Ego-Motion and Indirect Road Geometry Estimation Using Night Vision

Thomas B. Schön
Division of Automatic Control
Linköping University
SE-581 83 Linköping, Sweden
Email: schon@isy.liu.se

Jacob Roll
Autoliv Electronics AB
SE-583 30 Linköping, Sweden
Email: jacob.roll@autoliv.com

***Abstract*—The sensors present in modern premium cars deliver a wealth of information. We will in this work illustrate one way of making better use of the sensor information already present in modern premium cars. More specifically, we will show how a far infrared (FIR) camera can be used to enhance the estimates of the vehicle ego-motion and indirectly the road geometry in 3D. The FIR camera is primarily intended for pedestrian detection. The solution is obtained by solving a suitable sensor fusion problem, where we merge information from proprioceptive sensors with the FIR camera images. In order to illustrate the performance of the proposed method we have made use of measurement sequences recorded during night-time driving on rural roads in Sweden. The results illustrate that the FIR images can be used to improve the ego-motion estimation, especially during night time driving.**

I. Introduction

New sensors are often introduced in cars with a specific application in mind. However, using sensor fusion, the information from these sensors can typically be used for other purposes as well. Recently, the use of far infrared (FIR) cameras for pedestrian detection has gained significant interest. Systems of this kind are already present in some modern premium cars. See e.g., [13] for an overview of such a system. The FIR camera detects thermal radiation and turns it into an image, where a bright object is warmer than a dark object. This enables the system to operate during night, where normal cameras cannot be used. In Fig. 1 we give a typical example of the images received from the FIR camera used in this work. We will study how to make use of this FIR camera within a sensor fusion framework in order to improve the ego-motion estimate.

Sensor fusion is defined as the process of using information from *several different sensors* to compute an *estimate* of the state of a *dynamical system*. The main contribution of this work is a method for estimating the ego-motion of the vehicle and hence indirectly the road geometry in 3D. Besides the FIR camera we will make use of several proprioceptive sensors. More specifically, we make use of the longitudinal velocity (from the wheel speed sensors) and the yaw rate measurements.

The idea of estimating the road curvature by extracting the line markings from the camera images [7, 8] cannot be used since the lane markings have the same temperature as the rest of the road. However, as can be seen from Fig. 1 there is a large temperature difference between the road and the soil next to the road. This might be possible to use at least for rural roads. However, the fact that the FIR camera is mounted rather close to the ground makes the problem hard.

(a) Road scene, as seen with a standard camera.

(b) Same road scene as above, seen with the FIR camera.

Fig. 1. The images above shows a typical road scene at night time. The top image is acquired using a standard camera, whereas the bottom image is acquired, at the same time, using the FIR camera used in this work.

Both the measurements from the camera and the proprioceptive sensors contain errors, for example due to discretization and wrong landmark data association. The good thing is that the errors associated to the camera are not correlated with the errors of the proprioceptive sensors.

In order to derive our solution we have been inspired by the work conducted within the areas of visual odometry [4, 18] and simultaneous localization and mapping (SLAM) [2, 6, 9, 19]. The state vector is chosen as

$$x_t = \begin{pmatrix} x_t^{\mathrm{v}} \\ x_t^{\mathrm{l}} \end{pmatrix} \tag{1}$$

where x_t^{v} denotes the state describing the vehicle and x_t^{l} denotes the current landmarks. The main difference to the SLAM problem is that we are not concerned with the so called

loop closing problem. In fact, we remove the landmarks from the state vector as soon as the vehicle has passed them.

II. Dynamic Model

In order to properly derive the dynamic model used in this work we first introduce the relevant coordinate frames,

- **World** (w): This is considered an inertial frame. The position and orientation of the vehicle are resolved in this frame.
- **Body** (b): The body frame is attached to the vehicle. More specifically, it is positioned in the middle of the rear axis.
- **Camera** (c): This frame is positioned in the optical center of the camera. Hence, the body frame and the camera frame are rigidly connected.

For an illustration of the relationships between the different coordinate frames we refer to Fig. 2. The vehicle is modelled

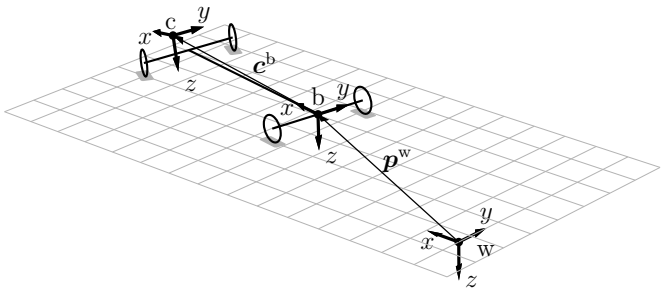

Fig. 2. Illustration of the coordinate frames used in this work. Note that we model the vehicle as if it only had two wheels. However, we have used four wheels in the illustration in order to properly show where the coordinate frames are positioned.

using a bicycle model, implying that it is modelled as if it only had two wheels. The state vector x_t^{v} is given by

$$x_t^{\mathrm{v}} = \begin{pmatrix} (\boldsymbol{p}^w)^T & v_x^w & \psi^w & \delta_{\mathrm{f}} & \alpha & \varphi \end{pmatrix}^T \tag{2}$$

where $\boldsymbol{p}^w$ is the vehicle body position in world coordinates, v_x^w is the longitudinal velocity of the vehicle, ψ_w is the yaw angle, δ_{f} is the front wheel angle, i.e., the angle between the longitudinal direction of the front wheel and the longitudinal axis of the host vehicle. Furthermore, α is the vertical angle between the road and the world coordinate frame (the pitch of the road) and φ is the pitch angle for the vehicle. Furthermore, we model the longitudinal acceleration as an input signal $u_t = a_{x,t}$. Using the geometry introduced in Fig. 2 we have the following expressions for the position and velocity dynamics of the vehicle,

$$\begin{pmatrix} p_{x,t+1} \\ p_{y,t+1} \\ p_{z,t+1} \\ v_{x,t+1} \end{pmatrix} = \begin{pmatrix} p_{x,t} + Tv_{x,t}\cos\psi_t\cos\alpha_t \\ p_{y,t} + Tv_{x,t}\sin\psi_t\cos\alpha_t \\ p_{z,t} - Tv_{x,t}\sin\alpha_t \\ v_{x,t} + Tu_t \end{pmatrix} + B(\psi_t,\alpha_t)w_t^{\mathrm{p}} \tag{3a}$$

$$B(\psi_t,\alpha_t) = \begin{pmatrix} T\cos\psi_t\cos\alpha_t \\ T\sin\psi_t\cos\alpha_t \\ T\sin\alpha_t \\ 1 \end{pmatrix} \tag{3b}$$

where T denotes the sampling time and w_t^{p} denotes the process noise, which is assumed to be independent and Gaussian according to

$$w_t^{\mathrm{p}} \sim \mathcal{N}(0,\, Q_t^{\mathrm{p}}). \tag{4}$$

When it comes to the vehicle orientation we only model the yaw angle and the pitch angle. The yaw and pitch dynamics are modelled according to

$$\psi_{t+1} = \psi_t + \frac{T}{l}v_{x,t}\tan\delta_{\mathrm{f},t} + w_t^{\psi}, \quad w_t^{\psi} \sim \mathcal{N}(0, Q_t^{\psi}), \tag{5a}$$

$$\varphi_{t+1} = C\varphi_t + w_t^{\varphi}, \quad w_t^{\varphi} \sim \mathcal{N}(0, Q_t^{\varphi}), \tag{5b}$$

where l denotes the wheel base and C denotes the damping term for the pitch dynamics.

Finally, the front wheel angle $\delta_{\mathrm{f},t}$ and the angle α_t between the road and the xy plane of the world coordinate frame are modelled as random walks according to

$$\delta_{\mathrm{f},t+1} = \delta_{\mathrm{f},t} + w_t^{\delta}, \quad w_t^{\delta} \sim \mathcal{N}(0, Q_t^{\delta}), \tag{6a}$$

$$\alpha_{t+1} = \alpha_t + w_t^{\alpha}, \quad w_t^{\alpha} \sim \mathcal{N}(0, Q_t^{\alpha}). \tag{6b}$$

The dynamic model introduced above is obviously a very simplified model of a vehicle, but given the sensor data that we have access to, it is hard to make use of more complicated models. However, if more sensors were available we could potentially obtain better results using more complicated vehicle models. For example, if we had access to a direct measurement of the front wheel angle, we could also include the float (body side slip) angle in order to model the direction of the velocity vector, see e.g., [15] for an illustration of how such a model can be used to improve the estimates. Furthermore, the pitch dynamics can be improved if measurements of the positions of the front and the rear suspension were available, see e.g., [16].

III. Sensors and Measurement Models

Proprioceptive sensors measure quantities that are internal to the vehicle, whereas the exteroceptive sensors provide information about the vehicle environment. The measurement equations for the proprioceptive sensors (longitudinal velocity and yaw rate) are introduced in Section III-A. Finally, the measurement model for the exteroceptive sensor (FIR camera) is given in Section III-B.

A. Proprioceptive Measurement Models

The longitudinal velocity $v_{x,t}^w$ and the yaw rate $\dot{\psi}$ are modelled as measurements,

$$y_t^{\mathrm{v}} = \begin{pmatrix} v_{x,t}^{\mathrm{m}} \\ \dot{\psi}_t^{\mathrm{m}} \end{pmatrix}, \tag{7}$$

where we have used superscript v to denote the fact that these measurements only depend on the vehicle state x_t^{v}. Furthermore, superscript m is used to denote that we refer to the actual measurements and not the states. The corresponding measurement equation is

$$y_t^{\mathrm{v}} = \begin{pmatrix} v_{x,t} \\ \frac{T}{l}v_{x,t}\tan\delta_{\mathrm{f},t} \end{pmatrix} + e_t^{\mathrm{v}}, \quad e_t^{\mathrm{v}} \sim \mathcal{N}(0, R_t^{\mathrm{v}}) \tag{8}$$

B. Far Infrared (FIR) Camera

The far infrared camera detects thermal radiation and turns it into an image, where a bright object is warmer than a dark object. An example of the type of image generated by the present FIR camera is given in Fig. 1. For more details on the FIR camera used in this work and its use for pedestrian detection, see [13]. Here it is also worth noting that mathematically we can treat the FIR camera as a standard camera. For details on mathematical camera models we refer to [17], which contains a solid introduction to camera geometry and calibration. On the other hand, an FIR image contains less details compared to a standard camera (used in day light), which can make it harder to find stable interest points. Before an image position is used as a measurement in an estimator, the position is adjusted according to the camera specific parameters, such as focal length, pixel sizes etc. This allows us to model the FIR camera as a device producing a normalized pinhole projection $\mathcal{P}_n$ according to

$$y^{\mathrm{l}}_{j,t} = \mathcal{P}_n\left(L^c_j\right) = \frac{1}{L^c_{j,x}}\begin{pmatrix} L^c_{j,y} \\ L^c_{j,z} \end{pmatrix} + e^{\mathrm{l}}_t, \quad e^{\mathrm{l}}_t \sim \mathcal{N}(0, R^{\mathrm{l}}_t), \quad (9)$$

where we have used L^c_j to denote the position of the j^{th} landmark expressed in the camera coordinate frame. Furthermore, it is worth noting that the x-axis is used as the optical axis. The exact details regarding the FIR camera measurement equation are deferred until Section IV-C, since it depends on the parameterization that we have used for the landmarks, which is introduced in the subsequent section.

IV. Landmark Parameterization and Management

This section deals with the important problem of landmark parameterization and the associated problem of landmark estimation and management. The parameterization used is described in Section IV-A. Section Section IV-B then explains how to initialize the landmarks, and the associated measurement models are introduced in Section IV-C. Finally, landmark extraction and management is described in Section IV-D.

A. Landmark Parameterization

The standard way of describing the landmark position is to use the minimal Euclidean parameterization in terms of the landmark x, y, z position in the world coordinate frame. This parameterization has several problems; for instance it typically requires delayed [3] or complicated undelayed initialization [6]. The reason for this is that there is no easy way to provide a good uncertainty description of the fact that the depth (i.e., distance) to the landmark is unknown. An elegant approximation which acknowledges this fact is provided by the inverse depth parameterization introduced in [5]. This parameterization will be used in this work, allowing us to straightforwardly include the landmarks directly when they are first observed, i.e., undelayed. Furthermore, it allows us to make use of very distant landmarks without any problems. These landmarks are of no or little use for inferring the camera translation, but they are very useful when it comes to the camera orientation.

The main idea underlying the inverse depth parameterization is to acknowledge the fact that when a landmark is first observed we can draw a ray from the landmark $\boldsymbol{l}^w$ through the image plane to the current position of the camera's optical center $\boldsymbol{c}^w_0$. This results in the following parameterization of the landmark

$$\boldsymbol{l}^w = \boldsymbol{c}^w_0 + \frac{1}{\rho}\underbrace{\begin{pmatrix} \cos\phi^w \cos\theta^w \\ \cos\phi^w \sin\theta^w \\ \sin\phi^w \end{pmatrix}}_{\boldsymbol{m}(\phi^w, \theta^w)}, \quad (10)$$

where ρ denotes the inverse depth to the feature, and the direction to the landmark is given by $\boldsymbol{m}$, which is encoded using spherical coordinates ϕ^w and θ^w, i.e., the azimuth and the elevation, respectively. This leads to the following state vector describing the position of a landmark

$$x^{\mathrm{l}} = \begin{pmatrix} (\boldsymbol{c}^w_0)^T & \theta^w & \phi^w & \rho \end{pmatrix}^T \in \mathbb{R}^6. \quad (11)$$

For a graphical illustration of the parameterization (10), see Fig. 3.

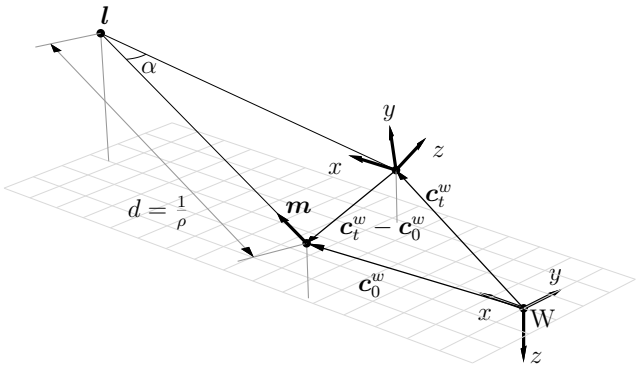

Fig. 3. The inverse depth parameterization used for the landmarks. The position of the landmark $\boldsymbol{l}$ is parameterized using the position $\boldsymbol{c}^w_0$ of the camera the first time the feature was seen, the direction $\boldsymbol{m}(\phi^w, \theta^w)$ and the inverse depth ρ.

B. Initialization

When a landmark is observed for the first time, it has to be initialized, which implies that initial values for the state vector and the corresponding covariance have to be assigned. The initial value for c^w_0 is directly given by $\hat{c}^w_{t|t-1} = \boldsymbol{c}^w_t$. Furthermore, the initial azimuth and elevation are given according to

$$\begin{pmatrix} \hat{\theta}^w_{t|t-1} \\ \hat{\phi}^w_{t|t-1} \end{pmatrix} = \begin{pmatrix} \arctan\left(\frac{m_y}{m_x}\right) \\ \arctan\left(\frac{m_z}{\sqrt{m_x^2 + m_y^2}}\right) \end{pmatrix} \quad (12)$$

where the directional vector $\boldsymbol{m}$ is given by

$$\boldsymbol{m} = R^{wc}\begin{pmatrix} 1 & y & z \end{pmatrix}^T \quad (13)$$

where $\begin{pmatrix} y & z \end{pmatrix}^T$ is the normalized (corrected for lens distortion and camera calibration) position of the landmark in the image

plane. Finally, the inverse depth ρ has to be initialized. For the present application it makes sense to initialize the landmarks quite far away, say according to

$$\hat{\rho}_{t|t-1} = \frac{1}{50}, \quad \text{Var}(\rho) = 0.1^2, \tag{14}$$

basically saying that we expect that the landmarks are 50 m in front of the car. The strength of the inverse depth parameterization is that it is straightforward to encode the fact that the depth is highly uncertain. Note that by having a standard deviation of 0.1, the 95% confidence interval for the inverse depth is $[0.22, \ldots, -0.18]$. The important thing is that the infinite depth is included in this interval, that is the range interval $[4.5, \ldots, \infty]$ is included in the above confidence interval. In other words, we include our uncertainty in the depth in the parameterization, without sacrificing the orientation information. Note that this is not possible using the standard Euclidean parameterization.

The new covariance matrix, taking the new landmark into account is given by

$$P_{t|t}^{\text{new}} = J \begin{pmatrix} P_{t|t} & 0 & 0 & 0 \\ 0 & P_{t|t}^{p^w p^w} & 0 & 0 \\ 0 & 0 & R_t & 0 \\ 0 & 0 & 0 & \text{Var}(\rho) \end{pmatrix} J^T, \tag{15}$$

$$J = \begin{pmatrix} I & & & & & & & & 0 & \\ \frac{\partial x^{\text{l}}}{\partial p^w} & 0 & \frac{\partial x^{\text{l}}}{\partial \psi} & 0 & 0 & \frac{\partial x^{\text{l}}}{\partial \varphi} & 0 & \ldots & 0 & \frac{\partial x^{\text{l}}}{\partial l^n} & \frac{\partial x^{\text{l}}}{\partial \rho} \end{pmatrix}, \tag{16}$$

where $P_{t|t}$ is the covariance matrix of the state x_t before the new landmark is included and $P_{t|t}^{p^w p^w}$ is the covariance of $\boldsymbol{p}_t^w$.

C. Measurement Model

We can without loss of generality assume that the position of the landmark in the image plane has been normalized. That is, we have compensated for lens distortions and the intrinsic camera parameters. Let y_t^{l} denote the normalized position (in the image plane) of the landmark at time t. The corresponding measurement equation will then be

$$y_t^{\text{l}} = h(x_t^{\text{l}}, p_t^w, \psi_t, \varphi_t) + e_t^{\text{c}}, \qquad e_t^{\text{c}} \sim \mathcal{N}(0, R^c), \tag{17a}$$

where

$$\begin{aligned} h(x_t^{\text{l}}, p_t^w, \psi_t, \varphi_t) = \mathcal{P}_n\big(R^{cb}(R^{bw}(\rho(\boldsymbol{c}_0^w - \boldsymbol{b}_t^w) \\ + \boldsymbol{m}(\theta^w, \phi^w) - \rho \boldsymbol{c}^b))\big). \end{aligned} \tag{17b}$$

Recall that $\mathcal{P}_n$ is used to denote the normalized pinhole projection according to (9).

D. Landmark Extraction and Management

We need a way to obtain measurements from the camera images that allows us to initialize landmarks according to the discussion in Section IV-B and make use of the measurement equation given in (17). The Harris corner detector [10] has been used in order to find interest points in the image, that are then used to initialize a landmark. In order to be able to track this landmark in subsequent images we need a descriptor of some form. Here, we have chosen to simply make use of an image patch. More specifically, we store an 11×11 patch of the image with its center at the detected interest point. In order to find this patch in the subsequent images we make use of the normalized cross-correlation (NCC); see e.g., [17] for details. Since we have an estimate of the vehicle motion between the successive images, this information can be used to predict where in the image we would expect this interest point to appear in the next image. A search region is then formed around this predicted position and the NCC is computed for all pixels within this region. If there is a significant maximum present, we choose this as the new measurement $y_{j,t}^{\text{l}}$ of that particular landmark. We require the maximum to be significantly larger than the second largest component which provides a good way of rejecting spurious features. This proved to add noticeable robustness to the estimates. Furthermore, if one associated interest point is far from its predicted position and the other interest points lie close to their predicted positions, this interest point is considered an outlier and hence not used as a measurement. Finally, we search new areas for new interest points. If there are new interest points found, these are initialized according to Section IV-B. Furthermore, landmarks that are behind the vehicle are removed from the map. Note, that this is no restriction in our case, since we do not expect to revisit the current position anytime soon. The procedure described above is summarized in Algorithm 1 in the subsequent section.

There are of course alternatives to the choices made above. The obvious problem with using image patches is that they are not invariant to changes in scale and rotation. This is something that can be overcome by using for example SIFT [14] interest points. Nevertheless, the results obtained using simple image patches are satisfactory and they are simple to use. Furthermore, it is straightforward to change to any other detector and data association method, as long as the output is a reliable set of correspondences between 2D positions in the image plane $y_{j,t}^{\text{l}}$ and the corresponding landmark state $x_{j,t}^{\text{l}}$ in the 3D world. This set will then serve as measurements in our estimator, which will be further explained in the subsequent section.

V. Sensor Fusion

In Section I we defined sensor fusion as the process of using information from *several different sensors* to compute an *estimate* of the state of a *dynamical system*. The dynamical system under study in this work and the associated measurement models are abstractly described by

$$x_{t+1}^{\text{v}} = f(x_t^{\text{v}}, u_t) + B(x_t^{\text{v}})v_t, \qquad v_t \sim \mathcal{N}(0, Q_t), \tag{18a}$$

$$x_{i,t+1}^{\text{l}} = x_{i,t}^{\text{l}}, \qquad i = 1, \ldots, M_t, \tag{18b}$$

$$y_t^{\text{v}} = h^v(x_t^{\text{v}}) + e_t^{\text{v}}, \qquad e_t^{\text{v}} \sim \mathcal{N}(0, R_t^{\text{v}}), \tag{18c}$$

$$y_{j,t}^{\text{l}} = h^{\text{l}}(x_t^{\text{v}}, x_{j,t}^{\text{l}}) + e_t^{\text{l}}, \qquad e_t \sim \mathcal{N}(0, R_t^{\text{l}}), \tag{18d}$$

where $f(x_t^{\text{v}}, u_t)$ and $B(x_t^{\text{v}})$ are given in (3), (5) and (6), respectively. Furthermore, the proprioceptive measurement equation $h^{\text{v}}(x_t^{\text{v}})$ and the camera related measurement equation $h^{\text{l}}(x_t^{\text{v}}, x_{j,t}^{\text{l}})$ are given by (8) and (17), respectively. In

describing the algorithm that we have used it is better to work with this more general model (18). This is a nonlinear model, implying that we are forced to an approximation of some sort in order to compute the state estimates. The most commonly used approximation is provided by the extended Kalman filter. The idea underlying the EKF is very simple: approximate the nonlinear model with a linear model subject to Gaussian noise and apply the Kalman filter [12] to this approximation. This linearization is standard, but we give the Jacobians here for future reference,

$$F_t^{\mathrm{v}} = \left.\frac{\partial f(x^{\mathrm{v}}, u_t)}{\partial x^{\mathrm{v}}}\right|_{x^{\mathrm{v}}=\hat{x}^{\mathrm{v}}_{t|t}} \qquad G_t^{\mathrm{v}} = B(\hat{x}^{\mathrm{v}}_{t|t}) \tag{19a}$$

$$H_t^{\mathrm{v}} = \left.\frac{\partial h^{\mathrm{v}}(x^{\mathrm{v}})}{\partial x^{\mathrm{v}}}\right|_{x^{\mathrm{v}}=\hat{x}^{\mathrm{v}}_{t|t-1}} \tag{19b}$$

$$H_{j,t}^{\mathrm{l}} = \left.\frac{\partial h^{\mathrm{l}}(x^{\mathrm{v}}, x_j^{\mathrm{l}})}{\partial x^{\mathrm{v}}, x_j^{\mathrm{l}}}\right|_{(x^{\mathrm{v}},x_j^{\mathrm{l}})=(\hat{x}^{\mathrm{v}}_{t|t-1},\hat{x}^{\mathrm{l}}_{j,t|t-1})} \tag{19c}$$

The state estimate is parameterized using a mean value $\hat{x}_{t|t}$ and a covariance $P_{t|t}$ according to

$$\hat{x}_{t|t} = \begin{pmatrix} \hat{x}^{\mathrm{v}}_{t|t} \\ \hat{x}^{\mathrm{l}}_{t|t} \end{pmatrix}, \qquad P_{t|t} = \begin{pmatrix} P^{\mathrm{v}}_{t|t} & P^{\mathrm{vl}}_{t|t} \\ P^{\mathrm{lv}}_{t|t} & P^{\mathrm{l}}_{t|t} \end{pmatrix} \tag{20}$$

The equations for updating the mean and the covariance over time are given by the EKF. For a solid account of the EKF we refer to [1, 11]. To be specific, the measurement update is given by

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t(y_t - h(\hat{x}_{t|t-1})), \tag{21a}$$

$$P_{t|t} = P_{t|t-1} - K_t H_t P_{t|t-1}, \tag{21b}$$

$$K_t = P_{t|t-1} H_t^T (H_t P_{t|t-1} H_t^T + R_t)^{-1}, \tag{21c}$$

where $(h = h^{\mathrm{v}}, H_t = H_t^{\mathrm{v}})$ or $(h = h^{\mathrm{l}}, H_t = H_{j,t}^{\mathrm{l}})$, depending on whether we are dealing with proprioceptive or exteroceptive measurements. If there are several measurements available at the same time we can either process them one at a time according to (21), or we can augment all the measurements and the corresponding Jacobians into one big measurement vector and one big gradient and process them all at once.

Once the new measurements have been included according to (21) we have to predict the states forward in time in order to be able to accommodate the new measurements at time $t+1$. This is accomplished by the following time update,

$$\hat{x}^{\mathrm{v}}_{t+1|t} = f(\hat{x}^{\mathrm{v}}_{t|t}, u_t), \tag{22a}$$

$$\hat{x}^{\mathrm{l}}_{t+1|t} = \hat{x}^{\mathrm{l}}_{t|t}, \tag{22b}$$

$$P_{t+1|t} = \begin{pmatrix} F_t^{\mathrm{v}} P^{\mathrm{v}}_{t|t} (F_t^{\mathrm{v}})^T & F_t^{\mathrm{v}} P^{\mathrm{l}}_{t|t} \\ P^{\mathrm{l}}_{t|t} (F_t^{\mathrm{v}})^T & P^{\mathrm{l}}_{t|t} \end{pmatrix} + \begin{pmatrix} G_t^{\mathrm{v}} Q_t (G_t^{\mathrm{v}})^T & 0 \\ 0 & 0 \end{pmatrix} \tag{22c}$$

Note that the uncertainty in the description for the landmarks $P^{\mathrm{l}}_{t|t}$ is left unchanged by the prediction. This is according to intuition, since the landmark positions are not affected by the motion of the vehicle. Furthermore, it is worth noting that this framework allows us to straightforwardly handle the fact that the proprioceptive and the exteroceptive sensors operates at different sampling frequencies. The sensor fusion algorithm is now summarized in Algorithm 1, where we, for reasons of brevity, refrain from repeating the equations. Instead we simply provide references.

Algorithm 1 Sensor fusion

1) Initialize the vehicle state $\hat{x}_{1|0}, P^{\mathrm{v}}_{1|0}$ and use the first image to initiate the first landmarks $x^{\mathrm{l}}_{j,1|0}, j = 1, \ldots, M_1$ using (12) – (15).
2) If there are new proprioceptive measurements (7) available, incorporate this information using (21) – (22).
3) Predict landmark positions in the new image.
4) Perform data association using the normalized cross-correlation.
5) Detect and remove outliers.
6) Update the vehicle state x^{v}_t and the landmarks $x^{\mathrm{l}}_{i,t}$ that passed the outlier test using the corresponding measurements $y^{\mathrm{l}}_{j,t}$ via (21) and (17).
7) In image areas without landmarks, search for new landmarks using the Harris detector and if available, initialize new landmarks according to (12) – (15).
8) Repeat from 2.

VI. Experiments and Results

In order to illustrate the performance of the method for ego-motion and indirect road geometry estimation developed in this work we have made use of measurement sequences recorded during night-time driving on rural roads in Sweden. There is no ground truth available. However, we are still able to show that the FIR camera is very useful in order to solve the estimation problem under study. We will show this simply by reprojecting the estimated ego-motion onto the first image, i.e., we plot the estimated position of the vehicle expressed in the world coordinate frame $\hat{p}^w_{t|t}$.

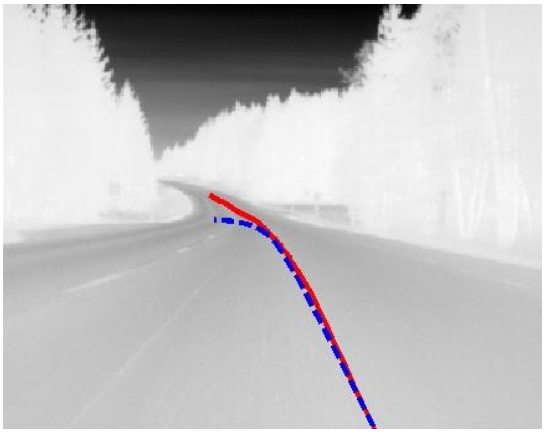

Fig. 4. Illustrating a traffic scenario, where the road geometry exhibits significant change in the z direction. The ego-motion estimates are reprojected into the first image. Clearly, the estimates are better using the FIR camera (red, solid curve) than just using the proprioceptive measurements (blue, dashed curve).

In Fig. 4 we show a traffic scenario, where the road geometry is clearly changing in all three dimensions. For this case we would expect that the information from the FIR camera is most useful, since the proprioceptive sensors used in this work only provide 2D information. As we shall see, the camera not only allows us to gain observability in the third dimension, it also improves the estimates in 2D. In Fig. 5 we show part of the motion in the world coordinate frame.

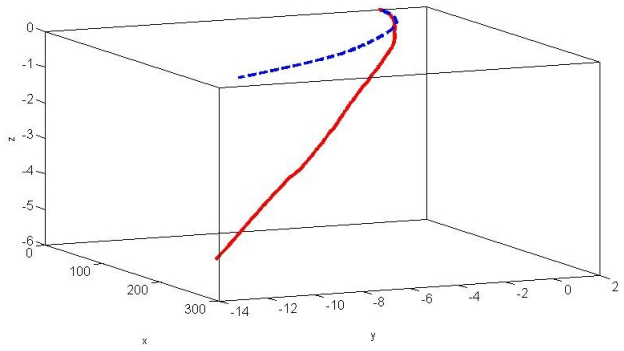

Fig. 5. Ego-motion estimates reprojected onto the image plane in Fig. 4. The result using the FIR camera is the red, solid curve and the results using only the proprioceptive measurements is the blue, dashed curve.

In Fig. 6 we show a traffic scenario, where the ground is almost flat, which means that the vehicle translation only takes place in the xy plane. This figure indicates that the information provided by the FIR camera is useful for planar scenarios.

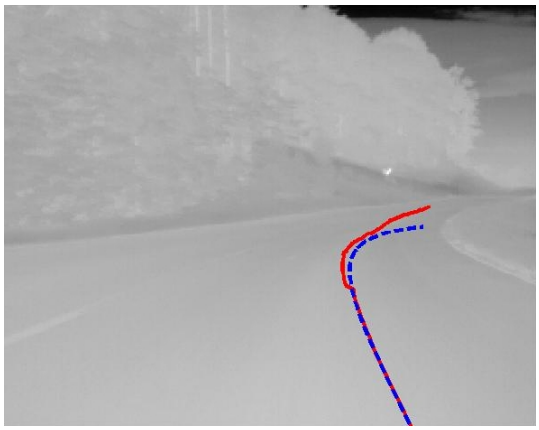

Fig. 6. This image is from an area where the ground is almost flat. Hence, the vehicle translation takes place in the $x-y$ plane. Clearly, the ego-motion estimates are better using the FIR camera (red, solid curve) than just using the proprioceptive measurements (blue, dashed curve).

VII. Conclusion and Future Work

We have in this contribution formulated and solved a sensor fusion problem based on measurements from standard proprioceptive sensors and a far infrared camera. The solution provides information about the ego-vehicle position in 3D and orientation in 2D. The approach has been evaluated using real and relevant data from rural roads in Sweden. The results illustrates the fact that the FIR images can be used to improve the ego-motion estimates, especially during night time driving, where normal cameras cannot be used.

Future work include using more advanced vehicle models, including better models of the pitch dynamics, float angle and slip angles for example. In this way we should be able to make even better use of the information from the camera to compute better estimates of the ego vehicle motion.

References

[1] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*, ser. Information and system science series. Englewood Cliffs, NJ, USA: Prentice Hall, 1979.

[2] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): Part II," *IEEE Robotics & Automation Magazine*, vol. 13, no. 3, pp. 108–117, Sep. 2006.

[3] M. Bryson and S. Sukkarieh, "Bearings-only SLAM for an airborne vehicle," in *Proceedings of the Australasian Conference on Robotics and Automation*, Sydney, Australia, Dec. 2005.

[4] Y. Cheng, M. W. Maimone, and L. Matthies, "Visual odometry on the Mars exploration rovers," *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 54–62, Jun. 2006.

[5] J. Civera, A. J. Davison, and J. M. M. Montiel, "Inverse depth parameterization for monocular SLAM," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 932–945, Oct. 2008.

[6] A. J. Davison, I. Reid, N. Molton, and O. Strasse, "MonoSLAM: Real-time single camera SLAM," *IEEE Transactions on Patterns Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1052–1067, Jun. 2007.

[7] E. D. Dickmanns, *Dynamic Vision for Perception and Control of Motion*. London, United Kingdom: Springer, 2007.

[8] E. D. Dickmanns and B. D. Mysliwetz, "Recursive 3-D road and relative ego-state recognition," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 14, no. 2, pp. 199–213, Feb. 1992.

[9] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping (SLAM): Part I," *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99–110, Jun. 2006.

[10] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proceedings of the 4th Alvey Vision Conference*, Manchester, UK, 1988, pp. 147–151.

[11] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation*, ser. Information and System Sciences Series. Upper Saddle River, NJ, USA: Prentice Hall, 2000.

[12] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME, Journal of Basic Engineering*, vol. 82, pp. 35–45, 1960.

[13] Q. Lin, F. Tjärnström, J. Roll, and B. Wass, "Developing a far infrared based night-vision system with pedestrian detection," in *Proceedings of the VDI Optische Technologien in der Fahrzeugtechnik*, Leonberg, Germany, Jun. 2008.

[14] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[15] C. Lundquist and T. B. Schön, "Road geometry estimation and vehicle tracking using a single track model," in *Proceedings of the IEEE Intelligent Vehicles (IV) Symposium*, Eindhoven, The Netherlands, Jun. 2008.

[16] ——, "Recursive identification of cornering stiffness parameters for an enhanced single track model," in *Proceedings of the 15th IFAC Symposium on System Identification (SYSID)*, Saint-Malo, France, Jul. 2009, accepted for publication.

[17] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry, *An invitation to 3-D vision – from images to geometric models*, ser. Interdisciplinary Applied Mathematics. Springer, 2006.

[18] D. Nistér, O. Neroditsky, and J. Bergen, "Visual odometry for ground vehicle applications," *Journal of Field Robotics*, vol. 23, no. 1, pp. 3–20, 2006.

[19] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, ser. Intelligent Robotics and Autonomous Agents. Cambridge, MA, USA: The MIT Press, 2005.