Camera Based Terrain Navigation

Examensarbete utfört i Reglerteknik
vid Tekniska högskolan i Linköping
av

Peter Rosander

LITH-ISY-EX--09/4179--SE
Linköping 2009
Camera Based Terrain Navigation

Examensarbete utfört i Reglerteknik
vid Tekniska högskolan i Linköping
av

Peter Rosander

LITH-ISY-EX--09/4179--SE

Handledare: David Törnqvist
            ISY, Linköpings universitet

            Hans Bohlin
            Saab Aerotech Ab

Examinator: Thomas Schön
            ISY, Linköpings universitet

Linköping, 12 January, 2009
Kamerabaserad terrängnavigering

The standard way for both ground and aerial vehicles to navigate is to use an Inertial Navigation System, INS, containing an Inertial Measurement Unit, IMU, measuring the acceleration and angular rate, and a GPS measuring the position. The IMU provides high dynamic measurements of the acceleration and the angular rate, which the INS integrates to velocity, position and attitude, respectively. While being completely impossible to jam, the dead-reckoned estimates will drift away, i.e., the errors are unbounded. In conjunction with a GPS, providing low dynamic updates with bounded errors, a highly dynamic system without any drift is attained. The weakness of this system is its integrity, since the GPS is easily jammed with simple equipment and powered only by a small standard battery. When the GPS is jammed this system falls back into the behavior of the INS with unbounded errors. To counter this integrity problem a camera can be used as either a back up to the GPS or as its replacement. The camera provides images which are then matched versus a reference, e.g., a map or an aerial photo, to get similar estimates as the GPS would provide. The camera can of course also be jammed by blocking the view of the camera with smoke. Bad visibility can also occur due to bad weather, but a camera based navigation system will definitely be more robust than one using GPS.

This thesis presents two ways to fuse the measurements from the camera and the IMU, both of them utilizing the Harris corner detector to find point correspondences between the camera image and an aerial photo. The systems are evaluated by simulated data mimicking both a low and a high accuracy IMU and a camera taking snapshots of the aerial photo. Results show that for the simulated camera images the implemented corner detector works fine and that the overall result is comparable to using a GPS.
Abstract

The standard way for both ground and aerial vehicles to navigate is to use an Inertial Navigation System, INS, containing an Inertial Measurement Unit, IMU, measuring the acceleration and angular rate, and a GPS measuring the position. The IMU provides high dynamic measurements of the acceleration and the angular rate, which the INS integrates to velocity, position and attitude, respectively. While being completely impossible to jam, the dead-reckoned estimates will drift away, i.e., the errors are unbounded. In conjunction with a GPS, providing low dynamic updates with bounded errors, a highly dynamic system without any drift is attained. The weakness of this system is its integrity, since the GPS is easily jammed with simple equipment and powered only by a small standard battery. When the GPS is jammed this system falls back into the behavior of the INS with unbounded errors. To counter this integrity problem a camera can be used as either a back up to the GPS or as its replacement. The camera provides images which are then matched versus a reference, e.g., a map or an aerial photo, to get similar estimates as the GPS would provide. The camera can of course also be jammed by blocking the view of the camera with smoke. Bad visibility can also occur due to bad weather, but a camera based navigation system will definitely be more robust than one using GPS.

This thesis presents two ways to fuse the measurements from the camera and the IMU, both of them utilizing the Harris corner detector to find point correspondences between the camera image and an aerial photo. The systems are evaluated by simulated data mimicking both a low and a high accuracy IMU and a camera taking snapshots of the aerial photo. Results show that for the simulated camera images the implemented corner detector works fine and that the overall result is comparable to using a GPS.

Sammanfattning

Standardsättet för både flygande och markgående fordon att navigera är att använda ett tröghetsnavigeringssystem, innehållande en IMU som mäter acceleration och vinkelhastighet, tillsammans med GPS. IMU:n tillhandahållit högfrekventa mätningar av acceleration och vinkelhastighet som integreras till hastighet, position och attityd. Ett sådant system är omöjligt att störa, men lider av att de dödräknade storheterna hastighet, position och attityd, med tiden, kommer att driva iväg ifrån de sanna värdena. Tillsammans med GPS, som ger lågfrekventa mätningar av positionen, erhålls ett system med god dynamik och utan drift. Svagheten i ett
sådant system är dess integritet, då GPS enkelt kan störas med enkel och billig utrustning. För att lösa integritetsproblemet kan en kamera användas, antingen som stöd eller som ersättare till GPS. Kameran tar bilder som matchas gentemot en referens ex. en karta eller ett ortofoto. Det ger liknande mätningar som de GPS ger. Ett kamerabaserat system kan visserligen också störas genom att blockera synfältet för kameran med exempelvis rök. Dålig sikt kan också uppkomma på grund av dåligt väder eller dimma, men ett kamerabaserat system kommer definitivt att vara robustare än ett som använder GPS.

Acknowledgments

I would like to thank Hans Bohlin for giving me the chance to do my thesis at Saab Aerotech in Malmslätt and for all interesting discussions about as separate things as navigation and macro economy. The people at Saab for your welcoming attitude, which made my time in Malmslätt a very pleasant stay. My supervisor at LIU, David Törnqvist, deserves special thanks for answering all my silly questions and for all the feedback given on this thesis. Thomas Schön, my examiner, for his positive attitude and useful tips. I would also like to thank Markus Olgemar who did his thesis at Saab in parallel to me, for all our profitable discussion about our theses and the, perhaps not always so profitable, discussions about football. Finally I would like to thank Maria for all her support and love.

Linköping Januari 2009

Peter Rosander
# Contents

1 **Introduction**  1  
1.1 Problem Formulation  1  
1.2 Related Work  2  
1.3 Chapter Outline  2  

2 **Pre-Requierements**  5  
2.1 Direction Cosine Matrix - DCM  5  
2.2 Skew Symmetric Transformation  6  
2.3 Earth Model  7  
2.3.1 Gravity Model  7  
2.4 Coordinate Systems  8  
2.4.1 Earth Centered Intertial Frame - ECI  8  
2.4.2 Earth Centered Earth Fixed Frame - ECEF  8  
2.4.3 Geodetic Frame - GEO  9  
2.4.4 Body Frame - BODY  10  
2.4.5 Universal Transversal Mercator - UTM  12  
2.5 Inertial Measurement Unit - IMU  12  
2.5.1 Accelerometers  14  
2.5.2 Rate Gyros  14  
2.6 Estimation Theory  14  
2.6.1 Kalman Filter  15  
2.6.2 Extended Kalman Filter  16  
2.6.3 Particle Filter  17  

3 **Image Processing**  19  
3.1 Camera Model  19  
3.2 Feature Extraction  21  
3.2.1 Harris Corner Detector  21  
3.3 Image Matching  22  
3.3.1 Feature Based  22  
3.3.2 Outlier Rejection via RANSAC  24  
3.3.3 Normalized Cross Correlation  25
Chapter 1

Introduction

The standard way for both ground and aerial vehicles to navigate is to use an Inertial Navigation System, INS, containing an Inertial Measurement Unit, IMU, measuring the acceleration and angular rate, and a GPS measuring the position. The IMU provides high dynamic measurements of the acceleration and the angular rate, which the INS integrates to velocity, position and attitude, respectively. While being completely impossible to jam, the dead-reckoned estimates will drift away, i.e., the errors are unbounded. In conjunction with a GPS, providing low dynamic updates with bounded errors, a highly dynamic system without any drift is attained. The weakness of this system is its integrity, since the GPS is easily jammed with simple equipment, powered only by a small standard battery. When the GPS is jammed this system falls back into the behavior of the INS with unbounded errors. To counter this integrity problem a camera can be used as either a back up to the GPS or as its replacement. The camera provides images which are then matched versus a reference, e.g., a map or an aerial photo, to get similar estimates as the GPS would provide. The camera can of course also be jammed by blocking the view of the camera with smoke. Bad visibility can also occur due to bad weather, but a camera based navigation system will definitely be more robust than one using GPS.

1.1 Problem Formulation

Using a camera together with an IMU to navigate a flying autonomous vehicle raises a number of question to be dealt with. These questions can be divided into two separate groups, one concerning the matching between a video image and a reference and one concerning the fusion of information and overall system design. This thesis focuses on the latter, while the prior has been investigated by Olgemar (2008) which is a thesis work carried out almost in parallell to this thesis. The focus for this thesis as, as mentioned above, to investigate how a camera and an IMU can be used together to navigate an UAV. This objective has been split into the following sub targets.
• Investigate how different image processing techniques affects the sensor fusion.
• Investigate how the quality of the IMU affects the measurements from the camera.
• Investigate what kind of filter is best at handling the measurements from the camera.
• Investigate how the dynamics and trajectory affects the possibilities for a successful image processing.
• Create a simulation tool for creating trajectories and sensor data.
• Discuss the performance and robustness of a camera based system.

1.2 Related Work

Fusioning measurements from an IMU and a camera, to estimate position and attitude of an object with 6 DOF has been studied by e.g., Hol (2008), Mirzaei and Roumeliotis (2008) and Kelly and Sukhatme (2008). In common for all three are that they all use a pre-defined scene. In Hol (2008) a quite realistic scene is used and an accurate 3-D model of the scene serves as the reference against which the images from the camera was matched. The fusion was done in a particle filter and a feature detector was used to find correspondences. In both Mirzaei and Roumeliotis (2008) and Kelly and Sukhatme (2008) a similar setup was used, but instead of having a complete scene, pre-defined landmarks were used, simplifying the image processing. Both used a Kalman filter to fusion the measurements from the IMU and the camera. In Conte (2007) a camera and an IMU are used to navigate a small autonomous helicopter. The nature of the helicopter simplifies the problem to two dimensions and enables the use of magnetometers to gain estimates of the attitude, which enables to use normalized cross correlation for image matching, as described in Olgemar (2008). The information from the camera and IMU are dealt with in a marginalized particle filter, where the position is described using particles and the velocity and attitude are represented with a Kalman filter. This setup has proved to work in real life situations. This thesis is a continuation of Börjesson (2005) which focuses on ways to estimate the parameters needed for the transformation between a camera image and a map. There also exists alternatives to using a camera instead of a GPS. In Nordlund (2002) a downward pointing radar is used to provide measurements of the ground distance together with an height data base to get estimates of the position.

1.3 Chapter Outline

Chapter 2 gives a review of the background theory, needed to understand the concepts of navigation and sensor fusion. The properties of a camera and the techniques to extract information from an image are presented in Chapter 3. In
Chapter 4 the simulation environment is explained. In Chapter 5 and Chapter 6 two approaches to fusion the measurements from the IMU and Camera are presented together with results. Chapter 7 gives the overall conclusions and suggestions on further work.
Chapter 2

Pre-Requierements

For completeness of this thesis, a theoretical background to navigation and fusion is given.

2.1 Direction Cosine Matrix - DCM

A direction cosine matrix is a matrix $R$ fulfilling the following properties

- $R$ is real and orthogonal,
- $\text{eig } R = 1, e^{\pm i\theta}$,
- $\det R = 1$.

The set of those matrices are called $SO(3)$, Special Orthogonal with dimension 3, and although seeming quite abstract they have a very simple geometric property. Given two orthogonal right hand coordinate systems $A$ and $B$ which spans $\mathbb{R}^3$ and whose base vectors are $(x^A, y^A, z^A)^T$ and $(x^B, y^B, z^B)^T$ respectively, the transformation from $B$ to $A$ can be written as a direction cosine matrix or more specific as a rotation of $B$ around $x^A, y^A$ and $z^A$, (Britting, 1971).

Assuming that $x^A$ and $x^B$ coincide the relationship between $A$ and $B$ can be seen as a rotation around $x^A$ with the angle $\alpha_x$, see Figure 2.1, which mathematically can be written as

$$
\begin{pmatrix}
  x^B \\
  y^B \\
  z^B
\end{pmatrix} =
\begin{pmatrix}
  1 & 0 & 0 \\
  0 & \cos \alpha_x & \sin \alpha_x \\
  0 & -\sin \alpha_x & \cos \alpha_x
\end{pmatrix}
\begin{pmatrix}
  x^A \\
  y^A \\
  z^A
\end{pmatrix},
$$

(2.1)

where $R_A^B$ is a direction cosine matrix, and the notation $R_A^B$ helps clarifying which coordinate systems are affected.

For the general case, where there are no coinciding base vectors, the relationship
rotation is

\[
\begin{pmatrix}
    x^B \\
y^B \\
z^B
\end{pmatrix} =
\begin{pmatrix}
    1 & 0 & 0 \\
0 & \cos \alpha_x & \pm \sin \alpha_x \\
0 & -\sin \alpha_x & \cos \alpha_x
\end{pmatrix}
\begin{pmatrix}
    \cos \alpha_y & 0 & -\sin \alpha_y \\
0 & 1 & 0 \\
\pm \sin \alpha_y & 0 & \cos \alpha_y
\end{pmatrix}
\begin{pmatrix}
\cos \alpha_z & \pm \sin \alpha_z & 0 \\
0 & 0 & 1 \\
-\sin \alpha_z & 0 & \cos \alpha_z
\end{pmatrix}
\begin{pmatrix}
x^A \\
y^A \\
z^A
\end{pmatrix},
\] (2.2)

where again $R^B_A$ are a direction cosine matrix.

### 2.2 Skew Symmetric Transformation

A skew symmetric transformation of a vector, denoted $\langle \cdot \rangle$, is used to rewrite a cross product of two vectors

\[
u \times v = \langle u \rangle \times \begin{pmatrix}
u_x \\
v_y \\
v_z
\end{pmatrix},
\] (2.3)

as a matrix times a vector

\[
u \times v = \begin{pmatrix}
u_x \\
v_y \\
v_z
\end{pmatrix} = \begin{pmatrix}
0 & -u_z & u_y \\
u_z & 0 & -u_x \\
-u_y & u_x & 0
\end{pmatrix}
\begin{pmatrix}
v_x \\
v_y \\
v_z
\end{pmatrix}.
\] (2.4)

Except from this obvious use it has another interesting property, which of course also accounts for the cross product. Given a DCM, e.g., $R^B_A$, and a free vector describing the relative angular velocity of frame A relative to frame B denoted $\omega^B_{AB}$ the relationship

\[
\dot{R}^B_A = \langle \omega^B_{AB} \rangle R^B_A
\] (2.5)

holds, see Britting (1971). The relationship (2.5) is a very important result and can be used as the starting-point for the derivation of the equations presented in Section 6.1.
2.3 Earth Model

The earth, of almost everyone nowadays thought to be "round", can quite accurately be modelled as an ellipsoid, as illustrated in Figure 2.2, although there are plenty of more detailed models, see Britting (1971) and Bohlin (2000). However, in this thesis the earth is assumed to be spherical, with radius $R = R_m = R_p = 6.38 \cdot 10^6 \text{m}$. This assumption will simplify equations where the earth radius is included, but it will not affect the simulation result in any significant way. For real data such an assumption is too restricting.

2.3.1 Gravity Model

In general the gravity vector consists of mass attraction and centripetal force, (Bohlin, 2000), which yield a quite complicated gravity model. In this thesis it is assumed that the gravity vector points towards the centre of mass of the earth, which would be the case when only regarding the mass attraction, but it will still be called gravity. Under the assumption of a spherical earth, the gravity will be perpendicular to the surface of the earth as illustrated in Figure 2.3. The gravity vector is denoted $G$ and, in the Geodetic frame, is modelled as

$$G^g = \begin{pmatrix} 0 \\ 0 \\ 9.8 \end{pmatrix}.$$  \hspace{1cm} (2.6)
2.4 Coordinate Systems

This section specifies a number of coordinate systems needed to describe the movement of an aircraft and the concepts of the inertial measurement unit. For a more thorough review of the coordinate systems specified here Britting (1971) is a good starting-point.

2.4.1 Earth Centered Inertial Frame - ECI

The Earth Centered Inertial Frame is a coordinate system which is used to model the inertial space. It has its origin at the center of gravity of the earth and is fixed relative to distant stars.

2.4.2 Earth Centered Earth Fixed Frame - ECEF

This coordinate system’s origin coincide with that of ECI, but is fixed relative to the earth. This means that the difference between ECEF and ECI, see Figure 2.4, will be the angular velocity of the earth, $\omega_{ie}^e$ times time,

$$
\begin{pmatrix}
  x^e \\
  y^e \\
  z^e
\end{pmatrix}
= 
\begin{pmatrix}
  1 & 0 & 0 \\
  0 & \cos \omega_{ie}^e t & \sin \omega_{ie}^e t \\
  0 & -\sin \omega_{ie}^e t & \cos \omega_{ie}^e t
\end{pmatrix}
\begin{pmatrix}
  x^i \\
  y^i \\
  z^i
\end{pmatrix}.
$$

(2.7)

Earth Rate

The angular velocity of the earth relative to the inertial space is approximately

$$
\omega_{ie}^e = \begin{pmatrix}
  7.292 \cdot 10^{-5} \\
  0 \\
  0
\end{pmatrix} \text{ rad/s.}
$$

(2.8)
2.4 Coordinate Systems

2.4.3 Geodetic Frame - GEO

The geodetic frame is used to model the position of the vehicle relative to the surface of the earth, as illustrated by Figure 2.5. It has its origin at the center of the aircraft and its base vector \( x^g \) points north, \( y^g \) east parallel to the tangent plane of the surface of the earth and \( z^g \) points downwards orthogonal to the herementioned tangent plane. The rotation from ECEF to GEO can be written as

\[
\begin{pmatrix}
    x^g \\
    y^g \\
    z^g
\end{pmatrix}
= \begin{pmatrix}
    \cos \Phi & 0 & \sin \Phi \\
    0 & 1 & 0 \\
    -\sin \Phi & 0 & \cos \Phi
\end{pmatrix}
\begin{pmatrix}
    1 & 0 & 0 \\
    0 & \cos \lambda & \sin \lambda \\
    0 & -\sin \lambda & \cos \lambda
\end{pmatrix}
\begin{pmatrix}
    1 & 0 & 0 \\
    0 & 0 & 1 \\
    0 & -1 & 0
\end{pmatrix}
\begin{pmatrix}
    x^e \\
    y^e \\
    z^e
\end{pmatrix}. \quad (2.9)
\]

Here the longitude, \( \lambda \), and the latitude, \( \Phi \), together with the altitude, \( h \), defines the geodetic position,

\[
r^g = \begin{pmatrix}
    \Phi \\
    \lambda \\
    h
\end{pmatrix}, \quad (2.10)
\]

which is often used to describe the position of an aircraft.

Craft Rate

When an aircraft moves across the surface of the earth the movement can be considered as an angular velocity, since the earth is spherical, or at least close to. The craft rate is defined as the angular velocity of GEO relative to ECEF, yielding
the relationship
\[
\omega_{eg}^g = \begin{pmatrix}
\dot{\lambda} \cos \Phi \\
- \dot{\Phi} \\
- \dot{\lambda} \sin \Phi
\end{pmatrix} = \begin{pmatrix}
\frac{v^g_y}{R+h} \\
\frac{v^g_z}{R+h} \\
\frac{v^g_y}{R+h} \tan \Phi
\end{pmatrix},
\tag{2.11}
\]
where Figure 2.5 can be used to obtain the last equality.

### 2.4.4 Body Frame - BODY

The Body frame origin coincides with that of GEO but $x^b$ points towards the front of the aircraft, $y^b$ towards the right wing and $z^b$ is parallel to the vertical axis of the aircraft, see Figure 2.6. The relation between GEO and BODY is
\[
\begin{pmatrix}
x_b^b \\
y_b^b \\
z_b^b
\end{pmatrix} = \begin{pmatrix}
1 & 0 & 0 \\
0 & \cos \phi & - \sin \phi \\
0 & \sin \phi & \cos \phi
\end{pmatrix} \begin{pmatrix}
\cos \theta & 0 & - \sin \theta \\
0 & 1 & 0 \\
\sin \theta & 0 & \cos \theta
\end{pmatrix} \begin{pmatrix}
\cos \psi & \sin \psi & 0 \\
- \sin \psi & \cos \psi & 0 \\
0 & 0 & 1
\end{pmatrix} \begin{pmatrix}
x^g \\
y^g \\
z^g
\end{pmatrix},
\tag{2.12}
\]
where $\phi$, $\theta$ and $\psi$ are called roll angle, pitch angle and heading/yaw, respectively. These angles are also commonly referred to as the Euler angles and are further elucidated in Figure 2.7. Given the direction cosine matrix $R_b^g$ the Euler angles can be recovered via
\[
\phi = \arctan \frac{R^g_b(2,3)}{R^g_b(3,3)}, \quad -\pi \leq \phi < \pi,
\tag{2.13a}
\]
\[
\theta = \arcsin -R^g_b(1,3), \quad \frac{-\pi}{2} \leq \theta < \frac{\pi}{2},
\tag{2.13b}
\]
\[
\psi = \arctan \frac{R^g_b(1,2)}{R^g_b(1,1)}, \quad -\pi \leq \psi < \pi.
\tag{2.13c}
\]
Figure 2.6. Definition of the BODY frame (Bohlin, 2000).

Figure 2.7. Definition of the Euler angles (Bohlin, 2000).
2.4.5 Universal Transversal Mercator - UTM

This coordinate system, described in among others Bugayevskiy and Snyder (1995), is a grid-based frame where the earth is divided into 60 zones, each sex degrees of longitude in width. Furthermore each zone is divided into 20 latitude bands of eight degrees height as illustrated by Figure 2.8. The UTM coordinate system is used to transform a geodetic position to a position on the map and the coordinates in UTM are given in meters. For a spherical earth it is the mantel piece of a cylinder which encloses the earth spheroid as shown in Figure 2.9. For longitudinal positions, $\lambda$, distant to the meridian $\lambda_0$ the true north and the grid/map north will deviate as illustrated in Figure 2.10. For a fixed $\delta \lambda = \lambda - \lambda_0$ the deviation increases with the absolute value of the latitude. In this thesis it is, however, assumed that the flight takes place close to the zero median, $\lambda_0$, and not too close to the poles, so that these effects can be ignored, yielding the transformation from GEO to UTM

\[\text{North} = \Phi \cdot R, \quad (2.14a)\]
\[\text{East} = \lambda \cdot R \cdot \cos \Phi, \quad (2.14b)\]

where $R$ is the earth radius and higher order terms are ignored.

2.5 Inertial Measurement Unit - IMU

The inertial measurement unit is one of the sensor used to navigate and it consists of three orthogonally placed accelerometers and rate gyros, which measures the acceleration and angular velocity relative to the inertial space. For the studied application it is assumed that the IMU is attached to the origo of the BODY frame and having sensors in the $x^b$, $y^b$ and $z^b$ directions.
Figure 2.9. Projection of the spherical earth on a flat surface (Bohlin, 2000).

Figure 2.10. True north and grid north (Bohlin, 2000).


2.5.1 Accelerometers

The accelerometers measure the force inferred on the aircraft and consequently measure both acceleration, gravity as well as the Coriolis force, yielding the model

\[ f^b = R^b_g (a^g + \omega^g_{eg} \times v^g_e + 2R^g_e \omega^e_{ie} \times v^g_e - G^g), \]  

(2.15)

where \( a^g \) is the acceleration of the aircraft, \( G^g \) models the gravity and the middle terms are the Coriolis force, (Bohlin, 2000). Note that low accuracy accelerometers might not be able to distinguish the influence of the earth rate.

2.5.2 Rate Gyros

The rate gyros measure the angular velocity of the aircraft relative to the inertial frame and similar to the accelerometers the rate gyros are affected by the rotation of the earth, yielding

\[ \omega^b_{ib} = \omega^b_{ie} + \omega^b_{eg} + \omega^b_{gb} = R^b_g R^g_e \omega^e_{ie} + R^b_g \omega^g_{eg} + \omega^b_{gb}. \]  

(2.16)

Just as for the accelerometers this model can be exaggerated thoroughly for low accuracy rate gyros since they cannot distinguish the earth rate.

2.6 Estimation Theory

Estimation theory is the theory of deducing as much information as possible about an unknown quantity, e.g., a state variable or parameter. The deduction is based on the outcome (observation) of a variable quantity which is related to the unknown variable. In general the unknown variable is denoted \( x_t \in \mathbb{R}^{n_x} \) while the observations are called \( y_t \in \mathbb{R}^{n_y} \), where \( n_x \) and \( n_y \) denotes the number of unknowns and the number of observations at a given time.

In taking the Bayesian approach to estimation, described in e.g., Gustafsson (2000), one is considering everything as stochastic, hence both \( x_t \) and \( y_t \) are described by their respective probability distributions \( p(x_t) \) and \( p(y_t) \) respectively. Assuming \( x_t \) is a state, see Torkel and Ljung (2003), the propagation in time of any physical system can be written as

\[ p(x_{t+1} | x_t), \]

\[ p(y_t | x_t), \]  

(2.17)

where \( p(x_{t+1} | x_t) \) is the probability distribution describing the state transitions and \( p(y_t | x_t) \) describes how a certain state transitions to the observation \( y_t \). Bayes’ formula \(^1\) gives us a way to calculate the conditional probability density function

\[^1 p(a|b) = \frac{p(b|a)p(a)}{p(b)}\]
for the state $x_t$ given all the measurement $y_{1:t}$, $p(x_t|y_{1:t})$ according to

$$
p(x_t|y_{1:t}) = \frac{p(y_t|x_t, y_{1:t-1})p(x_t|y_{1:t-1})}{p(y_t|y_{1:t-1})} = \frac{p(y_t|x_t)p(x_t|y_{1:t-1})}{p(y_t|y_{1:t-1})}.
$$

(2.18)

The last equality follows from the Markov property (Yates and Goodman, 2005). The expression (2.18) is called the measurement update in the Bayesian recursion. Using marginalization we get the time update

$$
p(x_{t+1}|y_{1:t}) = \int_{\mathbb{R}^n_x} p(x_{t+1}, x_t|y_{1:t}) dx_t
$$

$$
= \int_{\mathbb{R}^n_x} p(x_{t+1}|x_t, y_{1:t})p(x_t|y_{1:t}) dx_t = \int_{\mathbb{R}^n_x} p(x_{t+1}|x_t)p(x_t|y_{1:t}) dx_t.
$$

(2.19)

### 2.6.1 Kalman Filter

In general (2.19) and (2.18) do not have an analytical solution for the model (2.17), but for the dynamic model

$$
x_{t+1} = F_t x_t + B_t u_t + w_t, \ w_t \in \mathcal{N}(0, Q_t)
$$

$$
y_t = H_t x_t + e_t, \ e_t \in \mathcal{N}(0, R_t)
$$

$$
\text{Cov}[w_t, e_t] = 0, \ x_0 \in \mathcal{N}(0, P_0)
$$

(2.20)

where $u_t$ is a known input, there do exist an analytical solution, called the Kalman filter. The connection between (2.17) and (2.20) is that

$$
p(x_{t+1}|x_t) = p_{w_t}(x_{t+1} - B_t u_t - F_t x_t |x_t) = \mathcal{N}(x_{t+1}; F_t x_t + B_t u_t, Q_t),
$$

(2.21)

and

$$
p(y_t|x_t) = p_{e_t}(y_t - H_t x_t | x_t) = \mathcal{N}(y_t; H_t x_t, R_t).
$$

(2.22)

The exact derivation of the Kalman filter will be omitted, and only the recursion is presented.

Given the linear state space model (2.20) and assuming that $Q_t$, $R_t$, $P_0$ are all positive definite, $x_{t+1}$ and $x_t$ conditioned on $y_{1:t}$ are Gaussian distributed for any time $t \leq 0$, i.e.,

$$
p(x_t|y_{1:t}) = \mathcal{N}(x_t; \hat{x}_{t|t}, P_{t|t})
$$

(2.23a)

$$
p(x_{t+1}|y_{1:t}) = \mathcal{N}(x_{t+1}; \hat{x}_{t+1|t}, P_{t+1|t})
$$

(2.23b)
where the mean and the covariance, given initial values $\hat{x}_{0|-1} = 0$ and $P_{0|-1} = P_0$, propagate according to

**Time Update**

\[
\begin{aligned}
\hat{x}_{t+1|t} &= F_t x_t + B_t u_t \\
P_{t+1|t} &= F_t P_{t|t} F^T_t + Q_t
\end{aligned}
\]  

(2.24)

**Measurement Update**

\[
\begin{aligned}
\hat{x}_{t|t} &= \hat{x}_{t|t-1} + P_{t|t-1} H^T_t (H_t P_{t|t-1} H^T_t + R_t)^{-1} (y_t - H_t \hat{x}_{t|t-1}) \\
P_{t|t} &= P_{t|t-1} - P_{t|t-1} H^T_t (H_t P_{t|t-1} H^T_t + R_t)^{-1} H_t P_{t|t-1}
\end{aligned}
\]  

(2.25)

The **Time Update** is the solution of (2.19) and **Measurement Update** the solution of (2.18).

### 2.6.2 Extended Kalman Filter

There are not many systems which can be modelled as (2.20), since the linear dynamics often are a limitation. Allowing for non-linear dynamics, the state space model can be written as

\[
\begin{aligned}
x_{t+1} &= f(x_t, u_t) + w_t, \quad w_t \in \mathcal{N}(0, Q_t) \\
y_t &= h(x_t) + e_t, \quad e_t \in \mathcal{N}(0, R_t) \\
\text{Cov}[w_t, e_t] &= 0, \quad x_0 \in \mathcal{N}(0, P_0).
\end{aligned}
\]  

(2.26)

A local approximation of a linear system can then be achieved by Taylor-expansion around the latest estimate from the filter,

\[
\begin{aligned}
f(x_t) &\approx f(\hat{x}_{t|t}, u_t) + F_t (x_t - \hat{x}_{t|t}), \\
h(x_t) &\approx h(\hat{x}_{t|t-1}) + H_t (x_t - \hat{x}_{t|t-1})
\end{aligned}
\]  

(2.27)

where

\[
\begin{aligned}
F_t &= \frac{\partial f(x_t, u_t)}{\partial x_t} \bigg|_{x_t=\hat{x}_{t|t}} \\
H_t &= \frac{\partial h(x_t)}{\partial x_t} \bigg|_{x_t=\hat{x}_{t|t-1}}
\end{aligned}
\]  

(2.28)
This makes it possible to use the attractive equations of the Kalman filter, where the non-linear model is approximated with a linear approximation. Given the same prerequisites as for the Kalman filter, apart from the linear dynamics, the recursion is instead

### Time Update

\[
\hat{x}_{t+1|t} = f(\hat{x}_t, u_t) \tag{2.29a}
\]

\[
P_{t+1|t} = F_t P_{t|t} F_T + Q_t \tag{2.29b}
\]

### Measurement Update

\[
\hat{x}_{t|t} = \hat{x}_{t|t-1} + P_{t|t-1} H^T_t (H_t P_{t|t-1} H^T_t + R_t)^{-1} (y_t - h(\hat{x}_{t|t-1})) \tag{2.30a}
\]

\[
P_{t|t} = P_{t|t-1} - P_{t|t-1} H^T_t (H_t P_{t|t-1} H^T_t + R_t)^{-1} H_t P_{t|t-1} \tag{2.30b}
\]

### 2.6.3 Particle Filter

For the case when \( p(x_t|y_{1:t}) \) and \( p(x_{t+1}|y_{1:t}) \) are not Gaussian there basically do not exist any counterpart to the Kalman filter, for which only the mean and covariance need to be propagated, but rather the whole probability distribution need to be stored. The particle filter, first introduced by Gordon et al. (1993), solves this problem by approximating \( p(x_t|y_{1:t}) \) and \( p(x_{t+1}|y_{1:t}) \) with \( N_p \) samples from the sought distribution

\[
x_t^{(i)} \sim p(x_t|y_{1:t}), \quad i = 1, \ldots, N_p, \tag{2.31}
\]

where \( x_t^{(i)} \) are the samples, often referred to as particles. Each particle also has a weight \( q_t^{(i)} \) telling how probable that particular particle is to fit the correct distribution, and since the weights describe the probability they fullfill the constraints

\[
\sum_{i=1}^{N_p} q_t^{(i)} = 1, \quad q_t^{(i)} \geq 0, \quad \forall i. \tag{2.32}
\]

Given these weights and particles, an approximative distribution can be formed as

\[
\hat{p}(x_t|y_{1:t}) = \sum_{i=1}^{N_p} q_t^{(i)} \delta(x_t - x_t^{(i)}), \tag{2.33}
\]
where $\delta(\cdot)$ is the Dirac distribution (Kreyszig, 1989). There are many variants of
the particle filter and all of them gives different expressions for the weights $q_{t|t}^{(i)}$, but in this thesis only the implemented recursion will be presented. Given initial
values $x_0^{(i)} \sim p(x_0)$ and $q_0^{(i)} = \frac{p(y_0|x_0^{(i)})}{\sum_{j=1}^{N_p} p(y_0|x_0^{(j)})}$ the recursion can be written as

**Time Update**

$$
x_{t+1}^{(i)} = p(x_{t+1}|x_{t}^{(i)}), \quad i = 1, \ldots, N_p,
$$

(2.34)

**Measurement Update**

$$
q_{t|t}^{(i)} = \frac{q_{t|t}^{(i)} p(y_t|x_t^{(i)})}{\sum_{j=1}^{N_p} q_{t|t}^{(j)} p(y_t|x_t^{(j)})}, \quad i = 1, \ldots, N_p,
$$

(2.35a)

$$
q_{t+1|t}^{(i)} = q_{t|t}^{(i)}.
$$

(2.35b)

**Resampling**

After a while almost all particle will have a weight close to zero, known as depletion. To counter this, a new set of particles are drawn from the distribution $p(x_t|y_{1:t})$ in each time step. The probability of resampling a certain particle is equal to its
weight, which will yield that probable particles are multiplied and less probable
particles are thrown away.
Chapter 3

Image Processing

A camera is a very powerful sensor when it comes to the information available, but there are several issues to be solved before that information can be utilized. This chapter starts by describing the properties of the camera and then discusses techniques to extract the information in an image. Important to mention are that the implementation of the presented theory are based on Kovesi (2000) and Olgemar (2008). Some minor adjustments and tuning of parameters have been done for better performance.

3.1 Camera Model

The properties of a camera can be described with two coordinate systems, (Börjesson, 2005), the camera frame $c$ and the image frame $i$, where the former is aligned as in Figure 3.1 and the latter is a 2-dimensional frame which spans the image taken by the camera, see Figure 3.2. The relationship between the two frames is

$$x^i = -Fx^c_z,$$  \hspace{1cm} (3.1a)

$$y^i = -Fy^c_z,$$  \hspace{1cm} (3.1b)

Figure 3.1. Definition of the camera frame
**Figure 3.2.** Definition of the image frame

**Figure 3.3.** The pinhole camera model in the yz-plane

which can be seen by applying similar triangles in Figure 3.3, where the yz-plane is illustrated and $x^i$ follows analogously. Important to note is that the camera model (3.1) is a simplification which is only valid for the simulated data used in this thesis.

For the studied application it is assumed that the origo of the camera frame will coincide with that of the body frame, and that the camera is rigidly attached on the aircraft, which yields that $z^c$ will coincide with $z^b$, $x^c$ with $y^b$ and $-y^c$ with $x^b$ as illustrated by Figure 3.4. The transformation from the body to the camera frame can hence be written as

$$R^c_b = \begin{pmatrix} \cos \frac{\pi}{2} & \sin \frac{\pi}{2} & 0 \\ -\sin \frac{\pi}{2} & \cos \frac{\pi}{2} & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (3.2)$$

So the projection of a point in the world, $n^g = (n^g_x, n^g_y, n^g_z)^T$, onto the image plane
is given by
\begin{align}
  n^i_x &= -F \frac{n^c_x}{n^c_z}, \quad (3.3a) \\
  n^i_y &= -F \frac{n^c_y}{n^c_z}, \quad (3.3b)
\end{align}
where
\[ n^c = \begin{pmatrix} n^c_x \\ n^c_y \\ n^c_z \end{pmatrix} = R^c_b R^g_y (r^g - n^g). \quad (3.4) \]
Here \( r^g \) is the position of the aircraft and \( r^g - n^g \) translates the feature point \( n^g \) so that it ends up in the origo of the camera frame.

### 3.2 Feature Extraction

As mentioned before an image from a camera is very rich in terms of information, but it has to be processed in one way or another to extract this information. This section focuses on image comparison, for which a feature extractor has been used. A feature extractor detects either regional changes in an image or regions that are homogenous in some way. The concept can be explained by an image with a white background and a black rectangle. For this image either the change from black to white or the black and white regions will be detected.

#### 3.2.1 Harris Corner Detector

The Harris corner detector uses the fact that at a corner, e.g., change from black to white, the image intensity in different directions will change largely. To detect
such corners in an image, $I(x^i, y^i)$, the image gradient $(I_{x^i}, I_{y^i})$ is calculated, see Ma et al. (2003). These gradients are then smoothed by applying a gaussian filter, and each pixel $(x^i, y^i)$ are assigned a value accordingly

$$C(x^i, y^i) = \frac{I_{x^i}^2 I_{y^i}^2 - I_{x^i} I_{y^i}}{I_{x^i} + I_{y^i} + \epsilon}. \quad (3.5)$$

If the $C$-value is larger than a pre-defined threshold, $(x^i, y^i)$ is marked as a corner, but to avoid having a lot of pixels close to each other where all are detected as a feature nonmaximal suppression is performed, (Kovesi, 2000). This guarantees that the detected features will be separated by, at least, a pre-defined distance. To illustrate the properties of the Harris corner detector see Figure 3.5, where it has been applied to a typical camera image.

3.3 Image Matching

In the studied application the aircraft is flying at an altitude of 1000 meters and the simulated camera images will have a footprint of 150 by 150 meters with a resolution of 0.3 meters and the reference will be an aerial photo with a resolution of 1 meters. So simply put the camera image will be a magnification of some area of the reference and the daunting challenge is to decide where. Available to facilitate this decision are estimates of the attitude and position of the aircraft from the filter, see Chapter 5 and Chapter 6 for details.

3.3.1 Feature Based

To detect where the camera image fits in to the reference image the reference is transformed to the same plane as the camera image, using the estimated position
3.3 Image Matching

**Figure 3.6.** Detected corners in the camera and reference image before correspondence matching.

**Figure 3.7.** Corners having a putative correspondent
and attitude via (3.3). This transformation will most likely yield a result similar to the one shown in Figure 3.6, where the left image is the camera image and the right is the transformed reference with a position error of five meters in the eastern direction. Correspondencies are found by cutting out a neighbourhood of five pixels around each detected corner in both images. Corners within a distance of 15 pixels are matched together by crosscorrelating these areas. Features lacking a correspondence are thrown away, yielding a result illustrated in Figure 3.7. Although some outlier rejection has been performed there are still some matches that are incorrect, which is elucidated in Figure 3.8, where non horizontal lines clearly are outliers needed to be rejected.

### 3.3.2 Outlier Rejection via RANSAC

To detect the outliers in the set of putative matches an algorithm called RANSAC, RAndom SAmple Consensus, has been used, (Hartley and Zisserman, 2003). The algorithm works in the way that a random set of four features are chosen, and then $R^b_g$ and $r^g$ are calculated using the Direct Linear Transformation, (Hartley and Zisserman, 2003). Then for all detected features, the estimated position in the camera image are calculated and compared to their true position. The calculated $R^b_g$ and $r^g$ goodness are evaluated by these distances. Then a new set of four random features are chosen and the procedure starts all over again until a pre defined number trials have been performed and then the $R^g_b$ and $r^g$ which fitted best is chosen. For this choice the features fitting good are marked as inliers. The result from the RANSAC algorithm can be illustrated by Figure 3.9, where there are only a small number of outliers left.
3.3 Image Matching

3.3.3 Normalized Cross Correlation

For the case where the attitude, the Euler angles, are known or observed with, e.g., a magnetometer for a helicopter (only having a yaw angle) or if the rate gyros are of explicit accuracy, there exists an alternative to the method presented above. Instead of finding point correspondencies, which provides observations of both position and attitude, the camera image is correlated with the transformed reference image, where a peak will occur for the position with the best fit as illustrated in Figure 3.10. The weakness of this approach is, as mentioned, that it only provides estimates of the position given a certain attitude, while it will most likely perform significantly better for real data than the feature based matching. In addition, a correlation image contains a lot of information and setting negative values to zero and normalizing, the resulting correlation, see Figure 3.11, can be seen as a likelihood function $p(y_t|x_t)$, which would be ideal to use in particle filter.

Figure 3.9. Correct matches as detected by the RANSAC algorithm.
Figure 3.10. Correlation in the Image frame.

Figure 3.11. Likelihood in the Image frame.
Chapter 4

Simulation

This chapter gives a review of how the data, used in the consecutive chapters, have been created. The data simulated are the true trajectory, the outputs from the inertial measurement unit and the inertial navigation system, described in Section 6.1.

4.1 Aircraft Model

The aircraft model used is quite simplified and assumes that the yaw rate, $\dot{\psi}$, depends on the roll angle, $\phi$, which is a simplification since in reality the pitch angle has to be separate from zero, $\theta \neq 0$, for this relation to hold. However, in this thesis it is assumed that the aircraft flies on constant height and hence $\theta = 0$. This assumption yields that the aircraft only needs to be controlled in the horizontal plane. Furthermore, the velocity is assumed constant and to be strictly in the $v^b_x$-direction, yielding the model

$$v^b(t) = \begin{pmatrix} 40 \\ 0 \\ 0 \end{pmatrix} \text{ m/s,} \quad (4.1a)$$

$$\phi(s) = \frac{1}{s^2 + 2s + 2} u, \quad (4.1b)$$

$$\psi(s) = \frac{0.5}{s} \phi, \quad (4.1c)$$

$$\theta(s) = 0, \quad (4.1d)$$

where $u$ is the control signal and can be seen as ailerons command. The transfer function from the control signal, $u$, to the roll angle, $\phi$, has been chosen to get a fairly volatile behavior, as elucidated in Figure 4.1, where the step response is presented. The Euler angles can then via (2.12) be written as the rotation matrix $R^g_b$ and the velocity in the geodetic frame will be

$$v^g = R^g_b v^b. \quad (4.2)$$
Simulation

The geodetic position is dead reckoned as

\[ R_{g, e, t+1}^g = R_{g, e, t}^g + T_s < \omega_{eg}^g > R_{e, t}^g \]  \hspace{1cm} (4.3)

where \( T_s \) are the sample time and \( \omega_{eg}^g \) depends on \( v^g \) and is given by (2.11). The problem with dead reckoning of a rotation matrix is that it will lose the properties described in Section 2.1. These properties can be recovered by orthogonalization, (Britting, 1971),

\[ R_\perp = R_\perp + \frac{1}{2} (I - R_\perp R_\perp^T) R_\perp, \]  \hspace{1cm} (4.4)

where \( R_\perp \) is the non-orthogonalized matrix and \( R_\perp \) is the orthogonalized matrix.

4.2 IMU Model

From the simulation, the attitude, angular velocity, velocity and postion of the aircraft are known. Given those quantities the output from the accelerometers and the rate gyros can be calculated. As described in Section 2.5 the output from the IMU is

\[ \langle \omega^b \rangle = \langle \omega^b_{ie} \rangle + \langle \omega^b_{eg} \rangle + \langle \omega^b_{gb} \rangle \]  \hspace{1cm} (4.5a)

\[ f^b = R_g^b R_e^g \langle \omega^c_{ie} \rangle + R_g^b \langle \omega^g_{eg} \rangle + \langle \omega^b_{gb} \rangle, \]  \hspace{1cm} (4.5b)

So to get the output from the IMU both \( R_g^b R_e^g \langle \omega^c_{ie} \rangle + R_g^b \langle \omega^g_{eg} \rangle + \langle \omega^b_{gb} \rangle \) and \( R_g^b (a^g + (\omega^g_{eg} + 2R_e^g \omega^c_{ie}) \times v^g_e - G^g) \) have to be calculated, which is done by following these steps

- \( R_g^b \) depends on the Euler angles, see (2.12)
- \( R_e^g \) is known from the simulation of the aircraft
- \( \omega^c_{ie} \) is the rotation of the earth relative to the inertial frame and is given by (2.8)
- \( \omega^g_{eg} \) depends on the position and velocity of the aircraft and is given by (2.11)
- \( \omega^b_{gb} \) depends on the attitude and its derivatives and is calculated via

\[ \omega^b_{gb,x} = \dot{\phi} - \dot{\psi} \sin \theta \]  \hspace{1cm} (4.6a)

\[ \omega^b_{gb,y} = \dot{\theta} \cos \phi + \dot{\psi} \cos \theta \sin \phi \]  \hspace{1cm} (4.6b)

\[ \omega^b_{gb,z} = -\dot{\theta} \sin \phi + \dot{\psi} \cos \theta \cos \phi \]  \hspace{1cm} (4.6c)

- \( a^g \) can be calculated by numerical derivation of \( v^g \)
- the gravity vector \( G^g = (0, 0, -9.8)^T \).
4.3 Simulation Results

4.2.1 Disturbance Model

Apart from numerical errors in the simulated data some characteristical behavior of real life IMU-data is added such as scale factor, $s$, bias, $\Delta$, and white noise, $w$,

$$f^b = (1 + sf^b)f^b + \Delta f^b + w_t, \quad w_t \in \mathcal{N}(0, Q_{f^b}),$$  \hspace{1cm} (4.7a)

$$\omega^b_{ib} = (1 + s_{\omega^b_{ib}})\omega^b_{ib} + \Delta \omega^b_{ib} + w_t, \quad w_t \in \mathcal{N}(0, Q_{\omega^b_{ib}}).$$  \hspace{1cm} (4.7b)

4.3 Simulation Results

The simulated trajectory is illustrated in Figure 4.2 and Figure 4.3, where the position is shown, and in Figure 4.4 showing the attitude of the aircraft. The calculated IMU data are disturbed as described in (4.7) and dead-reckoned to estimates of the position, velocity and attitude. How the dead reckoning is done is given in detail in Section 6.1. Two different types of noise levels have been tested, where the first represent a high accurate IMU and the second an IMU of poor accuracy. The simulated data sets are five minutes and the characteristics of the two noise levels will be shown with a couple of figures. In common for both data sets are that IMU is assumed to supply measurements at a rate of 100 Hz.

4.3.1 High Accuracy IMU

This data set simulates a IMU of medium to high accuracy and the following noise levels have been used

<table>
<thead>
<tr>
<th>Noise Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_{f^b}$</td>
<td>$10^{-6}$ m/s$^2$</td>
</tr>
<tr>
<td>$\Delta f^b$</td>
<td>$0.5 \cdot 10^{-3}$ m/s$^2$</td>
</tr>
<tr>
<td>$Q_{f^b}$</td>
<td>$0.5 \cdot 10^{-3}$ m/s$^2$</td>
</tr>
<tr>
<td>$s_{\omega^b_{ib}}$</td>
<td>$10^{-7}$ rad/s</td>
</tr>
<tr>
<td>$\Delta \omega^b_{ib}$</td>
<td>$\frac{\pi}{180 \cdot 3600}$ rad/s</td>
</tr>
<tr>
<td>$Q_{\omega^b_{ib}}$</td>
<td>$\frac{\pi}{180 \cdot 3600}$ rad/s</td>
</tr>
</tbody>
</table>
Figure 4.2. The true simulated trajectory. The white crosses marks the pre-defined waypoints.

Figure 4.3. True simulated position.
Figure 4.4. True simulated attitude.
Figure 4.5. Trajectory for a high accuracy IMU. The true trajectory is solid and the dead reckoned is dashed and the white crosses marks the pre-defined waypoints.

The somewhat strange looking bias term for the rate gyros are chosen to get a drift of one degree per hour. This type of IMU makes it possible to dead reckon the position fairly accurate as shown in Figure 4.5 and in Figure 4.6. The dead-reckoned attitude is shown in Figure 4.7.

4.3.2 Low Accuracy IMU

To mimic an IMU of low quality the simulated noise levels are

| $s_{fb}$ | $10^{-3}$m/s² |
| $\Delta f_b$ | $10^{-2}$m/s² |
| $Q_{fb}$ | 1m/s² |
| $s_{\omega_{ib}}$ | $10^{-3}$rad/s |
| $\Delta \omega_{ib}$ | $10^{-3}$rad/s |
| $Q_{\omega_{ib}}$ | $10^{-2}$rad/s |

In contrast to the other IMU, this type of IMU is only able to dead reckon accurately for a couple of seconds which is shown in Figure 4.8, where the dead reckoned position leaves the map after approximately 150 seconds. The dead reckoned position is elucidated in Figure 4.9. The dead reckoned attitude is shown in Figure 4.10.
Figure 4.6. Dead-reckoned position for a high accuracy IMU.

Figure 4.7. Dead-reckoned attitude for a high accuracy IMU.
Figure 4.8. Trajectory for a low accuracy IMU. The true trajectory is solid and the dead reckoned is dashed and the white crosses marks the pre-defined waypoints.

Figure 4.9. Dead-reckoned position for a low accuracy IMU.
Figure 4.10. Dead-reckoned attitude for a low accuracy IMU.
Chapter 5

Hybrid Camera and IMU Navigation

For this section one fast sensor, the IMU, providing high frequency noisy updates of the acceleration and angular velocity, and one slow sensor, the camera, providing low frequency updates from which the attitude and position can be exerted. Basically the perfect setup for sensor fusion. The setup is similar to the one in Chapter 6, but they differ in some aspects, e.g., the filter suggested in this section will most likely perform better for low accuracy IMU, while it will handle flights across great distances poorly.

5.1 A Flat Earth Assumption

It will be assumed that, at least, locally the earth spheroid can be approximated by a flat surface. This assumption is of course only acceptable for short distances. The coordinate system will be called MAP, because of its similarity with the flatness of a map, and the position \( r^m = (x^m, y^m, z^m)^T \) is given in meters, c.f., (2.10). The position will also be referred to as north, east and down, \( r^m = (N, E, D)^T \). A flat assumption also simplifies the relation between the time derivative of the position and the velocity, c.f., (2.8), to simply

\[
\dot{r}^m = v^m. \tag{5.1}
\]

The relationship between the BODY frame and the MAP frame, see Figure 5.1, can, analogously to the transformation between the BODY and the GEO frame, be written as three rotations

\[
\begin{pmatrix}
  x^b \\
  y^b \\
  z^b
\end{pmatrix} = \begin{pmatrix}
  1 & 0 & 0 \\
  0 & \cos \phi & \sin \phi \\
  0 & -\sin \phi & \cos \phi
\end{pmatrix} \begin{pmatrix}
  \cos \theta & 0 & -\sin \theta \\
  0 & 1 & 0 \\
  -\sin \theta & 0 & \cos \theta
\end{pmatrix} \begin{pmatrix}
  \cos \psi & \sin \psi & 0 \\
  -\sin \psi & \cos \psi & 0 \\
  0 & 0 & 1
\end{pmatrix} \begin{pmatrix}
  x^m \\
  y^m \\
  z^m
\end{pmatrix}, \tag{5.2}
\]

where the same notation of the angles has been used as for the transformation from the body frame to the geographic frame, see Section 2.4.4. For that matter the Euler angles, both here and in (2.12), represent the attitude of the body relative to the earth, only that the earth is modelled differently here.
5.2 Dynamic Model

The simplification of a flat earth and the lack of possibilities to distinguish the earth rate and coriolis force makes the dynamic model presented here quite intuitive. Since the data used are simulated using (4.1) it would be natural to use the same model for the filter, but for generality all information of the dynamics of the aircraft is assumed unknown. This yields the model

\[
\begin{align*}
\dot{r}^m &= v^m, \quad (5.3a) \\
\dot{v}^m &= a^m, \quad (5.3b) \\
\dot{a}^m &= w_a, \quad (5.3c) \\
\dot{\Theta}_m^b &= f(\omega_{mb}^b, \Theta_m^b), \quad (5.3d) \\
\dot{\omega}_{mb}^b &= w_\omega, \quad (5.3e)
\end{align*}
\]

where \( w \in \mathcal{N}(0, Q) \) and \( \Theta \) are the Euler angles describing the attitude of the aircraft. The dynamics of the Euler angles, \( f(\omega_{mb}^b, \Theta) \), are

\[
\begin{align*}
\dot{\phi} &= \omega_{mb,x}^b + \omega_{mb,y}^b \sin \phi \tan \theta + \omega_{mb,z}^b \cos \phi \tan \theta, \quad (5.4a) \\
\dot{\theta} &= \omega_{mb,y}^b \cos \phi - \omega_{mb,x}^b \sin \phi, \quad (5.4b) \\
\dot{\psi} &= \omega_{mb,y}^b \sin \phi \cos \theta + \omega_{mb,z}^b \cos \phi \cos \theta. \quad (5.4c)
\end{align*}
\]

To model the attitude of the aircraft with Euler angles has two disadvantages, non-linear dynamics and being undefined when \( \theta = \frac{\pi}{2} \), while the advantages are that, compared to other parametrizations, they are intuitive and requires fewer parameters, which in turns makes the state space vector smaller.
5.3 Filtering Results

The measurements from the IMU and the camera are modelled accordingly

\[ y_{\omega,t} = \omega_{mb,t}^b + \epsilon_{\omega,t}, \]  
\[ y_{a,t} = R_{mb,t}^b (a_t^m - g_m^m) + e_{a,t}, \]  
\[ y_{c,t} = \frac{1}{n_c^c} \left( -F_{n_c^c}^x \right) + e_{c,t}, \]

(5.5a)  
(5.5b)  
(5.5c)

where \( \epsilon_t \in \mathcal{N}(0, R_t) \) and \( n \) is a detected feature and its coordinates in the camera frame are given by

\[ n_c^c = R_{cb}^c R_{mb,m}^b (r_m^m - n_m^m) \]  
(5.6)

For further explanation of the properties of the camera see Section 3.1.

5.2.1 Discretization of the Model

For the time continuous model, (5.3), the translational dynamics can be discretized exactly, (Törnqvist, 2006), yielding the discrete model

\[ \begin{pmatrix} r_{t+1}^m \\ v_{t+1}^m \\ a_{t+1}^m \end{pmatrix} = \begin{pmatrix} I & T_s I & \frac{T^2}{2} \\ 0 & I & T_s I \\ 0 & 0 & I \end{pmatrix} \begin{pmatrix} r_t^m \\ v_t^m \\ a_t^m \end{pmatrix} + \begin{pmatrix} \frac{T^3}{6} I \\ \frac{T^2}{2} I \\ T_s I \end{pmatrix} w_{a,t}. \]

(5.7)

The discretization of the attitude dynamics however is approximated with

\[ \begin{pmatrix} \Theta_{mb,t+1}^b \\ \omega_{mb,t+1}^b \end{pmatrix} = \begin{pmatrix} I + T_s f_t(\omega_{mb,t}^b, \Theta_t) & 0 \\ 0 & I \end{pmatrix} + \begin{pmatrix} T_s & \frac{T_s}{2} \frac{\partial}{\partial \omega_{mb,t}^b} f_t(\omega_{mb,t}^b, \Theta_t) \end{pmatrix} w_{\omega}, \]

(5.8)

where the approximation

\[ F_t = e^{A_t T_s} \approx I + A_t T_s, \]  
\[ G_t \approx T_s B_t + \frac{T_s^2}{2} A_t B_t, \]

(5.9a)  
(5.9b)

has been used. This approximation is valid for short sample intervals so that \( ||A^2 T_s^2||_2 \) is significantly smaller than \( ||AT_s||_2 \). The concatenated discrete model is given in (B.1).

5.3 Filtering Results

For the simulation the IMU provided updates at 100 Hz and the camera images at 4 Hz. The general results of the simulation is good and they show that it is possible to use an IMU and a camera to navigate. The results for each type of IMU are presented below together with the used parameters for the filters.
Figure 5.2. Difference between the true and the estimated position using high accuracy IMU.

5.3.1 High Accuracy IMU

The process noise was set to

$$Q_w = \begin{pmatrix} 10 \cdot I & 0 \\ 0 & I \end{pmatrix},$$

while the measurements was set to

$$R_{f_b} = I, \quad R_{\omega_{mb}} = 10^{-2} \cdot I, \quad R_c = 2 \cdot I,$$

where $R_{f_b}$ is the measurement noise for the accelerometers, $R_{\omega_{mb}}$ is the measurement noise for the accelerometers and $R_c$ the measurement noise for the correspondences. The result from the filter is a bit strange since the estimate in the position is not significantly better than for the low accuracy IMU, see Figure 5.2, although the attitude is estimated better, see Figure 5.4. The bad position estimates also effect the accuracy in the velocity as shown in Figure 5.3.
Figure 5.3. Difference between the true and the estimated velocity using high accuracy IMU.
Figure 5.4. Difference between the true and the estimated attitude using high accuracy IMU.
5.3 Filtering Results

Figure 5.5. Difference between the true and the estimated position using low accuracy IMU.

5.3.2 Low Accuracy IMU

The setup of the filter was

\[
Q_w = \begin{pmatrix} 10 \cdot I & 0 \\ 0 & I \end{pmatrix}, \quad R_f = 2 \cdot I, \quad R_{\omega_{\text{mb}}} = 10^{-1} \cdot I, \quad R_c = 2 \cdot I,
\]

where \(Q_w\) are the process noise, \(R_f\) measurement noise for the accelerometers, \(R_{\omega_{\text{mb}}}\) measurement noise for the accelerometers and \(R_c\) the measurement noise for the correspondences.

The result from the filter can be summarized by Figure 5.5, Figure 5.6 and Figure 5.7. It can be seen that the accuracy in position is close to five meters and the one in velocity somewhat better, both being acceptable results. The attitude however has an accuracy of 2 degrees in each angle and this can actually be a bit too rough for applications such as fire control. The estimation of the attitude is peculiar in another sense as well, since the estimates of each angle basically looks the same.
Figure 5.6. Difference between the true and the estimated velocity using low accuracy IMU.
Figure 5.7. Difference between the true and the estimated attitude using low accuracy IMU.
Chapter 6

Camera Aided Navigation

The normal navigation systems available today uses IMU, GPS and often also magnetometers and barometers, to determine the position and attitude of the aircraft. The common GPS aided navigation system works in the way that the GPS is used to estimate the error from the dead-reckoning of the IMU output, called INS, as described by Figure 6.1. This chapter investigates the possibilities to use a camera instead of the GPS as described by Figure 6.2. The advantage of trying to estimate the errors introduced by dead-reckoning rather than the actual value is that the errors tend to change much slower than the actual value, e.g., a Gripen can have a roll rate of 300 degrees per second, meaning that the roll will change rapidly while the error will still change modestly. In this chapter first the dead-reckoning algorithm, called Inertial Navigation System, INS, is presented, then the filter used to estimate the errors derivated via perturbation analysis of the inertial navigation system.

6.1 Inertial Navigation System - INS

An Integrated Navigation System, INS, uses the acceleration and angular velocity measured by the IMU to calculate the position, velocity and attitude of the aircraft. Apart from using the IMU some other sensors, e.g., magnetometers or a barometer are often used to increase the accuracy in the estimates, but for the equations given here such extra sensors have been excluded. The equations presented here are really the inverse of (2.15) and (2.16) and for further details please see Britting (1971) and Chatfield (1997). Before presenting the equations a note, on how the position of the aircraft will be denoted, is necessary. Its position can either be seen as a geodetic position, \( r^g = (\Phi, \lambda, h)^T \), see (2.10), or as a direction cosine matrix, \( R^g_{\xi} \), see (2.9). Here the former will be used as it gives more transparent equations, yielding the dynamic model

\[
\dot{r}^g = \begin{pmatrix}
\dot{\Phi} \\
\dot{\lambda} \\
\dot{h}
\end{pmatrix} = \begin{pmatrix}
\frac{v^g_x}{R+h} \\
\frac{v^g_y}{(R+h)\cos \Phi} \\
-v^g_z
\end{pmatrix}.
\]  

(6.1)
Figure 6.1. Overview of GPS aided INS

Figure 6.2. Overview of Camera aided INS
The dynamics of the velocity is given by
\[ \dot{v}^g = R_b^g f^b - (R_e^g 2\omega_{ie} + \omega_{eg}^g) \times v^g + G^g, \] (6.2)
where

- \( R_b^g \) is the rotation matrix from the body frame to the geodetic frame, (2.12),
- \( R_e^g \) is the rotation matrix from the earth frame to the geodetic frame, (2.9),
- \( \omega_{ie} \) is the earthrate, (2.8),
- \( \omega_{eg}^g \) is the craft rate, (2.11),
- \( f^b \) is the output from the accelerometers, see Section 2.5.

To represent the attitude of the aircraft the direction cosine matrix, \( R_b^g \), will be used, c.f. Section 5.2. Using (2.5) yields the dynamics
\[ \dot{R}_b^g = R_b^g \omega_{ib}^b - (R_e^g \omega_{ie}^e + \omega_{eg}^g), \] (6.3)
where \( \omega_{ib}^b \) is the output from the rate gyros and the other quantities are explained above.

Important to notice is that both \( \omega_{ie}^e \) and \( \omega_{eg}^g \) depend on the position \( r^g \) hence (6.1), (6.2) and (6.3) are connected. The equations presented here assumes that the IMU used are of such high accuracy that it can actually measure the earth rate and the coriolis force. For this to be achievable the simulated sensor disturbancies, see (4.7), need to be approximately less than \( 10^{-7} \).

### 6.2 Error Dynamic Modeling

Perturbating the three equations (6.1), (6.2) and (6.3) gives a relationship of how the error, introduced via bias and/or noise, on the output from the IMU will affect the estimates of the position, velocity and attitude calculated by the inertial navigation system. The error dynamics presented assumes that the output from the inertial navigation system can be seen as
\[ r_{INS}^{9.INS} = r^g + \delta r^g, \] (6.4a)
\[ v_{INS}^{9.INS} = v^g + \delta v^g, \] (6.4b)
\[ R_b^{9.INS} = (I - <\gamma^g>) R_b^g, \] (6.4c)
where \( \gamma^g = (\gamma_N, \gamma_E, \gamma_D)^T \) do not represent deviations in any certain angles but rather a transformation of \( R_b^g \) with an explicit orthogonality constraint (Britting, 1971). This transformation guarantees that \( R_b^g \) will still be a direction cosine matrix, see Section 2.1. The estimated \( \hat{R}_b^g \) can be recovered via
\[ \hat{R}_b^g = (I - <\gamma^g>)^{-1} R_{INS}^{9.INS}, \] (6.5)
and the Euler angles via the relation (2.13). As mentioned in Chapter 4 it is assumed that the aircraft flies on constant altitude, so that $\delta h = 0$ and $\delta v_D = 0$. For this thesis the assumption is made mainly to simplify the simulation, but for the error dynamics presented here it corresponds to the assumption that a separate perfectly calibrated barometer is used to stabilise the vertical channel, (Nordlund, 2002). The reason for not simply adding it as a state together with a vertical velocity is to keep the number of states small and hence also the number of particles used. The error equations presented here are based on derivations made in Nordlund (2002), Shin (2001) and Britting (1971) yielding

$$\dot{\delta r} = \left( \begin{array}{c} \delta \Phi \\ \delta \lambda \end{array} \right) = \left( \begin{array}{ccc} 0 & 0 & 0 \\ v_E \sin \Phi & 0 & 0 \\ \frac{v_E}{(R+h) \cos^2 \Phi} & 0 & \frac{1}{(R+h) \cos \Phi} \end{array} \right) \delta \nu^g, \quad (6.6)$$

$$\dot{\delta v} = \left( \begin{array}{c} \delta v_N \\ \delta v_E \end{array} \right) = \left( \begin{array}{ccc} -2v_E \omega_{ie}^c \cos \Phi - \frac{v_E^2}{(R+h) \cos^2 \Phi} & 0 & 0 \\ 2\omega_{ie}^c (v_N \cos \Phi - v_D \sin \Phi) & 0 & 0 \\ \frac{v_D}{R+h} & -2\omega_{ie}^c \sin \Phi - 2v_E \frac{v_N \tan \Phi}{R+h} & \frac{v_E v_N}{R+h} \end{array} \right) \delta \nu^g + \left( \begin{array}{c} -f_z^b \\ f_z^b \\ 0 \\ 0 \\ 0 \end{array} \right) \gamma^g + (r_1 r_2) \delta f^b, \quad (6.7)$$

where $R_b^g = (r_1 r_2 r_3)$, and

$$\dot{\gamma} = \left( \begin{array}{c} \dot{\gamma}_N \\ \dot{\gamma}_E \\ \dot{\gamma}_D \end{array} \right) = \left( \begin{array}{ccc} -\omega_{ie}^c \sin \phi & 0 & 0 \\ 0 & -\omega_{ie}^c \cos \phi - \frac{v_E}{(R+h) \cos^2 \Phi} & 0 \\ \omega_{ig,z}^g & \omega_{ig,y}^g & 0 \end{array} \right) \delta \nu^g + \left( \begin{array}{c} 0 \\ 0 \\ -1 \frac{1}{R+h} \end{array} \right) \frac{1}{R+h} \frac{1}{R+h} \dot{\delta \omega}^b_{ib}, \quad (6.8)$$

where $\delta f^b$ and $\delta \omega^b_{ib}$ are the disturbances in the output from the IMU, which will be modelled as white noise. In addition the position error could have been concatenated with $\delta h$ and $\delta v_D$ and included measurement from a simulated barometer. In a way this would likely have increased the accuracy in the estimates of the velocity in the northern and eastern directions as well. Together with the measurements
from the camera we get the continuous time model

\[
\dot{x}(t) = \begin{pmatrix}
\delta\Phi \\
\delta\lambda \\
\delta\dot{v}_N \\
\delta\dot{v}_E \\
\gamma_N \\
\gamma_E \\
\gamma_D
\end{pmatrix} = A(t)x(t) + B_w(t)w(t), \quad w(t) \in \mathcal{N}(0, Q),
\]

(6.9a)

\[
y_{c,t} = \begin{pmatrix}
-F_{n^c}^{n_z} \\
-F_{n^c}^{n_y}
\end{pmatrix} + e(t), \quad e(t) \in \mathcal{N}(0, R),
\]

(6.9b)

where \( n \) is a detected feature and its coordinates in the camera frame is given by

\[
n^c = R^c_br^b_g(r^b_g - n^g)
\]

(6.10)

The measurement from the camera, \( y_{c,t} \), are explained in Section 3.1 and \( A(t) \) and \( B_w(t) \) are given in (B.2) and (B.3), respectively.

### 6.2.1 Discretization of the Model

In order to implement the continuous time model (6.9) on a computer it needs to be discretized to the discrete state space model

\[
x_{t+1} = F_t x_t + G_{w,t} w_t, \quad w_t \in \mathcal{N}(0, Q_w).
\]

(6.11)

By assuming that \( A(t) \) and \( B(t) \) are constant during the sample interval and choosing the sample time \( T_s \) to \( \frac{1}{1000} \) s so that \( ||A^2T_s^2||_2 \) will be considerably smaller than \( ||AT_s||_2 \) an acceptable approximation, to the exact discretization technique given in Gustafsson (2000), is to use

\[
F_t = I + A_t T_s,
\]

(6.12a)

\[
G_{w,t} = B_{w,t} T_s + A_t B_{w,t} \frac{T_s^2}{2}.
\]

(6.12b)

### 6.3 Filtering Results

The derived dynamic model was implemented in a particle filter using 100000 particles and the IMU provided measurements at 100 Hz and the camera at 4 Hz.

#### 6.3.1 High Accuracy IMU

For the high accuracy IMU the process noise was set to

\[
Q_w = \text{diag}([5 \cdot 10^{-3}, \ 5 \cdot 10^{-3}, \ 5 \cdot 10^{-4}, \ 5 \cdot 10^{-4}, \ 5 \cdot 10^{-4}]),
\]

and the measurement noise for the correspondences was set to \( R_c = 10 \cdot I \).
The result from the filter was in general very good and can be summarized by Figure 6.3, Figure 6.4 and Figure 6.5. For the second turn, after 160 seconds, a slight decrease in the estimates can be seen, but the estimates are still unquestionable acceptable.

6.3.2 Low Accuracy IMU

The setup of the particle filter was to use the process noise

$$Q_w = \text{diag}(5 \cdot 10^{-2}, 5 \cdot 10^{-2}, 10^{-2}, 10^{-2}, 10^{-2}),$$

while the measurement noise for the correspondencies measurements was set to $R_c = 20 \cdot I$. The filtering results, see Figure 6.7, Figure 6.8 and Figure 6.9, shows the major weakness with the implemented model of the error in attitude. For a low accuracy IMU the drift in attitude is to big for the approximation (6.4c) to be valid. The skewsymmetric transformation $< \gamma >$ cannot make the estimated DCM $\hat{R}^b_g$ close enough to $R^b_g$ and eventually the filter is certain to diverge, which for the presented result happend after 200 seconds. By then the estimated attitude was to
Figure 6.4. Difference between the true and the estimated velocity using high accuracy IMU.
Figure 6.5. Difference between the true and the estimated attitude using high accuracy IMU.
poor resulting in that no correspondencies was found. To reinforce this reasoning one can calculate a $\hat{\gamma}$ by shifting (6.4c) to get

$$< \hat{\gamma} >= I - R^\theta_{b,INS} (R^\theta_{b})^{-1}.$$  \hfill (6.13)

For the case of a low accuracy IMU and having flown 100 seconds, for which the angles have drifted

$$\phi^{INS} - \phi \approx 1.9^\circ,$$  \hfill (6.14a)
$$\theta^{INS} - \theta \approx 0.04^\circ,$$  \hfill (6.14b)
$$\psi^{INS} - \psi \approx 4.4^\circ,$$  \hfill (6.14c)

the estimated skew symmetric transformation is

$$< \hat{\gamma} >= \begin{pmatrix}
3.0 \cdot 10^{-3} & 7.7 \cdot 10^{-2} & -9.5 \cdot 10^{-3} \\
-7.8 \cdot 10^{-2} & 3.5 \cdot 10^{-3} & 3.2 \cdot 10^{-2} \\
7.0 \cdot 10^{-3} & -3.2 \cdot 10^{-2} & 0.5 \cdot 10^{-3}
\end{pmatrix},$$ \hfill (6.15)

which, somewhat contradictory, clearly is not skew symmetric. The approximation (6.4c) is only valid when the true and dead-reckoned attitude is approximately the same. Investigating the situation after 200 seconds the drift in attitude are

$$\phi^{INS} - \phi \approx -6.6^\circ$$  \hfill (6.16a)
$$\theta^{INS} - \theta \approx -5.5^\circ$$  \hfill (6.16b)
$$\psi^{INS} - \psi \approx 9.1^\circ$$ \hfill (6.16c)

which yields the estimate

$$< \hat{\gamma} >= \begin{pmatrix}
2.3 \cdot 10^{-2} & 8.8 \cdot 10^{-2} & -1.9 \cdot 10^{-1} \\
-1.1 \cdot 10^{-1} & 9.8 \cdot 10^{-3} & 9.0 \cdot 10^{-2} \\
1.8 \cdot 10^{-1} & -1.1 \cdot 10^{-1} & 2.3 \cdot 10^{-2}
\end{pmatrix}.$$ \hfill (6.17)

This phenomenon will always occur for the proposed model, the higher the quality of the IMU the longer time it will take. To counter this weakness two options are available:

- Model the error in attitude differently.
- Feedback the estimated errors to the INS as illustrated in Figure 6.6.

The latter will result in that the drift in attitude (and all other states as well) will be reset after every correspondencies measurement. The stability of this system does, however, remain an open issue. Another way to counter this incompleteness would be to model $\delta R^\theta_{b}$, c.f., (6.4c), but that would require nine states instead of three. Nevertheless the issue has to be solved and using $\delta R^\theta_{b}$ is probably the most feasible way.
Figure 6.6. Feedbacked camera aided INS.

Figure 6.7. Difference between the true and the estimated position using low accuracy IMU.
Figure 6.8. Difference between the true and the estimated velocity using low accuracy IMU.
Figure 6.9. Difference between the true and the estimated attitude using low accuracy IMU.
Chapter 7

Concluding Remarks

Two ways to fuse the measurements from a camera and an IMU have been implemented and tested and the conclusions from these together with suggestions on further work are presented here.

7.1 Conclusions

For the simulated world studied in this thesis the concept of fusing measurements from an IMU and a camera to navigate has proved to work. The conclusions are summarized below

- For simulated camera images the implemented corner detector works and the RANSAC algorithm successfully detects outliers amongst the detected features.

- Normalized cross correlation is probably the most robust way to match a real life camera image against a reference, but it requires the attitude to be known.

- If a low accuracy IMU is used it is essential that the camera provides frequent updates or else the image processing might not be successful.

- Using correspondences and assuming Gaussian noise is a viable way to use the information in a camera image.

- It is possible to use the correspondencies measurements in both a Kalman filter and a particle filter.

- A system using a camera instead of a GPS has a similar performance as one using GPS, but is far more robust.
7.2 Future Work

As mentioned above the results show that a camera can be used instead of a GPS, at least in theory. The implemented image processing and filters need to be applied to real data to see how viable they really are. Although there is a difference between real life IMU data and simulated, the major issue concerns the image processing, for which a real camera image will be significantly different to a simulated one in terms of illuminance and contrast. Furthermore, a real camera image will also contain other details such as cars, differently ploughed cropland and weather effects which will aggravate the image processing even further. Once this has been tested and proved to work a camera can be used instead of GPS to navigate. The ideal way to fuse the information from the IMU and the camera is most likely in a marginalized particle filter, where particles are used to handle the camera measurements and a Kalman filter is used for the other states.
Bibliography


Appendix A

Notation

\(i\) Inertial coordinate frame.
\(e\) Earth-fixed coordinate frame.
\(g\) Geographic coordinate frame.
\(b\) Body coordinate frame.
\(m\) Map coordinate frame.
\(c\) Camera coordinate frame.
\(i\) Image coordinate frame.
\(n\) Detected feature in an image.
\(r\) Position vector.
\(v\) Velocity vector.
\(a\) Acceleration vector.
\(f\) Specific force.
\(F\) Focal length of Camera.
\(G\) Gravity vector.
\(\omega\) Angular velocity.
\(\phi, \theta, \psi\) Euler angles.
\(\lambda, \Phi\) Longitude, Latitude.
\(\langle \cdot \rangle\) Skew-symmetric transformation.
\(\hat{\cdot}\) Estimated quantity.
\(\Delta\) Bias.
\(Q_t\) Covariance of process noise at time \(t\).
\(R_t\) Covariance of measurement noise at time \(t\).
\(R^B_A\) Direction Cosine Matrix describing the rotation from frame A to frame B.
\(\mathcal{N}(\mu, \sigma)\) Gaussian distribution with mean \(\mu\) and covariance \(\sigma\).
\(\mathcal{N}(x; \mu, \sigma)\) Gaussian PDF with mean \(\mu\) and covariance \(\sigma\).
\(x \sim y\) \(x\) is distributed as \(y\).
A.1 Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BODY</td>
<td>Body coordinate frame.</td>
</tr>
<tr>
<td>GEO</td>
<td>Geodetic coordinate frame.</td>
</tr>
<tr>
<td>ECEF</td>
<td>Earth centerd coordinate frame.</td>
</tr>
<tr>
<td>ECI</td>
<td>Earth centerd inertial coordinate frame.</td>
</tr>
<tr>
<td>MAP</td>
<td>Map coordinate frame.</td>
</tr>
<tr>
<td>UTM</td>
<td>Universal Transversal Mercator.</td>
</tr>
<tr>
<td>DCM</td>
<td>Direction cosine matrix.</td>
</tr>
<tr>
<td>IMU</td>
<td>Inertial measurement unit.</td>
</tr>
<tr>
<td>INS</td>
<td>Inertial navigation system.</td>
</tr>
<tr>
<td>RANSAC</td>
<td>Random SAmple Cosensus.</td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System.</td>
</tr>
<tr>
<td>DOF</td>
<td>Degrees of freedom.</td>
</tr>
<tr>
<td>PDF</td>
<td>Probability distribution function.</td>
</tr>
</tbody>
</table>
Appendix B

Concatenated Dynamic Models

B.1 Discrete time model for a Flat World

\[
\begin{pmatrix}
    r_{m,t+1} \\
v_{m,t+1} \\
a_{m,t+1} \\
\Theta^b_{m,t+1} \\
w^b_{mb,t+1}
\end{pmatrix}
= 
\begin{pmatrix}
    I & T_s I & \frac{T^2}{2} & 0 & 0 \\
0 & I & T_s I & 0 & 0 \\
0 & 0 & I & 0 & 0 \\
0 & 0 & 0 & I + T_s f_t(\omega^b_{mb,t}, \Theta_t) & 0 \\
0 & 0 & 0 & 0 & I
\end{pmatrix}
\begin{pmatrix}
r_{t} \\
v_{t} \\
a_{t} \\
\Theta^b_{m,t} \\
w^b_{mb,t}
\end{pmatrix}
\]

\[+ \begin{pmatrix}
    \frac{T^3}{6} I \\
\frac{T^2}{2} I \\
T_s I \\
0 & \frac{T^2}{2} f_t(\omega^b_{mb,t}, \Theta_t) I & 0 & 0 \\
0 & 0 & T_s I
\end{pmatrix}
\begin{pmatrix}
w_{a,t} \\
w_{mb,t}
\end{pmatrix}
\]

(B.1)
### B.2 Error Dynamics

\[
A(t) = \begin{pmatrix}
0 & 0 & \frac{1}{R+h} & 0 & 0 \\
\frac{v_E \sin \Phi}{(R+h) \cos^2 \Phi} & 0 & 0 & \frac{1}{(R+h) \cos \Phi} & 0 \\
-2v_E \omega^e_{ie} \cos \Phi - \frac{v_E^2}{(R+h) \cos^2 \Phi} & 0 & \frac{v_D}{R+h} & -2\omega^e_{ie} \sin \Phi - 2v_E \tan \Phi & 0 \\
2\omega^e_{ie} (v_N \cos \Phi - v_D \sin \Phi) + \frac{v_E v_N}{(R+h) \cos \Phi} & 0 & 2\omega^e_{ie} \sin \Phi + \frac{v_E \tan \Phi}{R+h} & 0 & -f^b_x \\
-\omega^e_{ie} \sin \phi & 0 & 0 & \frac{R+h}{R+h} & f^b_y \\
0 & 0 & -\frac{1}{R+h} & -\tan \Phi & -\omega^n_{in,x} \\
-\omega^e_{ie} \cos \Phi - \frac{v_E}{(R+h) \cos^2 \Phi} & 0 & 0 & -\omega^n_{in,y} & -\omega^n_{in,x}
\end{pmatrix}
\]

(B.2)

\[
B_w(t) = \begin{pmatrix}
R^q_b(1,1) & R^q_b(1,1) & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
R^q_b(2,1) & R^q_b(2,1) & 0 & 0 & 0 \\
R^q_b(2,2) & R^q_b(2,2) & 0 & 0 & 0 \\
R^q_b(3,1) & R^q_b(3,1) & R^q_b(1,1) & R^q_b(1,1) & R^q_b(1,3) \\
0 & 0 & R^q_b(2,1) & R^q_b(2,2) & R^q_b(2,3) \\
0 & 0 & R^q_b(3,1) & R^q_b(3,2) & R^q_b(3,3)
\end{pmatrix}
\]

(B.3)