52+22+9) image pairs. The time difference between the frames to compare is about 1.5-2 seconds. The 'balloon1' and 'balloon2' test sets contain image pairs from a video-sequence captured by a flying balloon, while in the set 'Budapest', we find different image pairs taken from a plane. For each test set, the model parameters are estimated over 2-5 training pairs and we examine the quality of the segmentation on the remaining test pairs.

### 6.2 Reference methods and qualitative comparison

We have compared the results of the proposed three-layer model to three other solutions. The first reference method (Layer1) is constructed from our model by ignoring the segmentation and the second observation layers. This comparison emphasizes the importance of using the correlation-peak features, since only the gray level differences are used here. The second reference is the method of Farin and With [6]. The third comparison is related to the limits of [23]: the optimal affine transform between the frames (which was automatically estimated in [23]) is determined in our comparative experiments in a supervised way, through manually marked matching points. Thereafter, we create the change map based on the gray level difference of the registered images with using a similar spatial smoothing energy term to eq. 10.
Fig. 7 shows the image pairs, ground truth and the segmented images with the different methods. For numerical evaluation, we perform first a pixel based, then an object based comparison.

### 6.3 Pixel based evaluation

Denote the number of correctly identified foreground pixels of the evaluation images by $TP$ (*true positive*). Similarly, we introduce $FP$ for misclassified background points, and $FN$ for misclassified foreground points.
The evaluation metrics consists of the *Recall* rate and the *Precision* of the detection.

$$\text{Recall} = \frac{TP}{TP+FN} \qquad \text{Precision} = \frac{TP}{TP+FP}$$

The results are presented in Table 1 for each image-set independently. In Table 1, we use the $F$-measure [37] which combines *Recall* ($R$) and *Precision* ($P$) in a single efficiency measure (it is the harmonic mean of $P$ and $R$):

$$F = \frac{2 \cdot R \cdot P}{R+P}. \tag{13}$$

Regarding the 'balloon1'/'balloon2'/'Budapest' test sets, the gain of using our method considering the $F$-measure is 26/35/16% in contrast to the Layer1 segmentation and 12/19/13% compared to Farin's method. The results of the frame global affine matching, even with manually determined control points, is 5/10/11% worse than what we get with the proposed model.

| Set | | Recall | | | | Precision | | | |
|---|---|---|---|---|---|---|---|---|---|
| Name | Cardi-nality | Layer1 | Farin's | Sup. affine | **3layer MRF** | Layer1 | Farin's | Sup. affine | **3layer MRF** |
| balloon1 | 52 | 0.83 | 0.76 | 0.85 | **0.92** | 0.48 | 0.74 | 0.79 | **0.85** |
| balloon2 | 22 | 0.86 | 0.68 | 0.89 | **0.88** | 0.35 | 0.64 | 0.65 | **0.83** |
| Budapest | 9 | 0.87 | 0.80 | 0.85 | **0.89** | 0.56 | 0.65 | 0.65 | **0.79** |

Table 1: Numerical comparison of the proposed method (3-layer MRF) with the results that we get without the correlation layer (Layer1) and Farin's method [6] and the supervised affine matching. Rows correspond to the three different test image-sets with notation of their cardinality (e.g. number of image-pairs included in the sets).

| Set | | F-rate | | | |
|---|---|---|---|---|---|
| Name | Cardi-nality | Layer1 | Farin's | Sup. affine | **3layer MRF** |
| balloon1 | 52 | 0.61 | 0.75 | 0.82 | **0.87** |
| balloon2 | 22 | 0.50 | 0.66 | 0.75 | **0.85** |
| Budapest | 9 | 0.68 | 0.71 | 0.73 | **0.84** |

Table 2: Numerical comparison of the proposed and reference methods via the $F$-rate. Notations are the same as in Table 1.

### 6.4 Object based evaluation

Although our method does not segment the individual objects, the presented change mask can be the input of an object detector module. It is important to know, how many object-motions are correctly detected, and what is the false alarm rate.
If an object changes its location, two blobs appear in the binary motion image, corresponding to its first and second positions. Of course, these blobs can overlap, or one of them may be missing, if an object just appears in the second frame, or if it leaves the area of the image between the two shots. In the following, we call one such blob an 'object displacement', which will be the unit in the object based comparison.
Given a binary segmented image, denote by $M_o$ (missing objects) the number of object displacements, which are not included in the motion silhouettes, while $F_o$ (false objects) is the number of the connected blobs in the silhouette images, which do not contain real object displacements, but their size is at least as large as one expected object. For the selected image pairs of Fig. 7, the numerical comparison to Farin's and the supervised affine method is given in Table 1. A limitation of our method can be observed in the 'Budapest' #2 image pair: the parallax distortion of a standing lamp is higher than the length of the correlation search window side, which results in two false objects in the motion mask. However, the number of missing and false objects is much lower than with the reference methods.

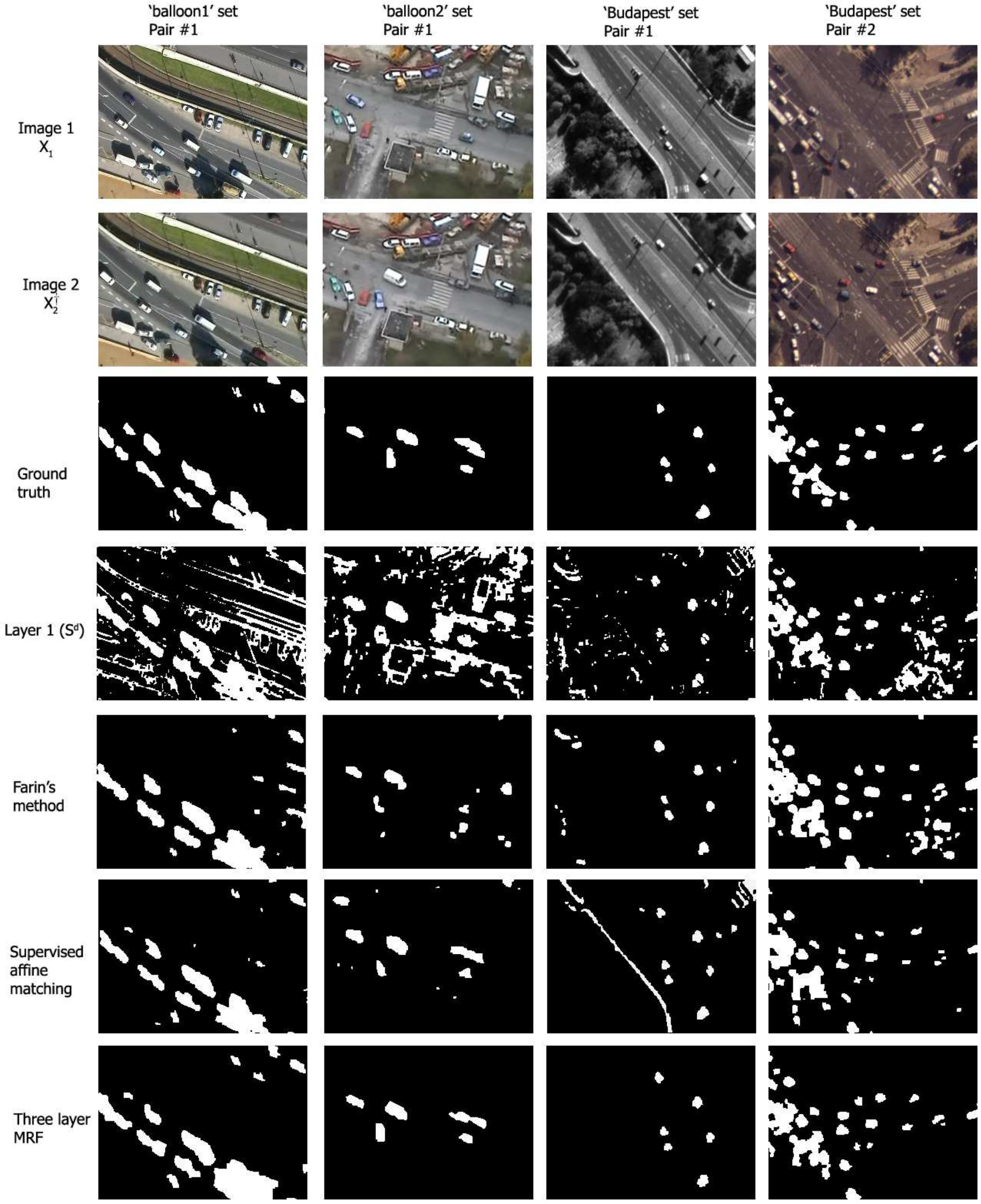


Figure 7: Test image pairs and segmentation results with different methods.

| Test pair | | $A_0$ | $M_o$ | | | $F_o$ | | |
|---|---|---|---|---|---|---|---|---|
| Set | No. | | Far. | Sup. aff. | **3lay. MRF** | Far. | Sup. aff. | **3lay. MRF** |
| balloon1 | #1 | 19 | 0 | 0 | **0** | 6 | 1 | **1** |
| balloon2 | #1 | 6 | 0 | 0 | **0** | 3 | 2 | **0** |
| Budapest | #1 | 6 | 1 | 0 | **0** | 7 | 7 | **0** |
| Budapest | #2 | 32 | 0 | 1 | **1** | 10 | 6 | **3** |
| | All | 63 | 3 | 1 | **1** | 26 | 16 | **4** |

Table 3: Object-based comparison of the proposed and the reference methods. $A_o$ means the number of all object displacements in the images, while the number of missing and false objects is respectively $M_o$ and $F_o$.

### 6.5 Significance of the joint segmentation model

In the proposed model, the segmentations based on the $d(.)$ and $c(.)$ features are not performed independently: they interact through the inter-layer cliques. Although similar approaches have been already used for different image segmentation problems [16]-[19], the significance of intra-layer connections should be justified with respect to the current task. Note, that increasing the number of connections in the MRF results in a more complex energy model (eq. 12), which increases the computational complexity of the method.
We demonstrate the role of the inter-layer cliques by comparing the proposed scheme with a sequential model, where first, we perform two independent segmentations based on $d(.)$ and $c(.)$ (i.e. we segment the $S^d$ and $S^c$ layers ignoring the inter-layer cliques), thereafter, we get the segmentation of $S^*$ by a per pixel AND operation on the $D$ and $C$ segmented images. In Fig. 8, we can observe that the separate segmentation gives noisy results, since in this case, the intra-layer smoothing terms do not take into account in the $S^*$ layer. Consequently, the proposed label fusion process enhances the quality of segmentation versus the sequential model.

### 6.6 Running speed

With C++ implementation and a Pentium desktop computer (Intel(R) Core(TM)2 CPU, 2GHz), processing $320 \times 240$ images takes $5-6$ seconds. For the main parts of the algorithm, the measured processing times are shown in Table 4. The calculation of the correlation map (i.e. the determination of the $c(.)$ feature in Section 3) and the MRF optimization (finding a good suboptimal labeling according to eq. 12 from Section 4) are detailed in Appendices A and B, respectively.

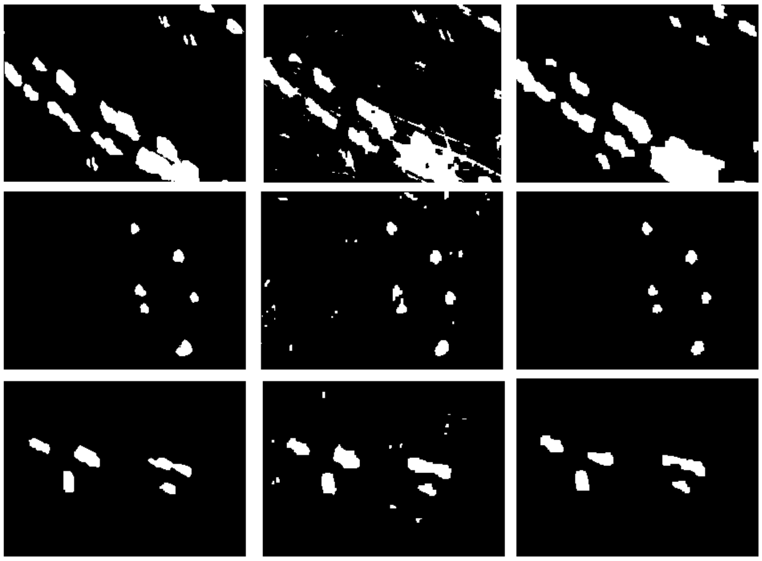

Figure 8: Illustration of the benefit of the inter layer connections in the joint segmentation. Col 1: ground truth, Col 2: results after separate MRF segmentation of the $S^d$ and $S^c$ layers, and deriving the final result with a per pixel AND relationship. Col 3. Result of the proposed joint segmentation model

| Procedure | FCS | PCH | Corr. map | MRF opt. |
|---|---|---|---|---|
| Time (sec) | 0.15 | 0.04 | 2.4 | 2.9 |

Table 4: Running time of the main parts of the algorithm. The calculation of the correlation map and the MRF optimization are detailed in Appendices A and B, respectively.

# 7 Applications

The proposed model can be used in different high level applications being developed by ongoing research projects.
The *Shape Modelling E-Team of the EU Project MUSCLE* is interested in learning shapes and recognizing shapes as a central part of image database indexing strategies. Its scope includes shape analysis and learning, prior-based segmentation and shape-based retrieval. In shape modelling, however, accurate silhouette extraction is a crucial preprocessing task.
The primary aim of the *Hungarian R&D Project ALFA* is to create a compact vision system that may be used as autonomous visual recognition and navigation system of unmanned aerial vehicles. In order to make long term navigational decisions the system has to evaluate the captured visual information without any external assistance. The civil use of the system includes large area security surveillance and traffic monitoring, since effective and economic solution to these problems is not possible using current technologies. The *Hungarian GVOP (3.1.1.-2004-05-0388/3.0)* tackles the problem of semantic interpretation, categorizing and indexing the video frames automatically. For all these applications, object motion detection provides significant information.

# 8 Conclusion

This report has addressed the problem of exploiting accurate change masks from image pairs taken by a moving camera. A novel three-layer MRF model has been proposed, which integrates the information from two different observations. The efficiency of the method has been validated through real-world aerial images, and its behavior versus three reference methods has been quantitatively and qualitatively evaluated.

# 9 Acknowledgement

This work was partially supported by the EU project MUSCLE (FP6-567752) and the Hungarian R&D Project ALFA. The authors would like to thank the MUSCLE Shape Modelling E-Team for financial support and to Xavier Descombes for his kind remarks and advices.

# References

[1] H. El–Askary, A. Agarwal, T. El–Ghazawi, M. Kafatos and J. Le Moigne, "Enhancing dust storm detection using PCA based data fusion," *Geoscience and Remote Sensing Symposium,* vol. 2 pp. 1424 – 1427, July 2005.

[2] S. T. Barnard, W. B. Thompson, "Disparity analysis of images," *IEEE Trans. on Pattern Analysis and Machine Intelligence,* vol. 2, pp. 333-340, 1980.

[3] J. Besag, "On the statistical analysis of dirty images," *Journal of Royal Statistics Society B*, vol. 48, No. 3, pp. 259–302, 1986.

[4] J-Y. Bouguet, "Pyramidal Implementation of the Lucas Kanade Feature Tracker: Description of the Algorithm", *Technical Report, Intel Corporation,* 1999.

[5] J. K. Cheng, T. S. Huang, "Image registration by matching relational structures," *Pattern Recognition*, vol. 17, pp. 149-159, 1984.

[6] D. Farin and P. With, "Misregistration Errors in Change Detection Algorithms and How to Avoid Them," in *Proc. International Conference on Image Processing (ICIP)*, vol. 2, pp. 438-441, Genoa, Italy, Sept. 2005.

[7] W. Feller, "An introduction to probability theory and its applications," *Wiley Series in Probability and Mathematical Statistics*, vol. 1, Second edition, p. 219, 1966.

[8] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images", *IEEE Trans. on Pattern Analysis and Machine Intelligence,* vol. 6, No. 6, pp. 721-741, Nov. 1984.

[9] R. Hartley and A. Zissermann, "Multiple View Geometry in Computer Vision," *Cambridge University Press*, Cambridge, 2000.

[10] H. Hirschmüller, P. R. Innocent, and J. Garibaldi, "Real-time correlation-based stereo vision with reduced border errors," *International Journal of Computer Vision*, vol. 47, no. 1/2/3, pp. 229–246, 2002.

[11] H. Hirschmüller, F.Scholten, and G. Hirzinger, "Stereo vision based reconstruction of huge urban areas from an airborne pushbroom camera (hrsc)," in *Proc. 27th DAGM Symposium*, (Vienna, Austria), pp. 58–66, LNCS 3663, Sept 2005.

[12] Q. Iqbal and J. K. Aggarwaly, "Feature Integration, Multi-image Queries and Relevance Feedback in Image Retrieval," *International Conference on Visual Information Systems,* Miami, Florida, pp. 467-474, Sep. 2003.

[13] Intel Corporation, "OpenCV documentation,"
`http://www.intel.com/technology/computing/opencv/index.htm`

[14] M. Irani and P. Anandan: "A Unified Approach to Moving Object Detection in 2D and 3D Scenes," *IEEE Trans. on Pattern Anal. Mach. Intell.*, vol. 20, No. 6., pp 577–589, 1998.

[15] P-M. Jodoin and M. Mignotte, "Motion Segmentation Using a K-nearest-Neighbor-Based Fusion Procedure, of Spatial and Temporal Label Cues," in *Proc. of International Conference on Image Analysis and Recognition*, pp. 778–788, Toronto, Canada, 2005.

[16] Z. Kato, T. C. Pong and G. Q. Song, "Multicue MRF Image Segmentation: Combining Texture and Color", in *Proc. of International Conference on Pattern Recognition*, vol. 1, pp. 660–663, Quebec, Canada, August 2002.

[17] Z. Kato, T. C. Pong and G. Q. Song, "Unsupervised segmentation of color textured images using a multi-layer MRF model," in *Proc. of International Conference on Image Processing,* vol. I, pp. 961–964, Barcelona, Spain, Sept. 2003.

[18] Z. Kato and T. C. Pong, "Video Object Segmentation Using a Multicue Markovian Model," in *Proc. Joint Hungarian-Austrian Conference on Image Processing and Pattern Recognition,* pp. 111–118, Veszprém, Hungary, May 2005.

[19] Z. Kato and T. C. Pong, "A Multi-Layer MRF Model for Video Object Segmentation," in *Proc. of Asian Conference on Computer Vision,* vol. LNCS 3852, Springer, pp. 953–962, Hyderabad, India, January 2006.

[20] Z. Kato and T. C. Pong, "A Markov Random Field Image Segmentation Model for Color Textured Images," *Image and Vision Computing,* vol. 24, No. 10, pp. 1103–1114, October 2006.

[21] Z. Kato, J. Zerubia, and M. Berthod, "Satellite Image Classification Using a Modified Metropolis Dynamics", in *Proc. International Conference on Acoustics, Speech and Signal Processing,* vol. 3, pp. 573–576, San-Francisco, USA, March 1992.

[22] S. Khan and M. Shah, "Object based segmentation of video using color, motion and spatial information", in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition,*, pp. 746–751, Hawaii, USA, December 2001.

[23] S. Kumar, M. Biswas and T. Nguyen, "Global motion estimation in spatial and frequency domain", in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing,* pp. 333–336, Montreal, Canada, May 2004.

[24] S. Kumar and U.B. Desai, "New algorithms for 3D surface description from binocular stereo using integration", *Journal of the Franklin Institute,* 331B, No. 5, pp. 531–554, 1994.

[25] R. Kumar, R. H. Sawhney, S. Samarasekera, S. Hsu, Hai Tao, Y. Guo, K. Hanna, A. Pope, R. Wildes, D. Hirvonen, M. Hansen and P. Burt, "Aerial video surveillance and exploitation," *Proceeding of the IEEE,* vol. 8, pp. 1518–1539, 2001.

[26] A. Kushki, P. Androutsos, K.N. Plataniotis and A.N. Venetsanopoulos, "Retrieval of images from artistic repositories using a decision fusion framework," *IEEE Trans. on Image Processing*, vol. 13, No. 3, pp. 277–292, 2004.

[27] L. Li and M.K.H. Leung, "Integrating intensity and texture differences for robust change detection," *IEEE Trans. on Image Processing,* vol. 11, no. 2, pp. 105–112, February 2002.

[28] B. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," in *Proc. of 7th International Joint Conference on Artificial Intelligence*, pp. 674–679, Vancouver, BC, Canada, August 1981.

[29] L. Lucchese, "Estimating Affine Transformations in the Frequency Domain," in *Proc. Int. Conf. on Image Processing*, vol. II, pp. 909–912, Thessaloniki, Greece, Sept. 2001.

[30] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, "Equation of State Calculations by Fast Computing Machines," *Journal of Chemical Physics*, vol. 21, pp. 1087–1092, 1953.

[31] I. Miyagawa and K. Arakawa, "Motion and shape recovery based on iterative stabilization for modest deviation from planar motion," *IEEE Trans. on Pattern Analysis and Machine Intelligence,* vol. 28, No. 7, pp. 1176–1181, 2006.

[32] J. M. Odobez and P. Bouthemy, "Detection of multiple moving objects using multiscale MRF with camera motion compensation," in *Proc. Int. Conf. on Image Processing*, vol. 2, pp. 257–261, Austin, Texas, USA, 1994.

[33] D.T. Oram, "Rectification for any epipolar geometry," in *Proc. British Machine Vision Conference,* pp. 653–662, London, UK, 2001.

[34] R. Pless, T. Brodsky and Y. Aloimonos, "Detecting independent motion: The statistics of temporal continuity." *IEEE Trans. on Pattern Analysis and Machine Intelligence,* vol. 22, No. 8, pp. 68–73, 2000.

[35] R. Potts, "Some generalized order-disorder transformation," *Proceedings of the Cambridge Philosophical Society,* vol. 48, pp. 106–109, 1952.

[36] B. Reddy and B. Chatterji, "An FFT-based technique for translation, rotation and scale-invariant image registration", *IEEE Trans. on Image Processing*, vol. 5, no. 8, pp. 1266–1271, 1996.

[37] C. J. Van Rijsbergen, "Information Retrieval," 2nd edition, London, Butterworths, 1979.

[38] E. Saber and A. Tekalp, "Integration of color, edge, shape, and texture features for automatic region-based image annotation and retrieval," *Journal of Electronic Imaging,* vol. 7, no. 3, pp. 684–700, July 1998.

[39] H.S. Sawhney, Y. Guo and R. Kumar, "Independent Motion Detection in 3D Scenes", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, No. 10, pp. 1191–1199, 2000.

[40] C. Sun, "Fast Stereo Matching Using Rectangular Subregioning and 3D Maximum-Surface Techniques", *International Journal of Computer Vision*, vol. 47. No. 1, pp. 99–117, 2002.

[41] Z. Szlávik, T. Szirányi and L. Havasi, "Stochastic view registration of overlapping cameras based on arbitrary motion", *IEEE Trans. on Image Processing*, vol. 16, No. 3, pp. 710–720, 2007.

[42] P. Viola and M. Jones, "Rapid Object Detection Using a Boosted Cascade of Simple Features," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition,* vol. 1, pp. 511–518, Hawaii, USA, December 2001.

[43] Y. Wang, K-F. Loe and J-K. Wu, "A Dynamic Conditional Random Field Model for Foreground and Shadow Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 2, pp. 279–89, 2006.

[44] J. Weng, N. Ahuja and T. S. Huang, "Matching two perspective views," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 806–825, 1992.

[45] Z. Zhang, R. Deriche, O. Faugeras and Q-T. Luong, "A Robust Technique for Matching Two Uncalibrated Images Through the Recovery of the Unknown Epipolar Geometry," *Artifical Intellignece,* vol. 78, pp. 87–119, 1995.

# Appendices

## A Calculation of the correlation map

In this appendix, we introduce an effective algorithm to calculate the correlation map used by the $c(.)$ feature (Section 3, eq. 4). The algorithm uses box filtering technique with the integral image trick similarly to [40]. However, since our method does not assume accurate epipolar matching, the region where we search for pixel correspondences is a rectangle instead of a line, like in [40], which works with epipolar rectified images [33]. On the other hand, exploiting that due to the preliminary registration and the expected low parallax distortion the corresponding pixels are relatively close to each other, we can also extend the box matching technique to search in the moving window. Here, we need a 4D representation of the local correlation map, instead of 3D [40].

### A.1 Integral image

Several features over a given rectangular neighborhood can be computed very rapidly using an intermediate representation for the image which is called the integral image [42].
Given an image $\Lambda \leftarrow S$, its integral image $\mathcal{I}_\Lambda \leftarrow S$ is defined as:

$$\mathcal{I}_\Lambda(x,y) = \sum_{i=1}^{x}\sum_{j=1}^{y}\Lambda(i,j).$$

With notation $\zeta(x,0) = 0$ and $\mathcal{I}_\Lambda(0,y) = 0$, $x = 1 \dots S_x$, $y = 1 \dots S_y$:

$$\zeta(x,y) = \zeta(x,y-1) + \Lambda(x,y),$$

$$\mathcal{I}_\Lambda(x,y) = \mathcal{I}_\Lambda(x-1,y) + \zeta(x,y),$$

the integral image can be computed in one pass over the original image.
With the integral-trick, the sum of the pixel values over a rectangular window can be computed via three additional operations independently from the window size $(c-a)\times(d-b)$:

$$\sum_{i=a}^{c}\sum_{j=b}^{d}\Lambda(i,j) = \mathcal{I}_\Lambda(c,d) - \mathcal{I}_\Lambda(a-1,d) - \mathcal{I}_\Lambda(c,b-1) + \mathcal{I}_\Lambda(a-1,b-1).$$

### A.2 Correlation

Let $\Upsilon_1$ and $\Upsilon_2$ be two $l_w \times l_h$ sized 2 dimensional real arrays, with mean values $\overline{\Upsilon}_1$ and $\overline{\Upsilon}_2$, respectively. Their normalized cross correlation is defined by:

$$\mathrm{Corr}(\Upsilon_1,\Upsilon_2) = \frac{\sum_{x=1,y=1}^{l_w,l_h}\left(\Upsilon_1(x,y)-\overline{\Upsilon}_1\right)\left(\Upsilon_2(x,y)-\overline{\Upsilon}_2\right)}{\sqrt{\sum_{x=1,y=1}^{l_w,l_h}\left(\Upsilon_1(x,y)-\overline{\Upsilon}_1\right)^2\sum_{x=1,y=1}^{l_w,l_h}\left(\Upsilon_2(x,y)-\overline{\Upsilon}_2\right)^2}}$$

### A.2.1 Local correlation map

Denote by $\mathcal{P}$ the set of images over $S$. Denote by $\Lambda_1, \Lambda_2 \in \mathcal{P}$ two images, $w_x$, $w_y$, $l_w$ and $l_h$ are scalars. $t_{\text{win}} = (2l_w + 1)(2l_h + 1)$ is the size of the comparison window.
Denote by $\Upsilon_1^{x,y}$ a $(2l_w + 1) \times (2l_h + 1)$ sized subimage of $\Lambda_1$, whose center is located at $[x, y]$. For sake of simplicity, we use also negative indices for identifying the elements of $\Upsilon_1^{x,y}$. Hence,

$$\Upsilon_1^{x,y}(i, j) = \Lambda_1(i + x, j + y),$$
$$-l_w \leq i \leq l_w, \;\; -l_h \leq j \leq l_h.$$

$\overline{\Upsilon_1^{x,y}}$ denotes the average of the elements in $\Upsilon_1^{x,y}$. $\Upsilon_2^{x,y}$ is defined similarly.

**Definition 1** ***(Local correlation map)*** *The local correlation map asserts a* $(2w_x + 1) \times (2w_y + 1)$ *array,* $C^{x,y}$ *to each pixel* $s = [x, y]$*:*

$$C^{x,y}(m, n) = \text{Corr}(\Upsilon_1^{x,y}, \Upsilon_2^{x+m,y+n}),$$
$$-w_x \leq m \leq w_x, -w_y \leq n \leq w_y.$$

To get an efficient computation, we introduce the following notations:
For a given image $\Lambda$, denote by $\Lambda^{\text{sq}}$ the “squared image”:

$$\Lambda^{\text{sq}}(x, y) = [\Lambda(x, y)]^2 .$$

Denote by $\Lambda^{m,n}$ the “offset image”:

$$\Lambda^{m,n}(x, y) = \Lambda(x + m, y + n).$$

Denote by $\mathcal{M} : \mathcal{P} \times \mathbb{N} \times \mathbb{N} \to \mathbb{R}$ the local average functional of a given image over $S$:

$$\mathcal{M}\{\Lambda, x, y\} = \frac{1}{t_{\text{win}}} \sum_{i=-l_w}^{l_w} \sum_{j=-l_h}^{l_h} \Lambda(x + i, y + j).$$

If the $\mathcal{I}_\Lambda$ integral image is available, $\mathcal{M}\{\Lambda, x, y\}$ can be computed with 3 addition and 1 division operations:

$$\mathcal{M}\{\Lambda, x, y\} = \frac{1}{t_{\text{win}}} [\mathcal{I}_\Lambda(x + l_w, y + l_h) + \mathcal{I}_\Lambda(x - l_w - 1, y - l_h - 1) -$$
$$-\mathcal{I}_\Lambda(x - l_w - 1, y + l_h) - \mathcal{I}_\Lambda(x + l_w, y - l_h - 1)].$$

We also introduce the following notations:

$$M_1(x, y) = \mathcal{M}\{\Lambda_1, x, y\}, \quad M_2(x, y) = \mathcal{M}\{\Lambda_2, x, y\},$$

$\Lambda_*^{m,n}$ image is given by

$$\Lambda_*^{m,n}(x, y) = \Lambda_1(x, y)\Lambda_2^{m,n}(x, y), \qquad \forall [x, y] \in S,$$

and

$$M_*^{m,n}(x,y) = \mathcal{M}\{\Lambda_*^{m,n}, x, y\}.$$

$$\mathcal{B}(\Lambda, x, y) = \sum_{i=-l_w}^{l_w} \sum_{j=-l_h}^{l_h} \left(\Lambda(x+i, y+j) - \mathcal{M}\{\Lambda, x, y\}\right)^2 =$$

$$= \sum_{i=-l_w}^{l_w} \sum_{j=-l_h}^{l_h} \Lambda^{\mathrm{sq}}(x+i, y+j) - 2\mathcal{M}\{\Lambda, x, y\} \sum_{i=-l_w}^{l_w} \sum_{j=-l_h}^{l_h} \Lambda(x+i, y+j) + t_{\mathrm{win}}[\mathcal{M}\{\Lambda, x, y\}]^2 =$$

$$= t_{\mathrm{win}} \left(\mathcal{M}\{\Lambda^{\mathrm{sq}}, x, y\} - [\mathcal{M}\{\Lambda, x, y\}]^2\right)$$

On the other hand,

$$\mathcal{A}(x, y, m, n) =$$

$$= \sum_{i=-l_w}^{l_w} \sum_{j=-l_h}^{l_h} (\Lambda_1(x+i, y+j) - M_1(x,y))(\Lambda_2(x+m+i, y+m+j) - M_2(x+m, y+m)) =$$

$$= \sum_{i=-l_w}^{l_w} \sum_{j=-l_h}^{l_h} \Lambda_1(x+i, y+j)\Lambda_2(x+m+i, y+m+j) -$$

$$-M_1(x,y) \sum_{i=-l_w}^{l_w} \sum_{j=-l_h}^{l_h} \Lambda_2(x+m+i, y+m+j) -$$

$$-M_2(x+m, y+m) \sum_{i=-l_w}^{l_w} \sum_{j=-l_h}^{l_h} (\Lambda_1(x+i, y+j)) + t_{\mathrm{win}} M_1(x,y) M_2(x+m, y+m) =$$

$$= t_{\mathrm{win}} \left(M_*^{m,n}(x,y) - M_1(x,y) M_2(x+m, y+m)\right).$$

With these notations, the local correlation map is determined by:

$$C^{x,y}(m,n) = \frac{\mathcal{A}(x, y, m, n)}{\sqrt{\mathcal{B}(\Lambda_1, x, y) \cdot \mathcal{B}(\Lambda_2, x+m, y+n)}}.$$

Finally, the steps of the algorithm which calculates the correlation map, and the $c(.)$ feature (defined in Section 3) are listed in Fig. 9.

| | |
|---|---|
| 1. | For $-w_x \leq m \leq w_x,\ -w_y \leq n \leq w_y$: |
| | • Calculate $\Lambda^{m,n}$ |
| | • Calculate $\Lambda^{m,n}_*$ |
| | • Calculate the integral image of $\Lambda^{m,n}_*$. |
| 2. | Calculate the integral images of $\Lambda_1$, $\Lambda_2$, $\Lambda_1^{\mathrm{sq}}$ and $\Lambda_2^{\mathrm{sq}}$. |
| 3. | For all $x$,$y$: |
| | • Calculate $M_1(x,y)$ and $M_2(x,y)$. |
| | • Calculate $\mathcal{B}(\Lambda_1,x,y)$ and $\mathcal{B}(\Lambda_2,x,y)$. |
| 4. | For all $x$,$y$: |
| | • Calculate $C^{x,y}(m,n)$ for all $-w_x \leq m \leq w_x,\ -w_y \leq n \leq w_y$. |
| | • Store the maximal correlation value (over $m$, $n$): with $s=[x,y], c(s) = \max_{m,n} C^{x,y}(m,n)$ |

Figure 9: Algorithm for a efficient determination of the correlation feature $c(.)$. Notations are defined in Section 3 and Appendix A.

| Window size ($W$) | $3\times3$ | $5\times5$ | $7\times7$ | $9\times9$ | $11\times11$ |
|---|---|---|---|---|---|
| Time (sec) | 0.5 | 1.1 | 2.4 | 4.2 | 6.3 |

Table 5: Processing time of the algorithm of Fig. 9 as a function of the search window sizes ($W$), using $320\times240$ images, C++ implementation and a Pentium desktop computer (Intel(R) Core(TM)2 CPU, 2GHz)

#### A.2.2 Complexity

Denote by $W=(2w_x+1)\times(2w_y+1)$ the size of the search window, $t_{\mathrm{win}}=(2l_h+1)\times 2(l_w+1)$ is the size of the correlation window, $\mathcal{S}$ is the size of the image. With a naive straightforward solution (without using the integral trick), the process needs $10\mathcal{S}\cdot W\cdot t_{\mathrm{win}}+2\mathcal{S}\cdot W$ operations, while the improved version uses only $10\mathcal{S}\cdot W+37\mathcal{S}$ operations. Hence, the complexity of the improved method does not depend on the *correlation* window size $t_{\mathrm{win}}$. For some *search* window sizes ($W$), we show the processing time in Table 5.
In the tests of Section 6, we have used $W=7\times7$ pixel search windows. If s larger $W$ is necessary, we can speed up the method with multi-resolution techniques [24]. If the fundamental matrix can be extracted (i.e. the PHC method works [45]), the $(2w_x+1)\times(2w_y+1)$ pixel rectangular search window is restricted to a section in the corresponding epipolar line [9] (see also Fig. 10).

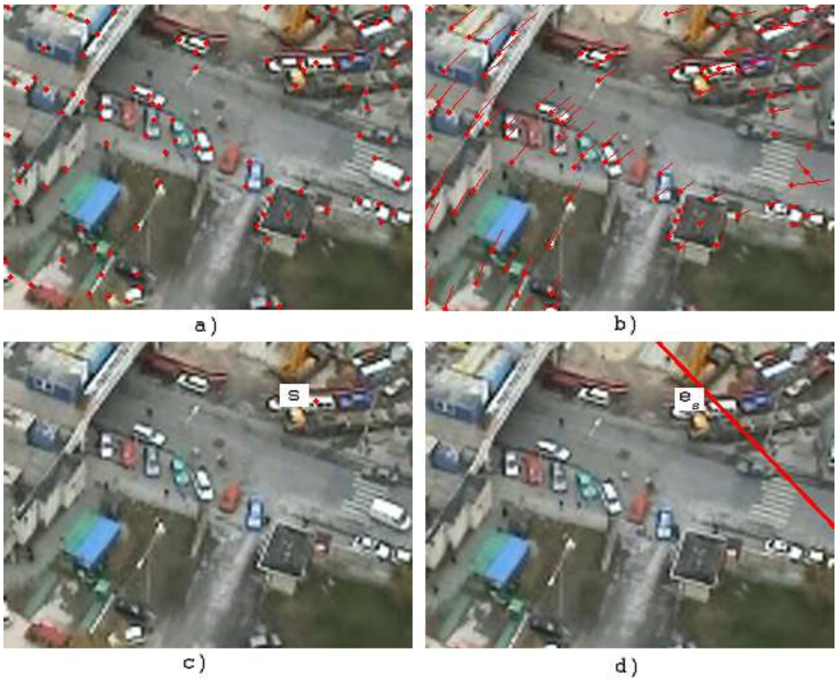


Figure 10: Illustration on how the PCH algorithm can restrict the correlation search window to a line. a) first input image ($X_1$) with the detected corner points b) result of the feature tracker [4] in $X_2$ for the previous corner pixels. The global motion is estimated based on the 2D displacement vectors corresponding to the corner points: the fundamental matrix, and the epipoles are calculated [9][13]. c) a selected pixel $s$ in $X_1$ and d) the corresponding epipolar line $e_s$ in $X_2$. For a given pixel $s$ in $X_1$, the corresponding pixel in $X_2$ must be located in line $e_s$. Note: as stated in Section 2.4, the PCH may fail for some inputs, however, as demonstrated here, it is efficient for test set 'balloon2', where the number of moving objects is quite low.

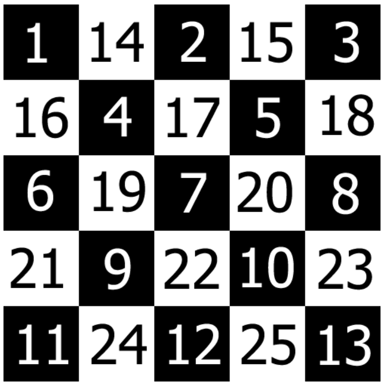


Figure 11: Ordinal numbers of the sites in a $5 \times 5$ layer according to the 'checkerboard' scanning strategy

# B MRF optimization

In MRF applications, the quality of the segmented images depends on:

- the appropriate data model structure and the probabilistic a priori model of the classes,
- the optimization technique which finds a good global labeling (considering eq. 12 in Section 4 in this research report). It is a key point, since the global optimum can only be reached usually by computationally expensive methods, such as [30].

We use the Modified Metropolis (MMD) [21] algorithm in this research report, since we have found it is nearly as efficient but significantly quicker than the original Metropolis [30]. We give the detailed pseudo code of the MMD adapted to the three layer segmentation model in Fig. 12. If we use ICM with our model [3], its processing time is negligible compared to the other parts of the algorithm, in exchange for some degradation in the segmentation results.

1. Pick up randomly an initial configuration $\underline{\omega}$, with $k := 0$ and $T := T_0$.
2. Denote by $|Q|$ the number of sites in the three-layer model. Assign to each site a unique ordinal number between 1 and $|Q|$, applying the 'checkerboard' scanning strategy (Fig. 11) for the consecutive layers. Let $j := 1$.
3. Let $q$ the $j^{\text{th}}$ site, $i \in \{d, c, *\}$ is the layer which contains $q$, while $s \in S$ is the corresponding pixel in the image lattice: $q = s^i$.
4. Denote the label of $q$ in $\underline{\omega}$ by $\omega(q)$. Flip the label of $q$ and denote it by $\breve{\omega}(q)$.
5. Compute $\Delta U$ as follows:

$$\Delta U := \Delta U_1 + \Delta U_2 + \Delta U_3, \quad \text{where}$$

a. Calculate $\Delta U_1$ as:

$$\Delta U_1 := \begin{cases} \log P\left(d(s)|\omega(q)\right) - \log P\left(d(s)|\breve{\omega}(q)\right) & \text{if } \ i = d \quad \text{(eq. 7)}, \\ \log P\left(c(s)|\omega(q)\right) - \log P\left(c(s)|\breve{\omega}(q)\right) & \text{if } \ i = c \quad \text{(eq. 8)}, \\ 0 & \text{if } \ i = * \quad \text{(eq. 9)} \end{cases}$$

b. Using eq. 10, calculate $\Delta U_2$ as:

$$\Delta U_2 := \sum_{r \in \Phi_s} \theta\left(\breve{\omega}(s^i), \omega(r^i)\right) - \theta\left(\omega(s^i), \omega(r^i)\right).$$

c. Denote by $\zeta_0 = \zeta\left(\omega(s^d), \omega(s^c), \omega(s^*)\right)$ (eq. 11). Calculate $\Delta U_3$ as:

$$\Delta U_3 := \begin{cases} \zeta\left(\breve{\omega}(q), \omega(s^c), \omega(s^*)\right) - \zeta_0 & \text{if } \ i = d, \\ \zeta\left(\omega(s^d), \breve{\omega}(q), \omega(s^*)\right) - \zeta_0 & \text{if } \ i = c, \\ \zeta\left(\omega(s^d), \omega(s^c), \breve{\omega}(q)\right) - \zeta_0 & \text{if } \ i = * \end{cases}$$

9. Update the label of $q$:

$$\omega(q) := \begin{cases} \breve{\omega}(q) & \text{if } \ \log \tau \leq -\frac{\Delta U}{T}, \\ \omega(q) & \text{otherwise.} \end{cases}$$

where $\tau$ is a constant threshold ($\tau \in (0, 1)$).

10. If $j < |Q|$: $\{j := j + 1$ and goto step 3.$\}$
11. Set $T := T_{k+1}$, $k := k + 1$, $j := 1$ and goto step 3, until convergence (i.e. the number of the changed labels between the $k^{\text{th}}$ and $(k+1)^{\text{th}}$ iteration is lower than a threshold.)

Figure 12: Pseudo-code of the Modified Metropolis algorithm used for the current task. Corresponding notations are given in Section 2, 3, 4 and in Appendix B. In the tests, we used $\tau = 0.3$, $T_0 = 4$, and an exponential heating strategy: $T_{k+1} = 0.96 \cdot T_k$

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399