Multi-sensors data fusion using dynamic bayesian network for robotised vehicle geo-localisation

Cindy Cappelle, Maan E. El Najjar, Denis Pomorski
LAGIS – CNRS UMR 8146
USTL – Polytech’Lille – Cité Scientifique
59655 Villeneuve d’Ascq, FRANCE
Cindy.Cappelle@polytech-lille.fr
Maan.El-Badaoui-El-Najjar@univ-lille1.fr
denis.pomorski@univ-lille1.fr

François Charpillet
LORIA – INRIA Lorraine
MAIA Team
Campus Scientifique, BP239
54506 Vandoeuvre-lès-Nancy, FRANCE
francois.charpillet@loria.fr

Abstract – This paper presents an outdoor geo-localisation method, which integrates several information sources: measurements from GPS, incremental encoders and gyroscope, 2D images provided by an on-board camera and a virtual 3D city model. A 3D cartographical observation of the vehicle pose is constructed. This observation is based on the matching between the acquired 2D images and the virtual 3D city model. This estimation is especially useful during long GPS outages to correct the drift of the only dead-reckoning localisation or when the GPS quality is deteriorated due to multi-path, satellites masks and so on particularly in urban environments. Moreover, the various sensors measurements are fused in Dynamic Bayesian Network formalism in order to provide a continuous estimation of the pose.

Keywords: Geo-localisation, Virtual 3D City Model, 3D Geographical Information System (3D-GIS), Global Positioning System (GPS), Vision, Dynamic Bayesian Network (DBN), 2D/3D Matching

1 Introduction

In order to take into account the road safety, more and more Advanced Driver Assistance Systems (ADAS) are integrated. Several applications of safety and driving comfort can be cited: Adaptive Cruise Control (ACC), adaptive lighting control, collision warning, lane departure warning … Many of these operations require real-time positioning of the vehicle with respect to the environment. Moreover, autonomous urban vehicles rely also a lot on the vehicle geo-localisation.

Positioning systems often rely on GPS, because of its affordability and convenience. However, GPS suffers from satellite masks occurring in urban environments, under bridges, tunnels or in forests. GPS appears then as an intermittently-available positioning system that needs to be backed up by a dead-reckoning (DR) system [1]. The use of dead-reckoning sensors is a simple and efficient method to tackle GPS in order to provide a continuous position estimation. In this paper, incremental encoders placed on the rear wheels and a gyroscope are used for this purpose.

A dead-reckoning pose is so obtained by integrating the elementary distance covered by the rear wheels and the elementary rotation of the wheels using a differential odometric model. In this work, the multi-sensor data fusion is performed in the Dynamic Bayesian Network (DBN) formalism.

But in case of long GPS outages or GPS low accuracy periods, the only dead-reckoning (DR) localisation can drift. Thus we propose to construct another observation of the pose which is based on the exploitation of a new source of information which is a virtual 3D city model of the environment where the vehicle navigates. This virtual 3D city model (also called geographical 3D model, textured geo-referenced 3D database or 3D map in the literature) is managed by a 3D Geographical Information System (3D-GIS). In order to compute an estimation of the absolute vehicle pose, the 3D model is matched with the current 2D image provided by an embedded camera. This observation, which is called “3D cartographical observation” in the following of this paper, can be used to correct the dead-reckoning localisation drift and/or to improve the GPS accuracy.

The outline of this paper is as follows. In section 2, the virtual 3D city model managed by a 3D-GIS is described. Then section 3 details the developed geo-localisation method. In this section, we present firstly the construction of the 3D cartographical observation i.e. the observation of the vehicle pose based on the matching between the virtual 3D city model and the acquired images. Then the data fusion algorithm which is designed in the DBN formalism is detailed. In section 4, experimental results obtained with real data are presented, before we conclude in section 5.

2 Virtual 3D city model and 3D-GIS

2.1 Overview

More and more applications require 3D models of the buildings, cities, landscapes, in addition to 2D digital maps commonly managed by Geographical Information Systems (GIS). These 3D models can be automatically
generated using aerial images, 2D digital map, laser profiler data, land registry and so on [11]. The German GEIST project [8] shows that such 3D models enable a better understanding of spatial relationships and are therefore ideal for demonstration and presentation purposes. The aim of GEIST project is to provide the users with a system that offers a personalized and lively tour in historic sites by superimposing virtual objects (e.g. buildings) showing the historical situation. Virtual 3D city model are used a lot in architecture, and town and country planning fields too. A 3D model of the environment can be a support tool to check the coherence of an architectural project or to detect possible anomalies. Virtual 3D city models are for example used during district redevelopment in order to measure the impact of the project on the environment.

Thanks to such applications, virtual 3D city models meet a rapid expansion [10,11,12]. For example, all the major cities in Japan have been covered since 2002 and are updated every six months. The French national geographic institute IGN (www.ign.fr/) which provides 2D maps aims to create the geographical 3D map of France for 2010 in its Bati3D project. The indisputable appeal of virtual 3D city models leads us to propose an utilisation of these ones in the intelligent transportations domain. We propose to construct a 3D cartographical observation of the vehicle pose by fusing 3D model with 2D images acquired by an on-board camera.

2.2 The used virtual 3D city model

Figure 1: Screenshots of the 3D database viewer showing Nancy downtown

The used geographical 3D model is a Tecnomade product (http://www.tecnomade.fr/). According to the constructor, the model accuracy is 1 meter. This model maps the Stanislas Square of Nancy, France (Fig.1). In order to navigate in real-time in the 3D database, a three dimensional geographical information system (3D-GIS) is required. A GIS is a computer system capable of integrating, storing, editing, analysing, sharing and displaying geographically-referenced information. Thus, we developed a 3D-GIS adapted to our applications in robotics and intelligent vehicle. The view seen by a vehicle whose path is given by its successive positions in a world geographical system as WGS84 can then be displayed in real-time. This has been possible by determining the transformation between the local frame and the world frame, i.e. the geo-referencing of the 3D model.

The inputs of the 3D-GIS are:
- the 3D model database
- the calibration parameters of the virtual camera, which is the observer

The extrinsic parameters are the 6 degrees of freedom (position and orientation) of the virtual camera with respect to the local frame attached to the 3D model. The intrinsic parameter is the horizontal field of view (FOV) of the virtual camera.

Figure 2: 3D-GIS output functionalities

The five outputs are (Fig.2):
- the bitmap virtual image which is the view of the scene captured by the virtual camera calibrated according to the calibration parameters (Fig.2a)
- the bitmap image of the segments of the database that are visible from the position and orientation of the virtual camera (Fig. 2b)
- the depth image which corresponds to the virtual image: the farther the object is the brighter the pixel is and the nearer the 3D point is the darker the pixel is (Fig. 2c)
- the depth file which is a text file that contains the depth information in meters for each pixel of the virtual image, this information is extracted from the Z-Buffer of the video card and gives the distance between the virtual camera and the 3D points represented by each pixel (Fig. 2d)
- a XML file which contains for each 3D segments of the model, the coordinates of the extremities and the visibility status of the segment (Fig. 2e)

3 Geo-localisation method

3.1 Overview

The synoptic of our approach is described on Figure 3. When a GPS measurement is available and the GPS quality is good enough, a DBN fuses on the one hand the predicted pose obtained thanks to incremental encoders and gyroscope and on the other hand the GPS measurement. The GPS quality is considered good enough when its quality index is greater or equal than two \( \left( Q_{\text{GPS}} \geq 2 \right) \) i.e. GPS operates in DGPS or RTK mode with respectively metric or centimetric accuracy. But during GPS outages or low quality periods, the 3D cartographical observation is used in order to correct the DR predicted pose.

3.2 3D cartographical observation

The 3D cartographical observation is the pose of the vehicle which is computed using a matching method that makes correspondences between a view in the virtual 3D city model and 2D images provided by an embedded camera. In this section, we described how the 3D cartographical observation is constructed.

The developed method is described in Figure 4. The first step is to extract the two needed images:
- the real image which is the image captured by the on-board camera
- the virtual image which is the image produced by the 3D-GIS

Figure 3: Description of the geo-localisation method

Figure 4: Synoptic of the construction of the 3D cartographical pose observation

In order to extract the virtual image, 3D-GIS needs the 3D model database and an initial estimation of the pose. This initial estimation is the pose predicted by the odometric model that uses the incremental encoders and gyroscope measurements. The virtual image corresponds
to the view of the virtual camera whose pose is given by
the initial estimation, which is in the neighborhood of
the real position of the vehicle. The second step of
the algorithm is to detect the Harris corner points of the real
and virtual images [5]. Then, the 2 sets of Harris points
had to be matched. To do that, for each interest point
detected in the real image, a correlation score is computed
with all the interest points detected in the virtual image
in a search region.

Next, the RANSAC (RANdom SAmple Consensus)
[4] method is applied in order to eliminate outliers. The
RANSAC algorithm is a popular way to make model
fitting in the presence of noisy data. It is an iterative
method to estimate parameters of a mathematical model
from a set of observed data which contains outliers.
The mathematical model used in our case is the 2D
homography, which defines the relation between the real
image and the virtual image.

After the matching and outliers elimination are done,
we have a set of 2D points in real image matched with
a set of 2D points in virtual image. We have now to
calculate the 3D coordinates of the points that are
projected to the detected Harris points in the virtual
image plane. The 3D@GIS returns not only the virtual image
but also a text file containing the depth information (Fig.3d).
This depth information is the distance in meters between
the virtual camera (i.e. the a priori estimation of the pose)
and the 3D points represented by the pixels of the virtual
image. This depth information in addition to the position
of the pixel in the virtual image are used in order to
calculate the 3D coordinates, in the local frame attached to
the 3D model, of the Harris points detected in the virtual
image.

Consider \( p \) a Harris point of the virtual image which
was matched with a Harris point of the real image. Let
denote \((u, v)\) the coordinates of this pixel \( p \) in the image
plane. Let denote \( P \) the 3D point which is projected at \( p \)
in the virtual image. As we consider that the pitch and roll
are negligible, the coordinates \((x_C, y_C, z_C)^T\) of \( P \) in
the camera frame can be computed as follows (Fig.5):

\[
\begin{align*}
    z_C &= -\frac{d}{m} f \\
    x_C &= -z_C \cdot \frac{v - v_0}{f} \\
    y_C &= z_C \cdot \frac{u - u_0}{f}
\end{align*}
\]

with
- \((u_0, v_0)^T\): the coordinates of the pixel situated on
the center of the image
- \(f\): the focal length in pixels
- \(d\): the distance in meters between the virtual
camera \( C \) (i.e. the predicted pose) and the 3D
point \( P \), this distance is extracted from the depth
file provided the 3D-GIS
- \(m\): the distance in pixels between the virtual
camera \( C \) and the pixel \( p \)

\[
m = \sqrt{f^2 + (u - u_0)^2 + (v - v_0)^2}
\]

The following relation permits next to determine the
coordinates \((x_M, y_M, z_M)^T\) of \( P \) in the frame attached to
the 3D model:

\[
\begin{pmatrix}
    x_M \\
    y_M \\
    z_M
\end{pmatrix} = R \cdot \begin{pmatrix}
    x_C \\
    y_C \\
    z_C
\end{pmatrix} + T
\]

with \((R, T)\) the rotation matrix and the translation vector
from the camera frame to the model frame.
sum of the square of the re-projection error of each match. The parameters \( R \) and \( T \) that minimize the cost function \( f \) permit to determine the position and the orientation of the camera in the 3D model frame, that is to say the 3D cartographical observation.

3.3 Data fusion with DBN

3.3.1 Dynamic Bayesian Network

A Bayesian Network (BN) is a probabilistic graphical model that represents a set of variables and their probabilistic independencies. A BN is a directed acyclic graph whose nodes represent variables and whose arcs encode conditional independencies between the variables.

A BN is defined by 2 aspects:
- a graph that determines the structure of the model:
  \[ G = (V, E) \]
  with \( V = \{X_1, \ldots, X_n\} \) the set of nodes of \( G \) and \( E \) the set of arcs of \( G \) if \( X_i \in Pa(X_j) \) where \( Pa(X_j) \) are the parents of \( X_j \)
- a set of local probability distributions that defines the parameters of the BN:
  \[ p(X_i|Pa(X_i)) \]

In a BN relative to a set of variables, the joint probability distribution (JPD) of the node values can be written as the product of the local distributions of each node and its parents:

\[ P(X_1, \ldots, X_n) = \prod_{i=1}^{n} p(X_i|Pa(X_i)) \]

For more details about Bayesian Networks, please refer to [3,7].

A Dynamic Bayesian network (DBN) is a BN used to model a temporal stochastic process. It can be created by specifying network (structure and parameters) for two consecutive “time slices” and then “unrolling” it into a static network of the required size. DBNs generalize two well-known signal modelling tools: Kalman filters for continuous state Linear Dynamic System (LDS) and Hidden Markov Models (HMMs) for classification of discrete state sequences. It has been shown that estimation in LDSs and inference in HMMs are special cases of inference in DBNs [6].

Consider a dynamical system whose parameters evolve in time according to some known model. This system can be described using the following set of state-space equations:

\[
X_i = A_i X_{i-1} + v_i \\
Y_i = C_i X_i + w_i
\]

This model is called Linear Dynamical System (LDS), where \( X_i \in \mathbb{R}^n \) is the hidden state variable at time \( t \), \( Y_i \in \mathbb{R}^m \) is the observation at time \( t \), and \( v_i \sim N(0, Q_i) \) and \( w_i \sim N(0, R_i) \) are independent Gaussian noise. The parameters of model: \( A_i, C_i, Q_i \) and \( R_i \) are assumed to be time-invariant. A LDS can be represented as a DBN (Fig.6).

![Figure 6: A DBN representation of a LDS](image_url)

The variables \( X_i \) and \( Y_i \) are continuous. Other observations can be introduced in this model. The conditional probability distributions for this model are as follows:

\[
P(X_i = x_i / X_{i-1} = x_{i-1}) = N(Ax_{i-1}, Q) \\
P(Y_i = y_i / X_i = x_i) = N(Cx_i, R)
\]

An important issue in BN is the computation of posterior probabilities of variables given observations. Several researchers developed algorithms to compute the exact inference and/or the approximate inference for different distributions. The most commonly algorithm used to compute the exact inference is known as the JLO (Jensen, Lauritzen, Olesen) algorithm [4]. The JLO algorithm is a recursive message passing algorithm that works on the junction tree of the BN. In the case of Kalman filter estimation, we need to find the posterior: \( P(\frac{X_i}{Y_i}) \).

3.3.2 DBN algorithm for sensors fusion

In this work, Dynamic Bayesian Network is the formalism chosen to fuse the various sensors measurements. Next paragraphs of this section present the evolution model and the GPS measurement model needed to fuse incremental encoders, gyroscope and GPS in DBN formalism.
3.3.2.1 Evolution model

The evolution model of the vehicle provides a prediction of the pose by the previous estimation of the pose and the measurements from the incremental encoders and the gyroscope. This model is non-linear:

\[
X_{k+1} = \begin{pmatrix} x_{k+1} \\ y_{k+1} \\ \theta_{k+1} \end{pmatrix} = f(X_k, U_k, \gamma_k) + \alpha_k
\]  

(4)

where:
- \( X_k = (x_k, y_k, \theta_k) \) is the state vector at time \( k \) that is composed by the position and orientation of the vehicle
- \( U_k = (d_k, \omega_k) \) is the vector of the measured inputs consisting of the elementary distance covered by the rear wheels (\( d_k \) is given by the incremental encoders) and the elementary rotation (\( \omega_k \)) provided by the gyroscope
- \( \gamma_k \) represents the measurement error of the inputs
- \( \alpha_k \) is the process noise
\( \alpha_k \) and \( \gamma_k \) are assumed to be uncorrelated and zero mean noise.

The evolution model can be expressed by:

\[
X_{k+1} = \begin{pmatrix} x_{k+1} \\ y_{k+1} \\ \theta_{k+1} \end{pmatrix} = \begin{pmatrix} x_k + d_k \cos(\theta_k + \omega_k/2) \\ y_k + d_k \sin(\theta_k + \omega_k/2) \\ \theta_k + \omega_k \end{pmatrix}
\]  

(5)

3.3.2.2 GPS observation model

When a GPS measurement is available and the GPS quality is good enough, a correction of the predicted pose is performed.

The measurement model is defined as:

\[
Y_k = \begin{pmatrix} x_{k,gps} \\ y_{k,gps} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_k \\ y_k \\ \theta_k \end{pmatrix} + \beta_{gps}
\]  

(6)

The NMEA sentence GPGST given by the GPS receiver also provides the deviation of the measurement. The covariance matrix of the GPS error can then be written as follows:

\[
Q_{\beta,gps} = \begin{pmatrix} \sigma_x^2_{gps} & Q_{x,y,gps} \\ Q_{y,x,gps} & \sigma_y^2_{gps} \end{pmatrix}
\]  

(7)

4 Experimental results

In order to test our approach, we carried out experiments in Nancy downtown with our electric vehicle, called CyCab, which is a Robosoft product (www.robosoft.fr).

4.1 Multi-sensors fusion results with DBN

This first part of results concerns the results obtained when GPS measurements are available. The pose of the vehicle is then estimated by fusing GPS measurements and data provided by DR sensors (incremental encoders and gyroscope) with a dynamic bayesian network (Fig.7). The test trajectory is about 600m long and the duration time is about 2 minutes.

A GPS outage is simulated in order to point up the drift of the DR localisation in case of long GPS absences. At the end of the GPS outage, the error is more than 3 meters.

![Figure 7: GPS-DR fusion with DBN](image)

4.2 3D cartographical observation

The previous results showed that during long GPS outages, the DR localisation drift. Thus, when GPS can not be used to put right the DR data, we compute an observation of the vehicle pose by matching the virtual 3D city model with captured 2D images. In Figure 8, the results of the computation of the 3D cartographical...
observation are displayed for the same experimentation. The continuous blue line is the true trajectory of the vehicle, i.e. the data of a centimetric RTK GPS. The red crosses represent the 3D cartographical observation.

![Figure 8: 3D cartographical observation results](image)

The errors of the 3D cartographical observation are plotted on Figure 9. For this experimentation, the maximum error is about 6.2m on x-axis and 8.5m on y-axis. The mean error is about 1.35m on x-axis and y-axis. One can notice that the reached accuracy is of the same order as the accuracy of the GPS in DGPS WAAS/EGNOS mode.

The obtained results could be improved by taking into account pitch and roll of the vehicle and by integrating a model correction with respect to focal distances of both real and virtual camera systems. Moreover, the initial estimation of the vehicle pose has to be accurate enough otherwise the Harris points matching failed.

**Conclusion**

This paper proposes a method to take advantage of a virtual 3D city model for geo-localisation in urban areas. The obtained results are promising. But we can suggest improvements in order to obtain more accurate 3D cartographical observations. During the computation of the 3D cartographical observation, the pitch and roll are neglected in this work, but without this simplification, the pose should be more accurate. Moreover, one can remark that the 3D cartographical observation is not sufficiently accurate when the used predicted pose is too far from the real one because of Harris point matching stage.

The choice of DBN for multi-sensors fusion rests on some proprieties of DBN that we would exploit in a forthcoming work. In this future work, our aim is to integrate the 3D cartographical observation of the pose in the DBN based fusion algorithm. DBN formalism will then permit to manage several hypotheses in ambiguous situations (i.e. several 3D cartographical observations are taken into account) until the situations become unambiguous [9]. Moreover, with this generic formalism, the command aspect for applications such as platooning could also be integrated.

Moreover, the only localisation aspect has been treated in this paper. But, we consider that in the future, such a 3D-GIS with the functionalities of standard 2D-GIS will able to accomplish services as fleet management, itinerary calculation ... The benefits of such a virtual 3D city model have already been illustrated for detecting, tracking and geo-localising obstacles using a GPS, a monovision system and the same 3D model in [2].

**Acknowledgment**

This work is realized in the frame of the French national project “Data Fusion for System Supervision: Application on autonomous Navigation System” of the GIS 3SGS and the regional project “Accurate Navigation for Vehicles in Urban Area” of CISIT (Nord-Pas-de-Calais Region).
References


