Moving Object Detection in Real-Time Using Stereo from a Mobile Platform

Maxime Derome, Aurelien Plyer, Martial Sanfourche, Guy Le Besnerais

Moving Object Detection in Real-Time using Stereo from a Mobile Platform

Maxime Derome, Aurelien Plyer, Martial Sanfourche, Guy Le Besnerais

Information Processing and Modeling Department, ONERA - The French Aerospace Lab,
Chemin de la Huniere, Palaiseau, 91123, France
E-mails: firstname.name@onera.fr

This paper presents a mobile object detection algorithm which performs with two consecutive stereo images. Like most motion detection methods, the proposed one is based on dense stereo matching and optical flow estimation. Noting that the main computational cost of existing methods is related to the estimation of optical flow, we propose to use a fast algorithm based on Lucas-Kanade paradigm. We then derive a comprehensive uncertainty model by taking into account all the estimation errors occurring during the process. In contrast with most previous works, we rigorously expand the error related to vision based ego-motion estimation. Finally we present a comparative study of performance on the challenging KITTI dataset which demonstrates the effectiveness of the proposed approach.

Keywords: Stereo; Motion detection; Real-time; Estimation error modeling; Dynamic environment; Image prediction.

1. Introduction

1.1. *Context and Problem Statement*

Understanding complex environment in presence of dynamic objects is crucial for autonomous robotics. Such situation awareness could benefit to Advanced Driver Assistance Systems (ADAS) as well as Search And Rescue (SAR) missions. One may also be contemplating, in a near future, autonomous assistant robots moving around during an exhibition to inform visitors (cf. Fig 1). Vision sensors are particularly suited for this task as they are cheap, lightweight, and can provide, through dedicated fast algorithms, both scene perception and ego-motion estimation. Besides, using a stereo rig enables 3D reconstruction of the scene at each frames, which can be used for mobile object detection. In the design of an embedded mobile object detection process, three main constraints have to be accounted for: real-time processing, high reactivity, and precise management of measurement and estimation errors to assess the reliability of the decisions. We address these three constraints in our work. We propose a new detection system which uses very fast algorithms for the low-level operations (stereo matching and optical flow (OF) estimation). The decision is based on the processing of two consecutive stereo images only. This features allows to maximize the reactivity of the system and also eases the modelling of error propagation. This last issue is rigorously addressed here thanks to a first order model based on the Implicit Function Theorem.

1.2. *Related Works*

Different approaches have been proposed to address the understanding of dynamic scenes from stereo-vision data. Algorithms based on sparse sets of feature points have been used in temporally integrated framework [1], or in graphical approaches to segment stereo-images according to 3D motion consistency [2]. However, because of their sparseness, these methods provide limited coverage of the scene.

A great deal of work has also been done using dense stereo-vision algorithms. Dense stereo provides the instantaneous 3D structure of the scene. It can be coupled with visual odometry (VO) that computes the camera rotation and translation [R,T] between two frames. From these informations, the scene geometry in a new camera frame can be predicted under the hypothesis of a static world. The discrepancies between the new observation and this prediction reveal the independent motions and are cues for the detection of moving objects. Detection then stems from thresholding some residual field.

Depending on the residual value which is used, or equivalently on the quantity which is predicted, two approaches can be distinguished. One can either synthesize a predicted image using previous image intensity (an approach which will be denoted by *image prediction methods* in the following) or directly predict geometrical quantities such as 3D points coordinates, optical flow (OF) and disparity (*direct methods*).

Direct methods have been applied with different residual values in the literature. For instance, [4],[5] and [6] consider the differences between observed and predicted 3D

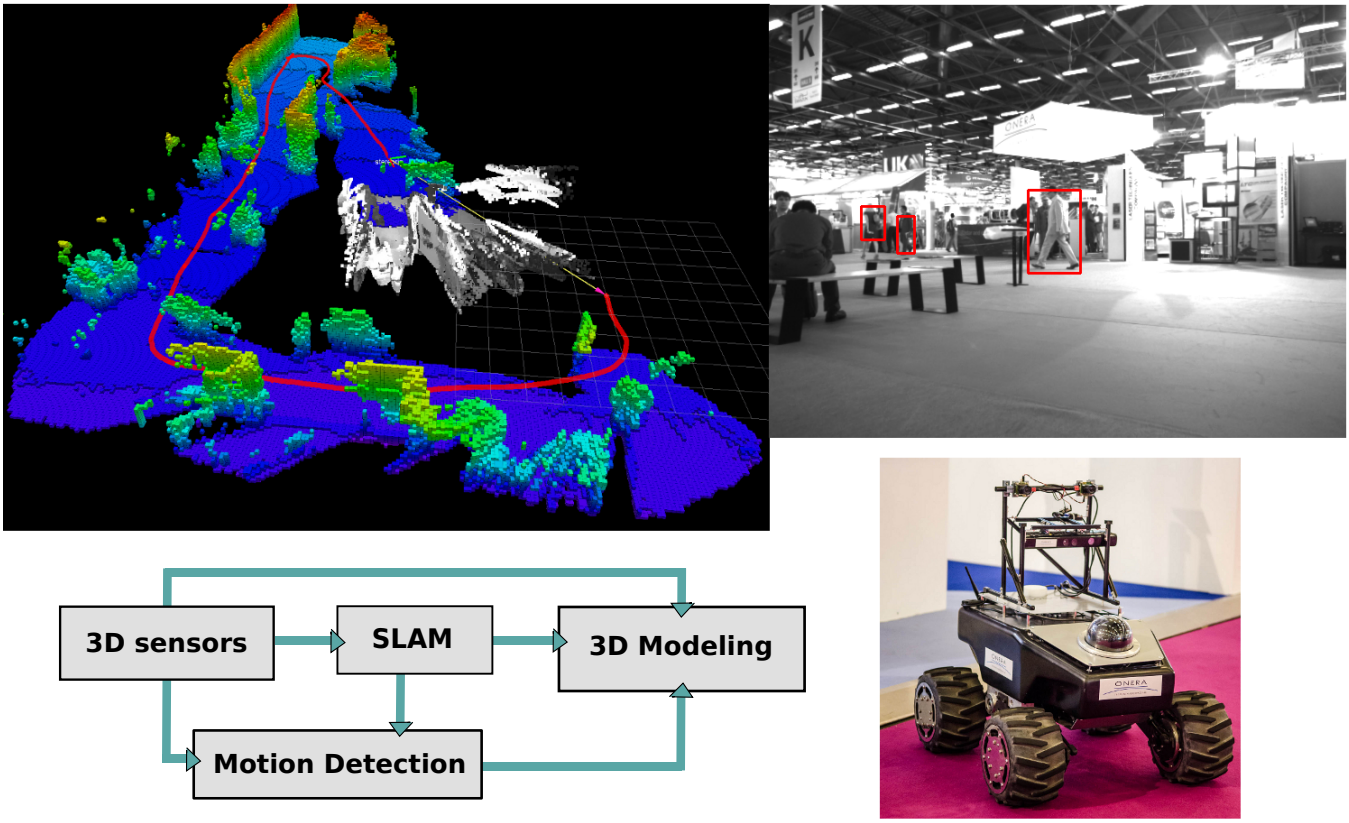

Fig. 1. Possible scenario: an autonomous assistant robot informing the visitors of an exhibition. Such a system would rely on an online ego-localisation and environment mapping system like 3DSCAN [3] (top left) enriched by the proposed detection algorithm (top right) and runs on a mobile robot equipped with a stereorig and a compter with a GPU. These results were obtained from data acquired during an exhibition with the robot shown in the bottom right corner.

points — a vector field which is called *Scene Flow*. In [5] the authors reduce Scene Flow noise using a Kalman filter for each pixel, with a state vector made of 3D position and velocity. Unfortunately, such temporal filtering reduce the system reactivity, since multiple frames are required for the Kalman filters to converge. Furthermore, this model may encounter difficulties with non-uniformly moving objects. In [6] the authors include disparity changes estimation in a variational minimization framework that also computes OF. Variational minimization methods are well known in the field of OF estimation, as they provide smooth and accurate solutions. But they require several solver iterations to converge to a good solution. Hence, in practice they are not real-time for an embedded system using high resolution stereo images. Other direct methods consider residual values expressed in the image space: OF and disparity in [7], OF alone in [8].

Alternatively, image prediction methods have been investigated. Dense comparison between observed and predicted image can be done by computing OF [9], or by evaluation of some similarity index within a small neighbourhood of the current pixel: [10] uses Sum of Absolute Differences (SAD) while a Zero-mean version (ZSAD) appears in [11].

Except for [10] and [11], all previous approaches rely on the computation of some 2D or 3D residual field (which we denote by M in the following), and the thresholding of a pixelwise motion likelihood written as a weighted norm of M:

$$\xi(M) = \sqrt{M^T \Sigma_M^{-1} M}. \quad (1)$$

If the covariance matrix Σ_M models accurately the uncertainty about the residual field M, criterion (1) is called a Mahalanobis distance, and leads to optimal decision. The main issue is that the residual M depends on several variables (disparity fields, OF, [R,T]) which stem from complex estimation processes. Estimating the resulting uncertainty on M is very difficult and requires some simplification. First attempts [7,10] simply considered $\Sigma_M = Id$, which leads to poor results. A formulation of Σ_M depending on disparity and optical flow is proposed in [6], based on residual minimization energy. However, the authors disregard the rotation R that is assumed equal to the identity matrix Id_3, and model only camera translation uncertainty, which is a rather crude hypothesis, even in the context of urban navigation. A Bayesian formulation of OF error covariance Σ_{OF}

is used in [8] to model Σ_M. The authors also consider [R,T] uncertainty, but assume independent rotational and translational errors without explicit mention of the ego-motion estimation process. In [9], the approach relies on first order expansion of the image displacement field with respect to the angular and translational velocity vectors $[\Omega, V]$. Both Σ_Ω and Σ_T are chosen as constant diagonal matrices which are evaluated a priori using a synthetic video. The first order expansion puts limits on the dynamic of the vehicle or on the framerate. Besides, the authors of [9] do not fully account for the $[\Omega, V]$ uncertainty but use 3σ bounds on the errors in the subsequent expressions. Finally, reference [4] suggests an heuristic to derive an approximate covariance matrix from the least-squares criterion (16) but does not explicitly include the error budget for disparity and temporal matching estimation. As a conclusion, to our knowledge, there is no previous paper presenting a comprehensive analysis of errors, especially regarding the uncertainty over ego-motion parameters [R,T].

1.3. *Contribution and Outline of the Paper*

Our contribution is threefold. We present a moving object detection system based on eFOLKI, a newly proposed fast OF method [12] which allows real-time processing of large images. We present a comprehensive analytical formulation of the uncertainty model of both direct and image prediction methods. In particular, we account for the fact that [R,T] parameters derive from the optimization of an ego-motion criterion where image measurements (point matches) are also involved. This indirect relationship is rigourously handled thanks to the Implicit Function Theorem. Finally, we conduct a comparison of various methods and error models through an evaluation protocol based on challenging KITTI datasets [13]. This experimental study demonstrates the efficiency of the proposed image prediction method and the benefit of the presented error model.

This work has been partly published in [14]. This paper presents in more details, the different residual fields that can be considered for the motion detection, as well as their uncertainty error model. The detection stage is slightly modified, and a deeper quantitative analysis of the behaviours of the methods is carried on KITTI datasets.

The paper is organized as follows. Section 2 describes the detection process and discuss low level operations and choice of the residual value. The uncertainty model is detailed in Sec. 3. The evaluation protocol and experimental results are presented in Sec. 4.

2. System Description

2.1. *Overview*

Fig. 2 presents a global overview of the moving object detection pipeline whose stages are illustrated in Fig. 3. Independently moving objects are detected by analysing two consecutive stereo images. Dense stereo is computed for each stereo acquisition time and dense optical flow is computed between successive times: these costly low-level operations are discussed in the following. We use the pose estimation algorithm presented in [15] which can run at 20Hz on a single core of an embedded CPU: some details on this estimation process will be reviewed in Sec. 3. With these informations we compute a residual field M that is null under static scene assumption. Given the error covariance matrix Σ_M derived according to some model of uncertainty, see Sec. 3, the Mahalanobis distance $\xi(M)$ of Eq. (1) is computed and thresholded. Bounding boxes are then fitted to the detected areas. In this section, we focus on low-level operations and choice of the residual value.

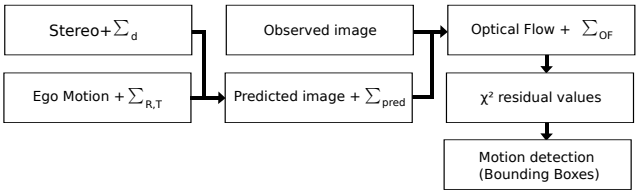

Fig. 2. Motion detection pipeline (see explanations in the text).

2.2. *Low-level Operations*

Several papers present algorithmic choices to reduce stereo computation time. A major breakthrough here is the publication of Semi Global Matching (SGM) [16], a dense stereo algorithm that can be implemented on FPGA [17] and run at 25Hz on images of 740x480 pixels for a disparity range of 128. However, for larger images and wider disparity range, the real-time capability of SGM can be questioned. Alternatively, one may consider a simple Block Matching (BM) algorithm that exhaustively searches stereo matches along the epipolar line. BM runs in real-time without needing a FPGA. The choice between SGM and BM is discussed in [9], and their relative performances evaluated. The inconvenient of BM is that not only the disparity map is less accurate, but it is also often unavailable on large regions (cf Fig. 5). This calls into question the benefit of dense methods, which is maximal coverage of the scene. As an in-between solution, we may finally consider ELAS (Efficient LArge-scale Stereo) described in [18]. This stereo algorithm relies on a set of support points robustly matched from the left to the right image, that build a disparity prior which is embedded in a graphical model used for the dense stereo estimation. ELAS can perform really fast for small images, on a single CPU.

Perhaps surprisingly, in previous works there are few discussions about the choice of the optical flow estimation algorithm. To our knowledge, all references use variational methods based on the framework originally presented by Horn and Schunck [19]. For instance, *Combined Local-Global Method* [20] is used in [8], while TV-L1 approaches

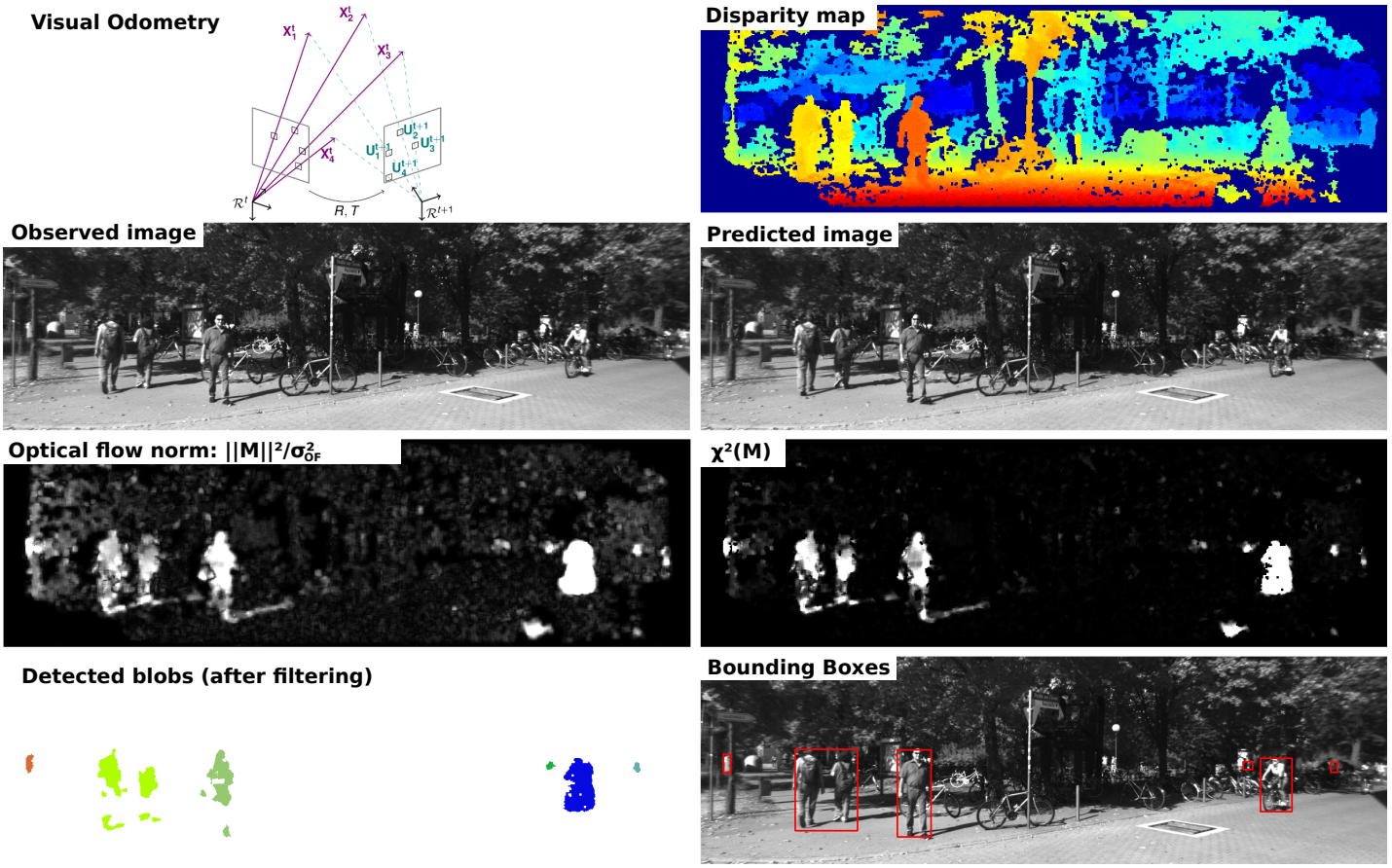

Fig. 3. Illustration of the different stages of the mobile object detection process.

close to the one presented in [21] are considered in [5] and [6]. However, these algorithms are not only computationally demanding, but also their robustness on real images is questionable. Indeed, TV-L1 approach requires expensive image pre-processing to deal with intensity changes in real world images, see for instance the computation of TV-L2 residual images discussed in [6].

One can conclude that the main point which precludes real-time operation of these methods is the use of such costly variational OF methods. Here we propose to use eFOLKI [12], a very fast OF algorithm based on Lucas-Kanade (LK) approach. Compared on the same GPU hardware, the runtime of eFOLKI is between one or two order of magnitude lower than variational methods such as TV-L1 [21] and Brox et al. [22]. Actually, looking at the OF benchmark of KITTI's website, eFOLKI appears among the very few methods able of real-time operation on 2 megapixels images. LK methods are generally considered as inaccurate in the computer vision community. However, Ref. [12] shows that it compares favourably with variational methods on the training dataset of KITTI, and that it provides useful solutions for various vision problems based on OF estimation. In the same line, we will show here that it leads to results of sufficient quality for our detection purpose.

2.3. *Choice of the residual field*

Here we present in more details several direct and image prediction methods, in order to introduce our original framework and compare various approaches in the experiments.

2.3.1. *Convention and Notations*

In the following, we consider the pinhole camera model and take the left camera as reference. We use these notations:

- I_t, left camera image at time instant t
- f, the camera focal in pixel (we suppose without loss of generality, that the horizontal and vertical focal are equal)
- b, the stereorig baseline
- (x, y), the discrete grid corresponding to the pixels' coordinates in image frame
- (x_0, y_0), the image coordinates of the projection of the optical center
- d_t, disparity map at time instant t
- (u, v), the optical flow between I_t and I_{t-1}
- U_t, the 2D coordinates in I_t

- X_t, the 3D coordinates in the camera frame at time instant t
- $X_t(x, y, d_t)$, the triangulation function at time instant t:

$$X_t(x, y, d_t) = -\frac{b}{d_t(x, y)} \begin{pmatrix} x - x_0 \\ y - y_0 \\ f \end{pmatrix} \tag{2}$$

- Π, the projection operator which maps a 3D point to the image plan
- R and T, the rotation and translation of the camera betweend $t-1$ and t such that $X_{t-1} = RX_t + T$

For a minimal lag in the detection process, we compute the detection in current image frame, by combining previous and current stereo images. For that purpose, for direct method, we need to match pixels from I_t to I_{t-1} (using optical flow) and we need to map 3D points from the current to the previous camera frame (using $X_{t-1} = RX_t + T$). Thus, we don't estimate the optical flow as usually done —forward in time—: $OF(I_{t-1} \to I_t)$. We estimate it backward in time: $OF(I_t \to I_{t-1})$, so that the estimation is done in the current pixel grid (x, y). As for the image prediction method, we estimate $OF(I_t \to I_t^{pred})$.

2.3.2. *Direct methods*

Direct methods have been applied either to Scene Flow, or to image quantities such as residual OF and disparity. These approaches differ essentially by the way they encode the depth information. We adopt the latter which eases the error modeling step. Recall that we proceed backward in time by considering the changes between time instants t and $t-1$ —and not between t and $t+1$ as usually done. Here the pixel grid (x, y) corresponds to I_t.
We assume that disparity maps d_{t-1} and d_t, and the optical flow (u, v) from I_t to I_{t-1} are available. Given the camera motion obtained from the visual odometry, the scene can be transferred into the coordinate frame at $t-1$:

$$X_{t-1}^{pred}(x, y, d_t) = RX_t(x, y, d_t) + T, \tag{3}$$

under a static scene hypothesis. Then the predicted disparity writes

$$d_{t-1}^{pred}(x, y) = \frac{-bf}{(0\ 0\ 1)\, X_{t-1}^{pred}(x, y, d_t)}, \tag{4}$$

and the predicted OF:

$$\begin{pmatrix} u_{pred} \\ v_{pred} \end{pmatrix}(x, y) = U_{t-1}^{pred}(x, y, d_t) - \begin{pmatrix} x \\ y \end{pmatrix}, \tag{5}$$

where

$$U_{t-1}^{pred}(x, y, d_t) = \Pi\left(X_{t-1}^{pred}(x, y, d_t)\right) \tag{6}$$

is the predicted image coordinates in previous frame.

The residual M is then:

$$M(x, y) = \begin{pmatrix} u(x, y) - u_{pred}(x, y) \\ v(x, y) - v_{pred}(x, y) \\ d_{t-1}(x + u, y + v) - d_{t-1}^{pred}(x, y) \end{pmatrix} \tag{7}$$

Note that instead of considering $M = \{\delta u, \delta v, \delta d\}$, some authors[8] use $M = \{\delta u, \delta v\}$ only.

2.3.3. *Image and disparity prediction method of [9]*

The predicted image in [10] and [9] is computed from previous grayscale image intensity and from the predicted 3D structure of Eq. (3):

$$I_t^{pred}\left(U_t^{pred}(x, y, d_{t-1})\right) = I_{t-1}(x, y) \tag{8}$$

where

$$\begin{cases} U_t^{pred}(x, y, d_{t-1}) = \Pi\left(X_t^{pred}(x, y, d_{t-1})\right) \\ X_t^{pred}(x, y, d_{t-1}) = R^{-1}\Big(X_{t-1}(x, y, d_{t-1}) - T\Big) \end{cases} \tag{9}$$

Note that in this case, the pixel grid (x, y) corresponds to previous frame I_{t-1}. In [10], image correlation techniques are used to check the consistency of the predicted image with respect to the observed one. In [9], the residual optical flow $(\delta u, \delta v)$ is computed between the observed image I_t and the synthetized one I_t^{pred}. As mentionned in [9], issues due to occlusion can also occur and need to be dealt with. Indeed two distinct points (x, y) and (x', y') may be mapped at the same pixel location $U_t^{pred}(x, y, d_{t-1})$. In this case, the disparity is used to keep only the closest point.

Note that pixel quantization, occlusions, etc., may lead to unallocated pixels in the predicted image: intensities taken from the observed image are used to fill these empty regions. Thanks to the robustness of OF codes, these problems affect the estimation only locally.
As well as they synthesize I_t^{pred}, the authors in [9] compute the predicted disparity map, which requires to map d_{t-1} in current camera frame:

$$d_t^{pred}\left(U_t^{pred}(x, y, d_{t-1})\right) = \frac{-bf}{(0\ 0\ 1)\, X_t^{pred}(x, y, d_{t-1})} \tag{10}$$

Finally, the residual OF $(\delta u, \delta v)$ is also used to compare d_t^{pred} with d_t:

$$\delta d = d_t(x, y) - d_t^{pred}(x + \delta u, y + \delta v) \tag{11}$$

and the resulting residual field is: $M = \{\delta u, \delta v, \delta d\}$

2.3.4. *Proposed method*

Our method is close to the one of Bak et al. [9], in the sense that we also compute a predicted image and then estimate a residual flow on it. However, unlike [9], we proceed backward by interpolating image intensities at $t-1$ from the reference frame coordinate at t.

We use Eq. (6) to compute U_{t-1}^{pred} then we synthesized $I_t^{pred}(x,y)$ by interpolating image intensity I_{t-1} at the positions $U_{t-1}^{pred}(x,y,d_t)$. We have to deal also with occlusions and we fill empty regions with current image intensity. Finally, we compute the residual OF $(\delta u, \delta v)$ from I_t to I_t^{pred}.

The main benefit of this approach is to simplify image interpolation. Indeed, in our formulation we need to interpolate irregular data from data located on a regular grid, while the approach of Bak et al. requires the opposite, ie. to interpolate regular data from irregularly arranged ones, which is more computationally demanding and may lead to local artifacts.

Furthermore, considering $U_{t-1}^{pred}(x,y,d_t)$ instead of $U_t^{pred}(x,y,d_{t-1})$ (as done in [9] and [10]), make the computation of $\xi^2(M) = M^T\Sigma_M^{-1}M$ easier since $M(x,y)$ and $\Sigma_{M(x,y)}$ are then expressed in current image pixel grid (x,y). On the contrary, when using $U_t^{pred}(x,y,d_{t-1})$, the resulting error covariance matrix $\Sigma_{M(x,y)}$ is expressed in previous image pixel grid, and further processing has to be done to associate Σ_M with M which is computed in current image pixel grid.

As previously done, we can add to the residual the difference between the observed and the predicted disparity given by Eq. 4:

$$\delta d = d_{t-1}\left(U_{t-1}^{pred}(x,y) + \begin{pmatrix}\delta u(x,y)\\ \delta v(x,y)\end{pmatrix}\right) - d_{t-1}^{pred}(x,y) \quad (12)$$

Then the residual writes $M = \{\delta u, \delta v, \delta d\}$.

However in our case we only consider $M = \{\delta u, \delta v\}$, as it leads to better results for the detection task as show in the experimental study (see Fig. 12).

2.4. *Detection of mobile objects*

Knowing the residual field and an estimation of its covariance, we can compute the Mahalanobis distance $\xi(M)$ of Eq. (1). As done in [8], we add a geometric constraint by only considering objects that are lower than $H_{max} = 2.5$m. Since we use KITTI datasets [13] in our experiments, we assume the camera horizontally oriented and positioned at $H_{cam} = 1.65$m from the floor. Under these assumptions, valid pixels satisfy:

$$H_{cam} + b\frac{y-y0}{d} < H_{max}. \quad (13)$$

For a fast segmentation of mobile objects, we a use the simple and computationally effective approach describe below. After applying a threshold to $\xi^2(M)$, we extract the connected components so as to form detected blobs. For each blob, we compute the median disparity that is used to calculate its depth attribute. To ease the following processes, blobs are considered as fronto-parallel planar regions. The surface area is measured for each blob given its depth attribute, and blobs of small size (e.g. below 0.01m^2) are suppressed. The remaining blobs are merged with one another if they are close enough (e.g. closer than 30cm) in 3D. When all neighboring blobs have been merged, small blobs aggregates are suppressed. To do so, we estimate the total surface of an aggregate by summing the surface associated to each pixel belonging the aggregate's blobs. We threshold this value (e.g. by 0.16m^2) and estimate bounding boxes for the remaining blobs aggregates, as well as their depth attributes. To prevent from the effect of a possible stereo failure for the furthest points, we only take into account blobs whose estimated depth is below a certain range Z_{max} (e.g. 40m).

Figure 4 shows an example of such estimated bounding boxes (in red) compared to ground truth bounding boxes (in blue), manually annotated using Vatic [23]. Let us recall that detections are made at each time independently.

Fig. 4. Bounding boxes estimated by the proposed framework (in red) compared to the BB annotated using Vatic (in blue).

3. Error Model

In this section, an error model for the residual field M, is studied. The objective is to model the error covariance matrix Σ_M. In the following, the analysis is carried on $M = \{\delta u, \delta v, \delta d\}$ and can be transposed to $M = \{\delta u, \delta v\}$ by ignoring the third dimension.

Whatever the method employed, the computation of M involves an estimation stage —e.g. estimating $\{u, v, d_{t-1}(x+u, y+v)\}$— and a prediction stage —e.g. predicting $\{u_{pred}, v_{pred}, d_{t-1}^{pred}\}$— that are subject to uncertainty. Assuming the estimation and the prediction error uncorrelated, we obtain:

$$\Sigma_M = \Sigma_{Estim} + \Sigma_{Pred} \quad (14)$$

Concerning the prediction stage, the methods described in 2.3.2 (direct approach) and 2.3.4 (image prediction approach) both rely on the computation of X_{t-1}^{pred} (3) to calculate U_{t-1}^{pred} (6) and d_{t-1}^{pred} (4). Since X_{t-1}^{pred} depends on X_t and $[R,T]$, its estimation can be perturbed by an error

occurring during the triangulation process but also during the ego-motion estimation. The impact of the triangulation error has been considered in many articles. However, to the authors knowledge, only Alcantarilla et al. [4] have proposed an ego-motion uncertainty model directly related to the visual odometry without considering parameters learned a priori.

Assuming that the triangulation and the odometry error are uncorrelated, we first study separately Σ_{X_t} and $\Sigma_{R,T}$, as well as their respective contribution to Σ_{Pred}. Then we describe Σ_{Estim} modeling, and finally come to the expression of Σ_M.

3.1. $X_t(x, y, d)$ *Estimation Error*

Each point observed in I_t and for which the disparity has been estimated by dense stereo, is triangulated according to (2). Because of pixel quantization, image coordinates (x, y) are prone to error. We model this by considering standard deviations σ_x and σ_y (e.g. equal to 0.2 pixel). We also represent the error of the disparity obtained with a dense algorithm, by σ_d (e.g. 1 pixel).

$\Sigma_{X_t(x,y,d)}$ is approximated by first order propagation of the error on (x, y, d):

$$\Sigma_{X_t(x,y,d)} = J_{X_t(x,y,d)} \begin{pmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_d^2 \end{pmatrix} J^T_{X_t(x,y,d)} \tag{15}$$

where $J_{X_t(x,y,d)}$ is the Jacobian of $X_t(x, y, d)$

3.2. (R, T) *Estimation Error*

To model $\Sigma_{R,T}$, we must know the energy minimized during the visual odometry (VO). In our case, we choose the same odometry as the one used in [15], i.e. we minimize in a RANSAC procedure[24] the reprojection error

$$\mathcal{E}(R, T) = \frac{1}{N} \sum_{k=1}^{N} \|U^k_{t-1} - \Pi(RX^k_t + T)\|^2 \tag{16}$$

where $\{X^k_t\}_k$ is a set of triangulated feature points that have been extracted in I_t, and $\{U^k_{t-1}\}_k$ their location in I_{t-1} obtained by temporal matching.

This energy is minimized over $\Theta = (\theta_x, \theta_y, \theta_z, T_x, T_y, T_z)$, with the three first parameter being Euler's angles of R.

3.2.1. *Pseudo-hessian based model*

A widespread approach to model the covariance matrix of parameters obtained by non-linear least square minimization, is to consider the inverse of the approximated Hessian matrix of criterion (16):

$$\Sigma_\Theta \propto \left(J^T_{f(\Theta)} J_{f(\Theta)} \right)^{-1} \tag{17}$$

with:

$$\begin{cases} f(\Theta)^T = \left(f_1(\Theta)^T, \cdots, f_N(\Theta)^T\right) \\ f_k(\Theta) = U^k_{t-1} - \Pi(RX^k_t + T) \end{cases} \tag{18}$$

As shown in [25], the proportionality factor in Eq. 17 can be fixed to σ_u^2 if the error on $\{U^k_{t-1}\}_k$ follows a gaussian law $\mathcal{N}(0, \sigma_u^2 Id_2)$. Thus by using this pseudo-hessian to approximate Σ_Θ, Alcantarilla et al.[4] only take into account the error related to the estimation of $\{U^k_t\}_k$ and its influence on the error on Θ. On the contrary, the comprehensive error model presented below includes the contribution of all the input data $\{U^k_{t-1}, X^k_t\}_k$.

3.2.2. *Comprehensive error model*

The relation between Θ and input data $\{z^k_t\}_k = \{U^k_{t-1}, X^k_t\}_k$ is implicit but can be recovered by applying the well-known Implicit Function Theorem (cf. [26], chap 5). Considering the implicit function $\varphi : (\Theta, z) \mapsto \frac{\partial \mathcal{E}}{\partial \Theta}(\Theta, z)^T$, we then obtain the error covariance matrix below:

$$\Sigma_\Theta = H^{-1} \left(\frac{\partial \varphi}{\partial z} \right) \Sigma_z \left(\frac{\partial \varphi}{\partial z} \right)^T H^{-T} \tag{19}$$

where $H = \frac{\partial^2 \mathcal{E}}{\partial \Theta \partial \Theta}(\Theta, z) \in \mathbf{R}^{6\times 6}$, is supposed invertible. Assuming the error independent for each feature k, Eq. (19) becomes:

$$\Sigma_\Theta = \sum_k H^{-1} \left(\frac{\partial \varphi}{\partial z_k} \right) \Sigma_{z_k} \left(\frac{\partial \varphi}{\partial z_k} \right)^T H^{-T} \tag{20}$$

As U^k_{t-1} and X^k_t are computed separately during the sparse temporal matching and the triangulation steps, we assume that they are not correlated, which leads to:

$$\Sigma_{z_k} = \left(\begin{array}{cc|c} \sigma_u^2 & 0 & \\ 0 & \sigma_v^2 & 0_{2\times 3} \\ \hline 0_{3\times 2} & & J_{X^k_t} \Sigma_{(x,y,d^*)} J_{X^k_t}{}^T \end{array} \right) \tag{21}$$

where the upper left diagonal matrix is the error model of the sparse temporal matching ($\sigma_u = \sigma_v = 0.5$ pixel), and d^* the disparity of the feature point whose error is modelled by σ_{d^*} — we choose $\sigma_{d^*} = 0.5$ pixel, i.e. a value lower than σ_d.

3.3. Σ_{Pred} *modelling*

X^{pred}_{t-1} is a function of triangulated point X_t and the ego-motion parameters Θ:

$$X^{pred}_{t-1}(X_t, \Theta) = R(\Theta)X_t + T(\Theta) \tag{22}$$

Thus we may approximated at the first order $\Sigma_{X^{pred}_{t-1}}$ by:

$$\Sigma_{X^{pred}_{t-1}} = J_{X^{pred}_{t-1}(X_t,\Theta)} \left(\begin{array}{c|c} \Sigma_{X_t} & 0_{3\times 6} \\ \hline 0_{6\times 3} & \Sigma_\Theta \end{array} \right) J^T_{X^{pred}_{t-1}(X_t,\Theta)} \tag{23}$$

Similarly, regarding (6) and (4) as functions of X_{t-1}^{pred} we can calculate:

$$\sigma^2_{d_{t-1}^{pred}} = J_{d_{t-1}^{pred}}(X_{t-1}^{pred}) \Sigma_{X_{t-1}^{pred}} J^T_{d_{t-1}^{pred}}(X_{t-1}^{pred}) \quad (24)$$

$$\Sigma_{U_{t-1}^{pred}} = J_{U_{t-1}^{pred}}(X_{t-1}^{pred}) \Sigma_{X_{t-1}^{pred}} J^T_{U_{t-1}^{pred}}(X_{t-1}^{pred}) \quad (25)$$

In the case of the direct method, the predicted OF (5) is compared to the estimated one. We consider that the OF estimation compensates the error due to pixel quantization, so the error covariance matrix of (u_{pred}, v_{pred}) is equal to $\Sigma_{U_{t-1}^{pred}}$.
For the proposed method, I_t^{pred} is obtained by interpolating I_{t-1} the location U_{t-1}^{pred}, so the prediction error is also modelled by $\Sigma_{U_{t-1}^{pred}}$

In both cases, the error related to the prediction is:

$$\Sigma_{Pred} = \left(\begin{array}{c|c} \Sigma_{U_{t-1}^{pred}} & 0_{2\times 1} \\ \hline 0_{1\times 2} & \sigma^2_{d_{t-1}^{pred}} \end{array}\right) \quad (26)$$

3.4. Σ_{Estim} and Σ_M modelling

As seen in 2.3.2 and 2.3.4, the disparity maps d_{t-1} and d_t must be estimated (d_t suffices if we only take the OF component M). The OF also have to be computed: either between I_t and I_{t-1} (noted (u, v) in the direct approach) or between I_t and I_t^{pred} (noted $(\delta u, \delta v)$ in the proposed approach). We take into account these sources of uncertainty via the error covariances σ_d^2 and $\Sigma_{OF} = \sigma^2_{OF} Id_2$, that we suppose constant (e.g. 1 pixel for σ_d and 0.5 pixel for σ_{OF}).

For the third dimension of M, the predicted disparity $d_{t-1}^{pred}(x, y)$ needs to be compared to $d_{t-1}(w(x, y))$, with $w(x, y) = (x + u(x, y), y + v(x, y))$ (direct approach) or $w(x, y) = U_{t-1}^{pred}(x, y) + \begin{pmatrix} \delta u(x, y) \\ \delta v(x, y) \end{pmatrix}$ (proposed approach). In the following we ignore the uncertainty on $w(x, y)$, and model the error on $d_{t-1}(w(x, y))$ via σ_d^2 only.

Finally, the estimation error covariance is modelled for both methods as:

$$\Sigma_{Estim} = \begin{pmatrix} \sigma^2_{OF} & 0 & 0 \\ 0 & \sigma^2_{OF} & 0 \\ 0 & 0 & \sigma_d^2 \end{pmatrix} \quad (27)$$

And then $\Sigma_M(x, y)$ is given by (14).

4. Experimental Results

In this section, experimental results obtained on two stereo sequences (09/28-0037 and 09/29-0071) of the publicly available KITTI datasets [13], are presented. These results are evaluated qualitatively by displaying residual images $\xi^2(M)$, but also quantitatively via *Precision/Recall* curves resulting from the evaluation protocol explained below.

4.1. *Evaluation Protocol*

To evaluate the various tested approaches, we have manually annotated ground truth bounding boxes BB_{GT} using Vatic [23]. To associate a score to the detected bounding boxes BB, we choose a discrete criterion based on the overlap ratio:

$$\omega(BB, BB_{GT}) = \frac{\mathcal{A}_{BB \cap BB_{GT}}}{\mathcal{A}_{BB \cup BB_{GT}}}. \quad (28)$$

An detected BB is valid when there exists a BB_{GT} such that $\omega(BB, BB_{GT})$ is below some threshold (e.g. 20%). To avoid multiple instances of the same detection, we count one True Positive for each BB_{GT} whatever the number of valid BB it is associated to. Other estimated bounding boxes are considered as False Positive, and ground truth bounding boxes with no associated True Positive detection are False Negative. Several evaluations were done using different thresholds on $\xi^2(M)$ (e.g. from 1 to 30) to construct *Precision/Recall* curves like Fig. 6. Note that because the evaluation is based on the overlap ratio $\omega(BB, BB_{GT})$, decreasing the threshold doesn't always increase the number of True Positive. Thus a *Precision/Recall* curve obtained that way, is not necessarily a monotonously decreasing function of the threshold (e.g. in Fig. 12, the blue curves).

4.2. *Comparison of Stereo Algorithms*

For the reasons exposed in 2.2, we have considered the following dense stereo algorithms: SGM, ELAS and BM. More precisely, we have used the OpenCV[a] version of SGM: Semi-Global Block Matching (SGBM) which mainly differs from the original version by its matching cost. We set the boolean parameter *fullDP* to true, in order to compute the matching cost along 8 directions as done in SGM. Concerning ELAS setup, we activate the subsampling option to speed-up the computation and enable real-time processing. The disparity maps computed by these three algorithms, are shown in Fig.5. Regarding the image coverage and the influence on the detection performances (cf. Fig. 6), ELAS appears to be a good choice for our application. Especially since it can perform in real-time on a single CPU, with the chosen settings (45ms on a CPU Intel Core i7, for KITTI images). In the following, we choose ELAS to compute the disparity maps.

[a] Available from http://sourceforge.net/projects/opencvlibrary/.

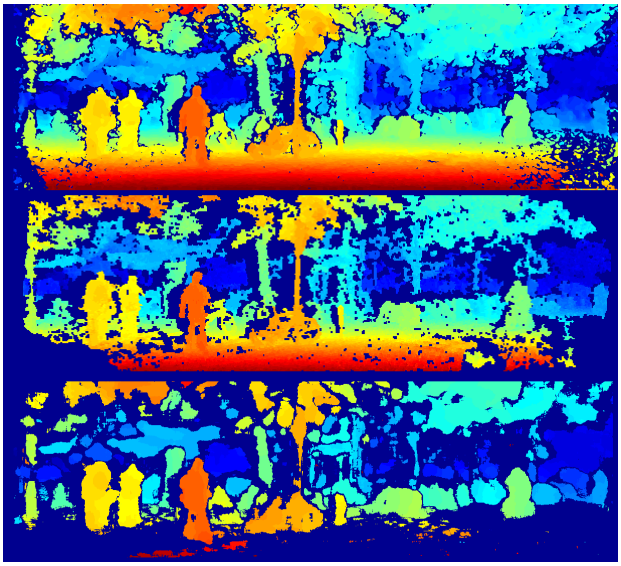

Fig. 5. Examples of disparity maps computed by using SGBM (top), ELAS (middle) and BM (bottom).

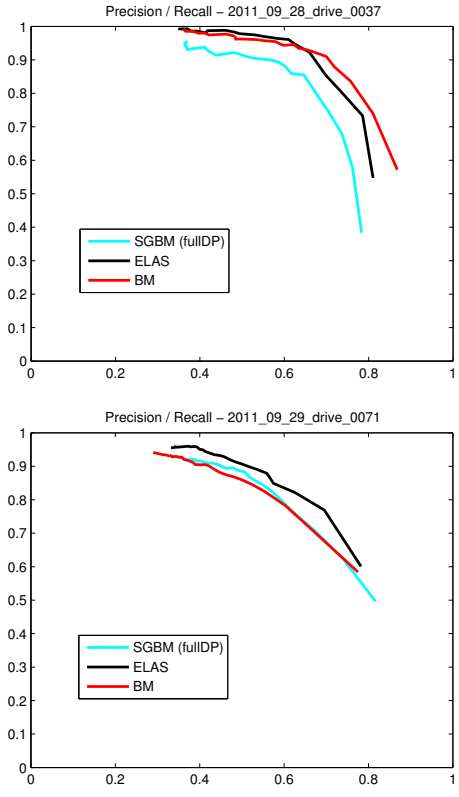

Fig. 6. *Precision/Recall* curves of sequences 09/28-0037(top) and 09/29-0071(bottom) considering $\xi^2(\delta u, \delta v)$ obtained with the proposed method and the comprehensive error model, using SGBM (blue), ELAS (black) and BM (red).

4.3. *Impact of the Optical Flow Algorithm*

We have also compared eFOLKI with the variational optical flow of Brox et al. [22] which is more accurate than TV-L1 on KITTI dataset, as shown in [12]. Parameters of eFOLKI are $J = 5$ resolution levels, $K = 5$ iterations, two window radii {8;4} and rank order 4. Opencv default parameters are used for Brox et al. OF i.e. $\alpha = 0.197$, $\gamma = 50$, 10/10/77 solver/inner/outer iterations and a pyramid scale of 0.8. Given these settings, the residual field obtained with Brox et al. OF is smoother than the one computed by eFOLKI (cf. Fig. 7 and Fig. 8), but doesn't improve the detection. Actually Brox et al. OF not only performs slower than eFOLKI (cf. Table 1) but also often leads to poorer results in term of detection as shown in Fig. 9. Brox et al appears to more precise for the detection only in sequence 09/29-0071 when considering a threshold of 1 or 2 (which corresponds to the right-most points of the Precision/Recall curves).

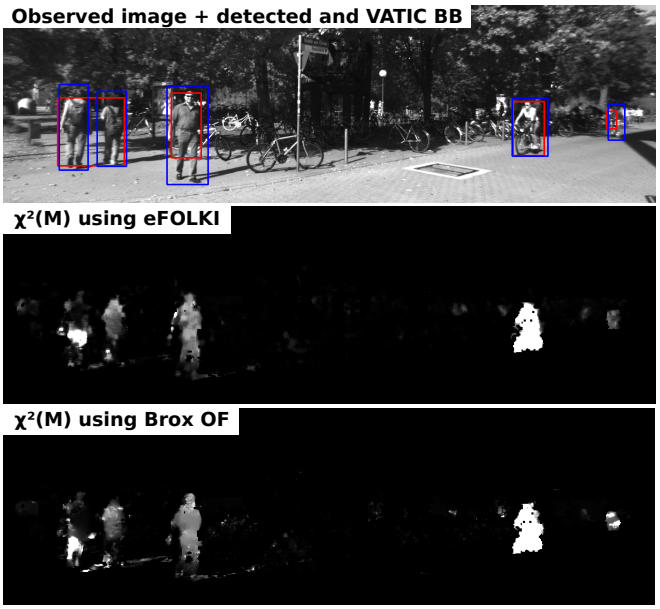

Fig. 7. Left image (top) with ground truth (blue) and estimated BBs (red); $\xi^2(\delta u, \delta v)$ resulting from the proposed method and eFOLKI [12] (middle) and Brox et al. OF [22] (bottom).

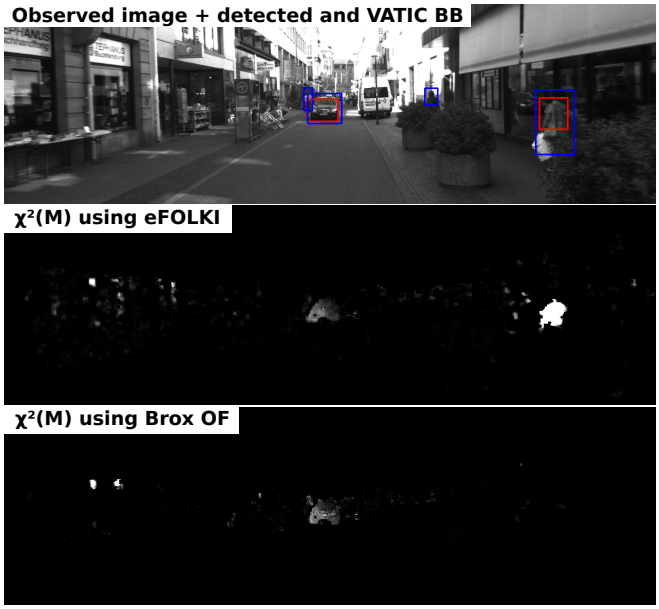

Fig. 8. Left image (top) with ground truth (blue) and estimated BBs (red); $\xi^2(\delta u, \delta v)$ resulting from the proposed method and eFOLKI [12] (middle) and Brox et al. OF [22] (bottom).

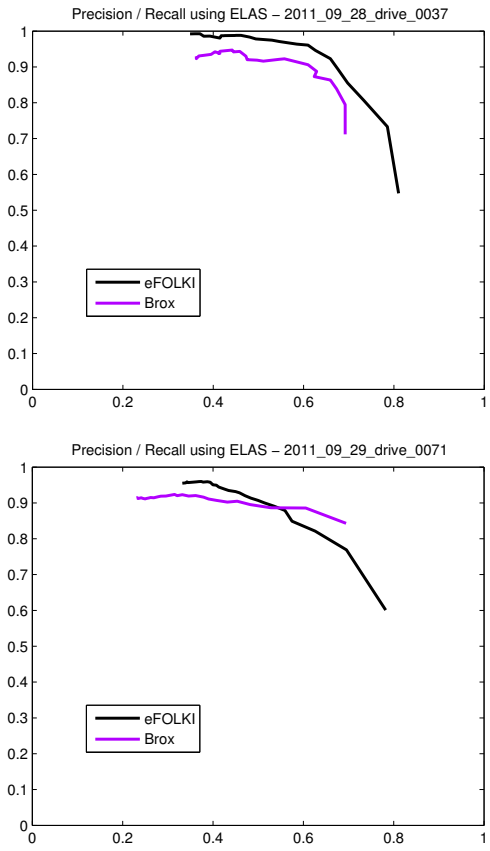

Fig. 9. *Precision/Recall* curves of sequences 09/28-0037 (top) and 09/28-0037 (bottom) considering $\xi^2(\delta u, \delta v)$ obtained with the proposed method and the comprehensive error model, using eFOLKI (black) and Brox et al. OF [22] (purple).

4.4. *Residual Fields Comparison*

Experiments have shown that the motion likelihood $\xi^2(\delta u, \delta v, \delta d)$ is more noisy than $\xi^2(\delta u, \delta v)$, whatever the chosen approach (cf. Fig. 10 and Fig. 11). Although the additional consideration of δd improve sometimes the detection (e.g. in Fig. 11 the pedestrians in the center of the image, that are walking away from the camera), the *Precision/Recall* curves in Fig. 12 suggest that it is not worth it.

Whether or not we use δd, the proposed image prediction methods appears to be the less noisy than the direct ones, as shown in Fig. 10 and Fig. 11. This is confirmed by the *Precision/Recall* curves in Fig. 12 where we notice a huge gap between the curves of both methods.

This superiority of image prediction methods over the direct ones may be explained by the fact that it is more suitable to compute the OF between I_t and I_t^{pred} than between I_t and I_{t-1}. Indeed, algorithms that belong to local or window-based approaches of optical flow —like eFOLKI— estimate $\mathbf{u}(\mathbf{x})$ as the minimizer of a criterion computed over a local window $W(\mathbf{x})$ centered on pixel $\mathbf{x}$:

$$\sum_{\mathbf{x}' \in W(\mathbf{x})} \alpha(\mathbf{x}' - \mathbf{x}) \| I_1(\mathbf{x}') - I_0\left(\mathbf{x}' + \mathbf{u}(\mathbf{x})\right) \|^2 \qquad (29)$$

This 2D regularization corresponds to the assumption that the optical flow constant over a local window. This assumption is unrealistic in the case of OF(I_t,I_{t-1}), but suits to OF(I_t,I_t^{pred}) since $\mathbf{u}$ is supposedly constant (equal to zero) in the static parts of the image. In other words, OF(I_t,I_t^{pred}) acts like a low-pass filter in the static parts of the image.

Finally, as shown in Fig. 13, the noise in the motion likelihood $\xi^2(M)$ can be further reduced by a choosing a finer uncertainty model which takes into account the ego-motion uncertainty Σ_Θ, though at the cost of a lower SNR on moving objects. Note that the gain resulting from the comprehensive odometry error model Σ_Θ is more important in sequence 09/28-0037 during which the camera encounters a large rotation whose uncertainty introduces a lot of noise in the estimated residual field M. The camera movement in sequence 09/29-0071 is relatively slow and purely translational, and in that case experience has shown that Σ_Θ has less impact on the residual field M.

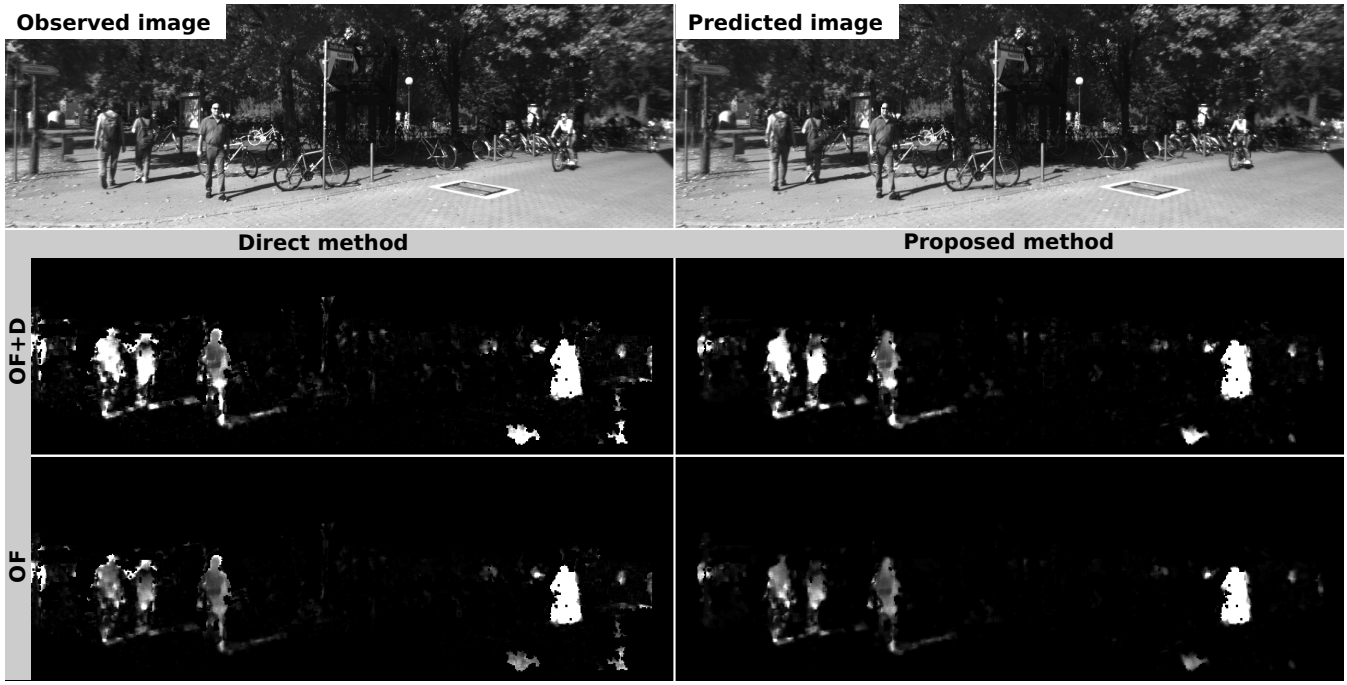

Fig. 10. Several $\xi^2(M)$ images from sequence 09/28-0037, for $M = \{\delta u, \delta v, \delta d\}$ (second row) and $M = \{\delta u, \delta v\}$ (third row) by applying direct approach (left column) and the proposed one (right column). First row shows the observed and the predicted image.

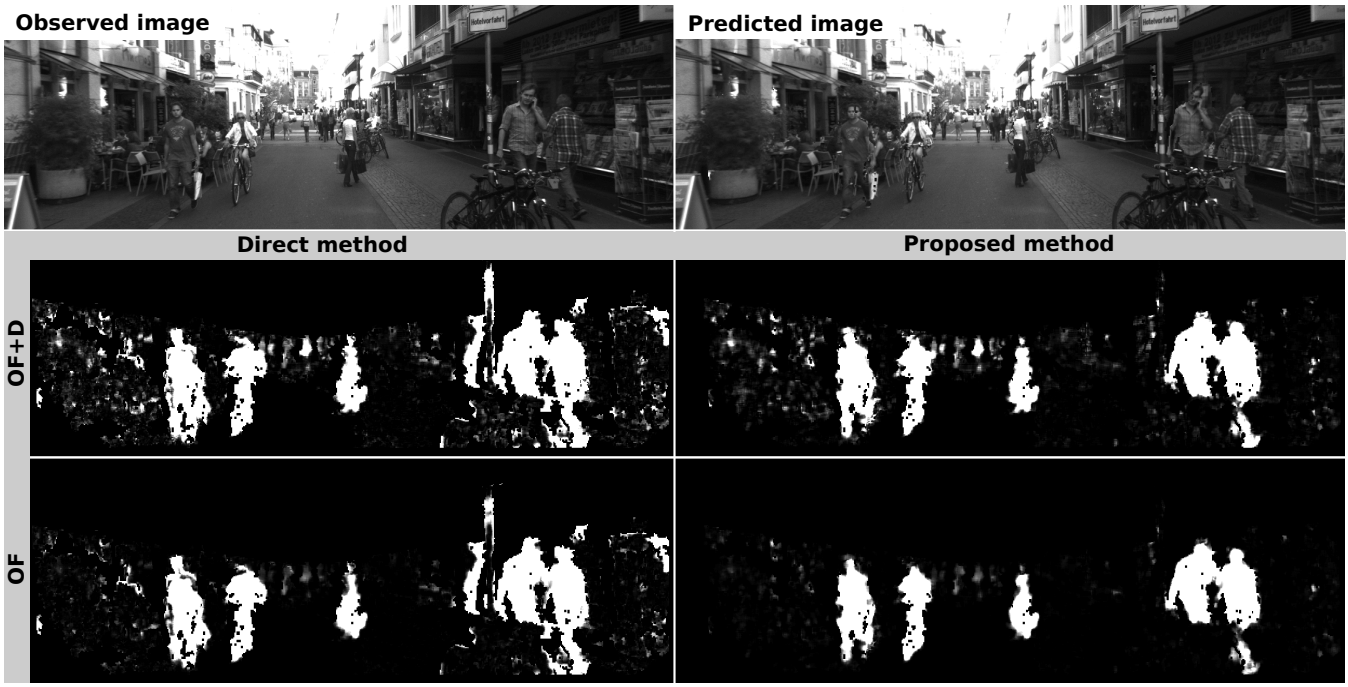

Fig. 11. Several $\xi^2(M)$ images from sequence 09/29-0071, for $M = \{\delta u, \delta v, \delta d\}$ (second row) and $M = \{\delta u, \delta v\}$ (third row) by applying direct approach (left column) and the proposed one (right column). First row shows the observed and the predicted image.

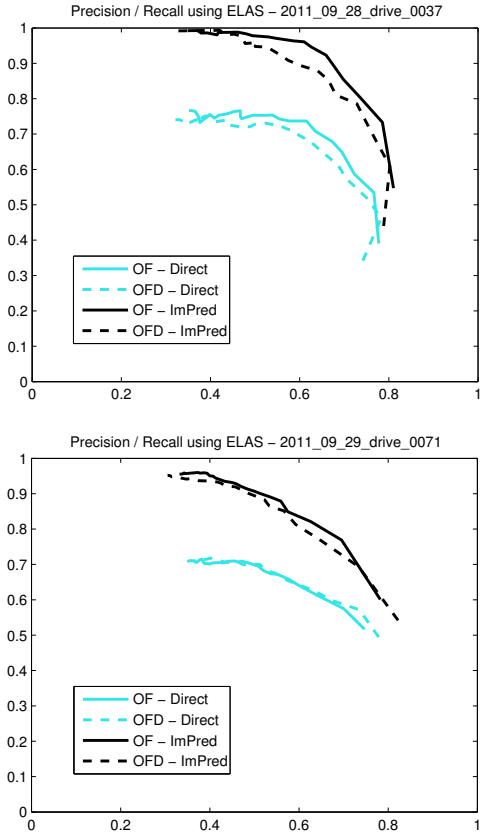

Fig. 12. *Precision/Recall* curves of sequences 09/28-0037 (top) and 09/29-0071 (bottom) considering $\xi^2(\delta u, \delta v)$ (plain line) or $\xi^2(\delta u, \delta v, \delta d)$ (dashed line) obtained with the direct (blue) and the proposed method (black), with the comprehensive error model.

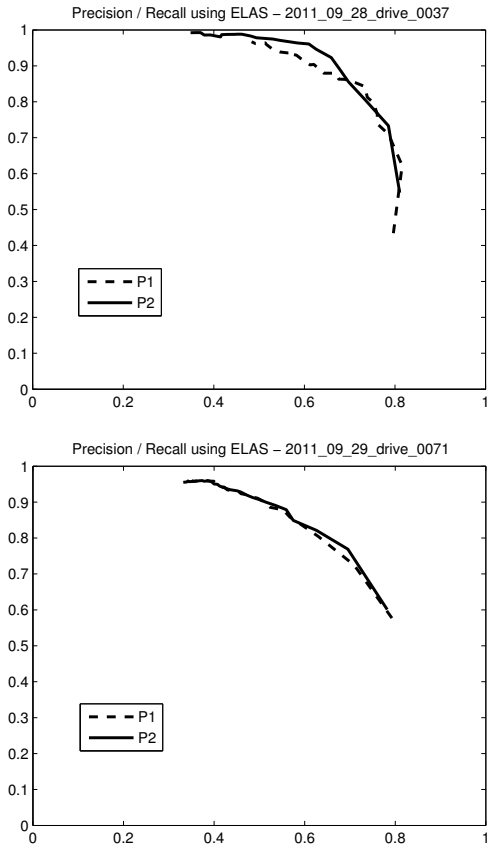

Fig. 13. *Precision/Recall* curves of sequences 09/28-0037 (top) and 09/28-0037 (bottom) considering $\xi^2(\delta u, \delta v)$ obtained with the proposed method and the comprehensive error model(plain line) or when neglecting pose uncertainty Σ_Θ (dashed line).

4.5. *Processing Time*

Table 1. Processing Time For Each Stage

VO & ELAS	Prediction	OF: eFOLKI/Brox	$\xi^2(M)$	CC + BB
45ms	2ms	27ms/129ms	7ms	5-10ms

Table 1 summarizes processing times with a CPU Intel Core i7 and a GPU GeForce GTX TITAN. For the disparity map estimation, we use ELAS which can run in parallel with the visual odometry in 45ms. Thus the whole pipeline performs in less than 100ms for KITTI images of size 370x1224. The use of the fast algorithm eFOLKI saves a considerable amount of time and enables the whole process to achieve video framerate (i.e. 10Hz). Note that multithreading the VO and ELAS would significantly decrease their runtimes on multicore systems.

Note that each stage required to achieve the computation of $\xi^2(M)$ is done in a controlled amount of time — either by deterministic operations or in a random loop with limited number of iterations. The time required for the last stage (connected component extraction and merging process) may vary according to the number of detected blobs in the binary image obtained by applying a threshold on $\xi^2(M)$. In our experiments, this stage has been perform in few milliseconds (typically 5ms).
From now, the bottleneck appears to be the visual odometry and the dense stereo. One could also use geometrical informations returned by the system to speed-up stereo — e.g. the disparity range may be deduced from previous disparity map and camera pose.

5. Conclusion

We have presented a framework for mobile object detection from a moving stereo rig based on an image prediction strategy. It is compatible with real-time processing thanks to a fast dense OF estimation. A new comprehensive error model has been derived, which allows to handle rigorously the uncertainty related to visual odometry. We have conducted an experimental study on several real videos from the KITTI website to compare our approach with various proposals of the literature. This study first show that the image prediction strategy improves the SNR of the motion likelihood. It also show that the fast OF algorithm eFOLKI is compatible with good detection rates.

We now plan to integrate our framework to an online ego-localization and environment mapping such as [3] and add temporal filtering by modelling the dynamics of the detected mobile objects.

Acknowledgments

This work was sponsored by the Direction Générale de l'Armement (DGA) of the French Ministry of Defense.

References

[1] H. Badino and T. Kanade, "A headwearable short-baseline stereo system for the simultaneous estimation of structure and motion", in IAPR *Conference on Machine Vision Application* (Nara, Japan, 2011), pp. 185–189.

[2] P. Lenz, J. Ziegler, A. Geiger, and M. Roser, "Sparse scene flow segmentation for moving object detection in urban environments", in *IEEE Intelligent Vehicles Symposium (IV)* (Baden-Baden, Germany, 2011), pp. 926–932.

[3] M. Sanfourche, A. Plyer, A. Bernard-Brunel, G. Le Besnerais "3DSCAN: Online Ego-Localization and Environment Mapping for Micro Aerial Vehicles", *AerospaceLab*, vol. 8, no. 2, pp. 1–17, 2014.

[4] P. Alcantarilla, J. Yebes, J. Almazan, and L. Bergasa, "On combining visual SLAM and dense scene flow to increase the robustness of localization and mapping in dynamic environments", in *IEEE International Conference on Robotics and Automation (ICRA)* (Saint Paul, MN, USA, 2012), pp. 1290–1297.

[5] C. Rabe, T. Mller, A. Wedel, and U. Franke, "Dense, robust, and accurate motion field estimation from stereo image sequences in realtime", in *European Conference on Computer Vision (ECCV)* (Crete, Greece, 2010), pp. 582–595.

[6] A. Wedel, T. Brox, T. Vaudrey, C. Rabe, U. Franke, and D. Cremers, "Stereoscopic scene flow computation for 3D motion understanding", in *International Journal of Computer Vision*, vol. 95, no. 1, pp. 29–51, 2011.

[7] A. Talukder and L. Matthies, "Realtime detection of moving objects from moving vehicles using dense stereo and optical flow", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Sendai, Japan, 2004), pp. 3718–3725.

[8] V. RomeroCano and J. I. Nieto, "Stereobased motion detection and tracking from a moving platform", in *IEEE Intelligent Vehicles Symposium (IV)* (Gold Coast City, Australia, 2013), pp. 499–504.

[9] A. Bak, S. Bouchafa, and D. Aubert, "Dynamic objects detection through visual odometry and stereovision: a study of inaccuracy and improvement sources", in *Machine vision and applications*, vol. 25, no. 3, pp. 681–697, 2014.

[10] M. Agrawal, K. Konolige, and L. Iocchi, "Realtime detection of independent motion using stereo", in *Seventh IEEE Workshops on Application of Computer Vision WACV/MOTIONS* (Breckenridge, CO, USA, 2005), pp. 207–214.

[11] A. Bak, S. Bouchafa, and D. Aubert, "Detection of independently moving objects through stereo vision and egomotion extraction", in *IEEE Intelligent Vehicles Symposium (IV)* (San Diego, CA, USA, 2010), pp. 863–870.

[12] A. Plyer, G. Le Besnerais, and F. Champagnat, "Massively parallel lucas kanade optical flow for real-time video processing applications", in *Journal of Real-Time Image Processing*, pp. 1–18, April 2014.

[13] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite", in *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on* (Providence, RI, USA, 2012), pp. 3354–3361.

[14] M. Derome, A. Plyer, M. Sanfourche and G. Le Besnerais, "Real-Time Mobile Object Detection Using Stereo", in *IEEE International Conference on Control Automation Robotics and Vision (ICARCV)* (Marina Bay Sands, Singapore, 2014), pp. 1021–1026.

[15] M. Sanfourche, V. Vittori, and G. Le Besnerais, "eVO: a realtime embedded stereo odometry for MAV applications", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Tokyo, Japan,

2013), pp. 2107–2114.
[16] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 2, pp. 328–341, 2008.
[17] S. K. Gehrig, F. Eberli, and T. Meyer, "A Real-Time Low-Power Stereo Vision Engine Using Semi-Global Matching", in *International Conference on Computer Vision Systems (ICVS)* (Liege, Belgium, 2009), pp. 134-143.
[18] A. Geiger, M. Roser, and R. Urtasun. "Efficient large-scale stereo matching", in *Computer Vision – ACCV* (Queenstown, New Zealand, 2010), pp. 25–38.
[19] B. K. Horn and B. G. Schunck, "Determining optical flow", in *Artificial Intelligence*, vol. 17, pp. 185-203, 1981.
[20] A. Bruhn, J. Weickert, and C. Schnrr, "Lucas/Kanade meets Horn/Schunck: combining local and global optic flow methods", *International Journal of Computer Vision*, vol. 61, no. 3, pp. 211–231, 2005.
[21] C. Zach, T. Pock, and H. Bischof, "A duality based approach for realtime tvl1 optical flow", in *Symposium of Pattern Recognition (DAGM)* (Heidelberg, Germany, 2007), pp. 214–223.
[22] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping", in *European Conference on Computer Vision (ECCV)* (Prague, Czech Republic, 2004), pp. 25–36.
[23] C. Vondrick, D. Patterson, and D. Ramanan, "Efficiently scaling up crowdsourced video annotation", in *International Journal of Computer Vision*, vol. 101, no. 1, pp. 184–204, 2013.
[24] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography", in *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
[25] Y. Bard, *Nonlinear Parameter Estimation.*Academic Press, 1974.
[26] O. Faugeras, *Three dimensional computer vision: A geometric viewpoint.* the MIT Press, 1993.

Maxime Derome graduated from Telecom ParisTech and received his Master degree in Mathematics, Vision and Learning (MVA) from ENS Cachan in 2013. He is a Ph.D student in second year from Universite Paris-Saclay, working in the Information Processing and Modeling Departement of the French Aerospace Lab (ONERA). He is holder of a scholarship from the Direction General de l'Armement (DGA) of the French Ministry of Defense.

Aurelien Plyer graduated from Universite Pierre et Marie Curie (Paris 6) in 2008 and received the Ph.D degree in Image Processing from the Universite de Paris 13, in 2013. His research deals with low level video processing and 3D environement perception for robotics, he uses GPU programming in order to implement real-time processing.

Martial Sanfourche graduated from Universite de Cergy-Pontoise in computer Sciences (2001) then received the Ph.D. degree in image and signal processing from the Universite de Cergy-Pontoise in 2005. After a postdoctoral position at CNRS- LAAS, he joined ONERA/DTIM in 2007 where is now a research engineer in computer vision. His current research interest include online and offline visual localization and mapping for robotic systems.

Guy Le Besnerais graduated from the Ecole Nationale Superieure de Techniques Avancees in 1989 and received the Ph.D. degree in physics from the Universite de Paris-Sud, Orsay, France, in 1993. He joined the ONERA in 1994, where he is now a senior scientist in the Information Processing and Modeling Department. His work concerns inversion problems in imagery and computer vision.