A Vision-aided Nonlinear Observer for Fixed-wing UAV Navigation

Jesper Hosen∗

Kongsberg Defence Systems, Kjeller, Norway

Håkon H. Helgesen†, Lorenzo Fusini‡, Thor I. Fossen§, Tor A. Johansen¶,

Norwegian University of Science and Technology, Centre for Autonomous Operations and Systems, Trondheim, Norway

This paper presents a vision-aided uniformly semi-globally exponentially stable (USGES) nonlinear observer for estimation of attitude, gyro bias, position, velocity and specific force of a fixed-wing Unmanned Aerial Vehicle (UAV). The nonlinear observer uses measurements from an Inertial Measurement Unit (IMU), a Global Navigation Satellite System (GNSS) receiver, and a video camera. This paper presents a nonlinear observer representation with a computer vision (CV) system without any assumptions related to the distance to objects in the images and the structure of the terrain being recorded. The CV utilizes a monocular camera and the continuous epipolar constraint to calculate body-fixed linear velocity. The observer is named a Continuous Epipolar Optical Flow (CEOOF) nonlinear observer. Experimental data from a UAV test flight and simulated data are presented showing that the CEOOF nonlinear observer has robust performance. Experimental results are compared with an Extended Kalman Filter (EKF) and illustrate that the estimates of the states converge accurately to the correct values. Results show that using the proposed CV in addition to IMU and GNSS improves the accuracy of the estimates. The CV provides accurate information about the direction of travel of the UAV, which improves the attitude and gyro bias estimate.

I. Introduction

Unmanned Aerial Vehicles (UAV) have in the last decade gained an increasingly interest, and already plays a major role in military use. The field of applications for UAVs will grow even more in the future, and the demands for robustness, safety and reliability are considered to be crucial. Robust navigation is one of the most important factors when working with UAVs. A challenge in navigation systems is to maintain accurate estimates of the states with low-cost measurement units. The output of such low-cost sensors are typically contaminated by noise and bias. As it is desirable to have low energy consumption on UAVs, it is necessary to find light weight navigation systems with good performance. The Kalman filter has been the preferred filter algorithm, but in recent years nonlinear observers, like the nonlinear complementary filter, have gained increased attention [1–6].

The use of cameras for navigational purposes is expected to grow quickly since video cameras are lightweight, energy efficient and the prices are constantly decreasing. As magnetometers are sensitive to disturbances, such as electromagnetic fields [7], cameras might be a good alternative or complementary to the magnetometer. The camera images can be used to output the body-fixed velocity of a UAV [8], but depend on favourable atmospheric conditions, light and detection of visual stationary features.
Computer Vision (CV) and Optical flow (OF) have been used for different applications in UAV navigation including indoor maneuvering [9, 10], linear and angular velocity estimation [11, 12] and obstacle avoidance [13–17], as well as height above the ground estimation [18]. [19, 20] uses OF to assist UAV landing without external sensor inputs. OF from a single camera is used in [21, 22] to estimate body axes angular rates of an aircraft as well as wind-axes angles. [23, 24] uses OF as input in Kalman filter-based navigation systems, fusing OF measurements with acceleration and angular velocity measurements. [25, 26] used camera as sensor for navigating in GPS-denied environments.

Attitude estimation has received significant attention as a stand-alone problem [1, 27–35]. In addition, other researchers have integrated Inertial Navigation System (INS), magnetometer/compass and GNSS to estimate the navigation states of a vehicle. [4] expanded the vector-based observer proposed by [1] and [32] to include GNSS velocity measurements. [27] and [28] built globally exponentially stable (GES) attitude estimators based on multiple time-varying reference vectors or a single persistently exiting vector. A similar observer was developed in [5, 36] to include also gyro bias and GNSS integration. [3] extended [36] to use linear velocity and specific force as reference vectors. [3] proved that feedback of estimated North-East-Down (NED) velocity and specific force in NED from the translational motion observer to the attitude observer, yield USGES in the origin of the error dynamics.

In this paper the observer presented in [3] is denoted as Ground Truth Optical Flow (GTOF) nonlinear observer. By assuming known distance to every feature in the camera image, the body-fixed velocity was recovered from the relationship between ego-motion and theoretical optical flow. This relationship is called the GTOF relationship between velocity and OF. The distance to every point was recovered by assuming flat horizontal terrain coinciding with NED, measured distance to the terrain by a laser altimeter and measured roll and pitch of the UAV relative to NED by an inclinometer. The assumption of flat and horizontal terrain will cause the CV in the GTOF nonlinear observer to produce erroneous velocity measurements in the case of flying over rugged terrain. Therefore it is desirable to exchange the CV of the GTOF nonlinear observer with a CV system with no requirement of flat horizontal terrain.

OF describes how objects in an image plane moves between two consecutive images. The motion in the image plane is caused by relative motion between the camera and the visual features being detected. In the simplest case it could be understood as the pixel displacement of a single feature between two successive images. The OF can be represented as multiple vectors describing the change in the image plane in time. Several methods exists for determining the OF of a series of images [37–40].

A camera fixed to a UAV can be used to recover the motion of the vehicle relative to the scene. An effective principle for recovering ego-motion of a camera is epipolar geometry. Epipolar geometry has been applied in e.g. navigation, landing and collision avoidance [12, 23, 41–45]. [46] presented the epipolar constraint in the continuous case. [47] and [48] have used the continuous epipolar constraint to recover the velocity of a UAV.

In this paper OF vectors together with the continuous epipolar constraint [46] are used to calculate the normalized body-fixed velocity of the UAV, and fed into the nonlinear observer as a reference vector. The use of the continuous epipolar constraint eliminates the dependency on the depth in the image. In practice this means that prior information about the distance to features and structure of the terrain are not required any more. Thus the observer is applicable when flying over any terrain.

A. Contribution of this Paper

This paper presents a more robust CV subsystem for the nonlinear observer from [3]. In [3] the ground truth optical flow (GTOF) relationship between motion and OF were used to recover the ego-motion of the UAV. A fundamental restriction in [3] was that the distance to every feature corresponding to an OF vector must be known in order to calculate the body-fixed linear velocity. The CV in this paper utilizes epipolar geometry [49] and only depends on the angular velocity of the UAV. Furthermore it works without knowing the distance to the features in the image. To the authors knowledge, this is the first time the continuous epipolar constraint has been employed in a USGES nonlinear observer.

Experimental and simulated results show that the proposed CEOF observer has comparable performance with the GTOF observer from [3] when flying over flat horizontal terrain. Simulations show that the proposed CEOF observer is structure independent, and that it outperforms the GTOF observer when flying above rugged and elevated terrain. Moreover, results show that using CV increases the accuracy of the estimates, compared to using only IMU and GNSS measurements. This is particularly clear in the attitude, as CV provides information about the direction of the body-fixed velocity. A pure IMU and GNSS approach assumes zero crab and flight path angle, and thus looses important information about the attitude. The experimental
results are compared to an EKF, while the simulated results are compared to the known reference. The results imply that the CEOF observer is a robust option to the GTOF nonlinear observer.

The last contribution is a stability proof showing that the CEOF observer has the same stability properties as the GTOF observer, namely a USGES equilibrium point at the origin of the error dynamics.

II. Notation and Preliminaries

Matrices and vectors are represented by uppercase and lowercase letters respectively. $X^{-1}$ and $X^+$ denote the inverse and the pseudo-inverse of a matrix respectively, $X^T$ the transpose of a matrix or vector, $\hat{X}$ the estimated value of $X$, and $\hat{X} = X - \hat{X}$ the estimation error. $\| \cdot \|$ denotes the Euclidean norm, $I_{n \times n}$ the identity matrix of order $n$, and $0_{m \times n}$ the $m \times n$ matrix of zeros. A vector $x = [x_1, x_2, x_3]^T \in \mathbb{R}^3$ is represented in homogeneous coordinates as $\tilde{x} = [x_1, x_2, x_3, 1]^T$. The function sat($\cdot$) performs a component-wise saturation of its vector or matrix argument to the interval $[-1, 1]$. The operator $[x]_x$ transforms the vector $x$ into the skew-symmetric matrix

$$[x]_x = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}$$

The inverse operation is denoted as vex($\cdot$), such that vex($[x]_x$) = $x$. The determinant of a matrix $A$ is denoted $\det(A)$. The skew symmetric part of a square matrix $A$ is obtained by the operator $P_a(A) = \frac{1}{2}(A - A^T)$.

The North-East-Down, camera- and the body-fixed reference frames are used in this paper as shown in Fig. 1: the body-fixed frame are denoted $\{B\}$ and the North-East-Down (NED) frame denoted $\{N\}$ (Earth-fixed, considered inertial), while the camera frame is denoted $\{C\}$. The rotation from $\{B\}$ to $\{N\}$ is represented by the matrix $R_b^N \in SO(3)$, with $SO(3)$ representing the Special Orthogonal group. The image plane is denoted $\{M\}$. $\{B\}$ and $\{C\}$ are assumed to be aligned, i.e. the camera is strapped to the body.

A vector decomposed in $\{B\}$ and $\{N\}$ has superscript $b$ and $n$ respectively. The subscript of a vector indicates which frame is measured relative to. For instance $p_{b/n}^n$ is the position of $\{B\}$ relative to $\{N\}$ expressed in $\{N\}$. The camera location w.r.t. $\{N\}$ is described by $c^n = [c_{x}^n, c_{y}^n, c_{z}^n]^T$. A point in the environment expressed w.r.t. $\{N\}$ is $p^n = [x^n, y^n, z^n]^T$. The same point expressed in $\{C\}$ is $p^c = [x^c, y^c, z^c]^T$. It will also be assumed that every point is fixed w.r.t. $\{N\}$. The Greek letters $\phi$, $\theta$, and $\psi$ represent the roll, pitch, and yaw angles respectively, defined according to the $zyx$ convention for principal rotations [6], and they are collected in the vector $\Theta_{b/n} = [\phi, \theta, \psi]^T$. A 2-D camera image has coordinates $x^m = [r, s]^T$, with the $y^b$- and $x^b$-axis respectively (see Fig. 3). The corresponding homogeneous image coordinate is denoted $\tilde{x}^m = [x^m, y^m, z^m]^T$. The derivative $[\dot{r}, \dot{s}]^T$ of the image coordinates is the OF. The subscript $cv$ indicates a quantity evaluated by means of the computer vision, $imu$ indicates a quantity measured by the IMU, while $GPS$ indicates that the quantity is measured by the GNSS.

A. Measurements and Sensors

The observer is designed to take use of a IMU, a GPS receiver and a video camera, providing the following measurements:

- **GPS**: NED position $p^n$ and NED velocity $v^n$.
- **IMU**: biased angular velocity $\omega_{imu} = \omega_{b/n}^b + b_{gyro}^b$, where $b_{gyro}^b$ represents the gyro bias, and specific force $f_{imu}^b = f_{b/n}^b$.
- **Camera**: 2-D projections $x^m = [r, s]^T$ onto the image plane $\{M\}$ of points $[x^n, y^n, z^n]^T$ in $\{N\}$.

Detailed information on the actual sensors employed in the experiment is presented in Section V.

III. Computer Vision

The observer presented in Section IV depends on body-fixed velocity measurements from the on-board camera. These measurements are generated through OF, therefore it is necessary to compute the OF vectors
for consecutive images before these vectors are transformed to velocity measurements. The OF calculation and the transformation are presented in the forthcoming section.

A. Optical flow computation

There exist several methods for computing OF. For the experiment presented in Section V two specific methods are chosen. The first one is SIFT [39] which provided the overall best performance in [8]. The second method is a region matching-based method [8], namely template matching utilizing cross-correlation [50].

SIFT uses a feature-based approach to compute OF. A set of features are extracted from two consecutive images with a feature detector. The detected features are then matched together to find common features in successive images. An OF vector is created from the displacement of each feature. The total number of such vectors in each image depends on the number of features detected and successfully matched.

It is desired to make sure that the OF algorithm produces at least two OF vectors to calculate the body-fixed velocity. It is not possible to guarantee a given number of vectors with SIFT since homogeneous environments, like snow or the ocean, increase the difficulty of finding distinct features. Therefore the OF vectors created by SIFT are combined with OF vectors from template matching [51].

The combination of two individual OF methods increases the probability of having OF vectors distributed across the whole image, as well as maintaining a high number of OF vectors. An example of OF vectors computed with SIFT and template matching from UAV test flights is displayed in Fig. 2.

In case of mismatches, both methods create erroneous OF vectors. It is desired to locate and remove these vectors. Therefore a simple outlier detector is implemented before the vectors are used to calculate body-fixed velocities. The outlier detector utilizes a histogram to find the vectors that deviates from the mean with respect to direction and magnitude.

B. Transformation from optical flow to velocity

For the OF computations to be useful in the observer a transformation to body-fixed velocity is necessary. The transformation is motivated by the continuous epipolar constraint and the pinhole camera model [52].

The camera-fixed coordinate system, \{C\}, is related to \{N\} as illustrated in Fig. 3. The focal point of the camera is for simplicity assumed to coincide with the origin of \{B\}. A point \(p\) in the terrain is projected from \{C\} to \{M\} by the pinhole camera model by

\[
\vec{x}^m = \frac{1}{z^c} K \vec{p}^c
\]  

(1)
Figure 2: a) Image captured at time $t_0$. b) Image captured at time $t_0 + \Delta t$. c) Optical flow vectors between image a) and b), generated by SIFT (red) and Template Matching (green).

where $\mathbf{x}_m$ is the homogeneous image coordinate and $K$ is a projection matrix mapping points in the camera frame to the image plane. It is defined as

$$K = \begin{bmatrix} 0 & f & 0 \\ -f & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$ (2)

where $f$ is the focal length of the camera. The focal length of a camera can be verified by the computer vision toolbox in Matlab. The same toolbox can be used to estimate coefficients describing the distortion of the camera. These coefficients can be used to generate undistorted images. For the rest of this paper, it is assumed that the distortion is insignificant.

$u^c$ is defined as the back-projected point lying on the projection ray between the origin of $\{C\}$ and $p^c$ with unity $z$-component

$$u^c = K^{-1} \mathbf{x}_m$$ (3)

Epipolar geometry [49] relates the motion of the camera frame with the motion in the image plane independent of the distance to the scene and the structure being recorded. By assuming that all matched features are at rest w.r.t $\{N\}$, the continuous epipolar constraint [46] can be expressed as

$$\left( \dot{u}^c T + u^c T \left[ \frac{\omega^c_{c/n}}{v^c_{c/n}} \times \right] T \right) (v^c_{c/n} \times u^c) = 0$$ (4)

where $\omega^c_{c/n}$ and $v^c_{c/n} = [v_x, v_y, v_z]^T$ are the angular and linear velocity of the camera relative to $\{N\}$ expressed in $\{C\}$, respectively. Note that the epipolar geometry has an inherited sign ambiguity due to the fact that the scale is not preserved. This means that it is only possible to determine the body-fixed velocity up to scale.
Using now the properties of a triple product [53], (4) can be rewritten as
\[ \mathbf{v}_{c/n}^T \left( \mathbf{u}^c \times \left( \dot{\mathbf{u}}^c T + \mathbf{u}^c T \left[ \omega_{c/n}^c \times \right] \right) \right) = 0 \] (5)

(5) might be rewritten as a linear equation in \( \mathbf{v}_{c/n}^c \). The crossproduct term is defined as:
\[ \mathbf{c} := \mathbf{u}^c \times \left( \dot{\mathbf{u}}^c T + \left[ \omega_{c/n}^c \times \right] \mathbf{u}^c \right) = [c_x, c_y, c_z]^T \]

If the angular velocity is measured, then all quantities in the crossproduct term \( \mathbf{c} \) are known. Using the definition of \( \mathbf{c} \), (5) is rewritten as
\[ \mathbf{v}_{c/n}^T \mathbf{c} = \mathbf{c}^T \mathbf{v}_{c/n}^c = 0 \] (6)

Assuming that a fixed-wing UAV will never have zero forward velocity, then since \( \{C\} \) and \( \{B\} \) are aligned, one can divide (6) by the forward velocity component \( v_x \neq 0 \)
\[ \frac{1}{v_x} \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \mathbf{c} = \frac{1}{v_x} \mathbf{v}_{c/n}^c = \begin{bmatrix} c_y, c_z \end{bmatrix} = 0 \]
\[ [c_y, c_z] \begin{bmatrix} v_y \\ v_z \end{bmatrix} = -c_x \] (7)

As can be seen from (7), one ends up with a linear equation. Assuming \( N \) features, the scaled body-fixed velocity with unity forward component can be found as:
\[ \mathbf{v}_{c/n}^c = v_x \mathbf{A}^T \mathbf{b}, \quad v_x \neq 0 \]
\[ \mathbf{A} = \begin{bmatrix} c_{y,1} & c_{z,1} \\ \vdots & \vdots \\ c_{y,N} & c_{z,N} \end{bmatrix} \]
\[ \mathbf{b} = -\begin{bmatrix} c_{x,1} \\ \vdots \\ c_{x,N} \end{bmatrix} \]
\[ \mathbf{u}_j^c \times \left( \dot{\mathbf{u}}_j^c T + \left[ \omega_{c/n}^c \times \right] \mathbf{u}_j^c \right) = [c_{x,j}, c_{y,j}, c_{z,j}]^T \] (8)
This gives a correct solution only if $A$ has full rank. This can only happen if the OF algorithm chooses linearly independent feature points and OF vectors as defined in Def. 1. Linearly independent OF vectors are in general obtained by not choosing all features from the same line in the image plane. $u_j^c = K^{-1}x_j^m$ and $\dot{u}_j^c = K^{-1}[\dot{r}_j, \dot{s}_j, 0]^T$ are the back projected coordinate and OF of feature $j$ respectively. Recall the sign ambiguity of the epipolar geometry, meaning that one must know the sign of $v_x$ to recover the normalized linear velocity. For a fixed-wing UAV the forward velocity will always be greater than zero, $v_x > 0$.

**Definition 1. Linearly Independent Optical Flow Vectors**

A pair of image features and their corresponding optical flow vectors $x_1^m, \dot{x}_1^m$ and $x_2^m, \dot{x}_2^m$, are said to be linearly independent if and only if the rank of $A$ in (8) is full, yielding $[v_y, v_z]^T = v_x A^+ b$ to be uniquely defined. The rank is full if and only if some $2 \times 2$ sub-matrix of $A$, $A_{2 \times 2}$, has $\det(A_{2 \times 2}) \neq 0$.

**IV. Observer Design**

**A. Kinematics**

The kinematics of attitude, position, and velocity are described by

\[
\begin{align*}
\dot{R}_n^b &= R_n^b \left[ \omega_{b/n} \right] \times \\
\dot{p}_{b/n}^n &= v_{b/n}^n \\
\dot{v}_{b/n}^n &= \mathbf{f}_{b/imu}^n + g^n
\end{align*}
\]

The objective is to estimate the attitude $R_n^b$, the position $p_{b/n}^n$, and the velocity $v_{b/n}^n$ with exponential convergence rate. In addition to this, an estimator for the gyro bias $\mathbf{b}_{gyro}^b$ is also provided.

**B. Assumptions**

The observer design by [3] is based on the following assumptions:

**Assumption 1.** The gyro bias $\mathbf{b}_{gyro}^b$ is constant, and there exists a known constant $L_b > 0$ such that $\|\mathbf{b}_{gyro}^b\| \leq L_b$.

**Assumption 2.** There exists a constant $c_{obs} > 0$ such that, $\forall t \geq 0$, $\|v_{cco}^b \times f_{imu}^b\| \geq c_{obs}$.

Assumption 2 states that the UAV cannot have a specific force parallel to the velocity of the UAV. Furthermore neither the specific force nor the velocity can be identically equal to zero. In practice this condition restricts the types of maneuvers that ensure guaranteed performance of the proposed observer. This is however not a problem for fixed-wing UAVs as they always have forward speed to remain airborne. Moreover the observer does not converge while the vehicle is at rest without aiding from e.g. a magnetometer, but presents no issues during flight.

For the CEOF observer, two assumptions are introduced to ensure that CV can recover the body-fixed velocity.

**Assumption 3.** The UAV has forward body-fixed velocity, $v_x > 0$.

**Assumption 4.** The OF algorithm provides at least two linearly independent OF vectors, as defined in Def. 1.
Figure 4: Block diagram of the observer. Σ₁ represents the attitude observer, and Σ₂ the translational motion observer. The feedback illustrated in green have been proved to yield USGES stability of the nonlinear observer. The stability of the gyro bias feedback illustrated in blue has not been analyzed.

C. Observer Equations

Provided Assumptions 1-4 hold, the CEOF observer representation is stated as

\[
\begin{align*}
\Sigma_1 & \begin{cases}
\dot{\hat{\theta}}_b &= \hat{\theta}^n \left(\hat{\omega}^b_{\text{imu}} - \hat{b}^b_{\text{gyro}}\right) + \sigma K_P \dot{\hat{\theta}}^n \\
\dot{\hat{b}}^b_{\text{gyro}} &= \text{Proj}(\hat{b}^b_{\text{gyro}}, -K_I \text{vec}(\hat{P}^n_a(\hat{R}^n_s K_P \dot{\hat{\theta}}^n)))
\end{cases} \\
\Sigma_2 & \begin{cases}
\dot{\hat{\theta}}_b/n &= \hat{\theta}^n_{b/n} + K_{PP}(\hat{P}^n_{GPS} - \hat{\theta}^n_{b/n}) \\
\dot{\hat{b}}^n_{b/n} &= \hat{\theta}^n_{b/n} + g^n + K_{PV}(\hat{P}^n_{GPS} - \hat{\theta}^n_{b/n}) \\
\dot{\hat{x}} &= -\sigma K_P \dot{\hat{\theta}}^n_{b/n} + K_{Pv}(\hat{P}^n_{GPS} - \hat{\theta}^n_{b/n}) \\
\dot{\hat{f}}^n_{b/n} &= \hat{\theta}^n_{b/n} + \hat{x}
\end{cases}
\end{align*}
\]

(10)

\[
\begin{align*}
\nu^b_{cv} &= \text{sign}(\nu_x) \frac{\nu_x}{||\nu_x||} \\
\nu_x &= \frac{\nu^n_{b/n}}{||\nu^n_{b/n}||} = [1, (A^*b)^T]^T, \quad \nu_x \neq 0 \\
u^b_j \times \left(\omega^b_{\text{imu}} - \hat{b}^b_{\text{gyro}}\right) &= \left[c_{x,j}, c_{y,j}, c_{y,j}\right]^T
\end{align*}
\]

(12)

The subsystem Σ₁ represents the attitude observer, whereas Σ₂ represents the translational motion observer. The CV gives (12), together with (8). σ ≥ 1 is a scaling factor tuned to achieve stability, \(k_f\) is a positive scalar gain and \(K_P\) is a symmetric positive definite gain matrix. Proj(·, ·) represents a parameter projection [54] that ensures that \(\|\hat{b}^b_{\text{gyro}}\|\) does not exceed a design constant \(L_b > L^*_b\) (see Appendix), and \(\hat{R}_s = \text{sat}(\hat{R}^n_b)\). \(K_{PP}, K_{PV}, K_{Pv}, K_{Pv}, K_{Pv}, K_{\xi_P}, K_{\xi_v}\) are observers gains, and \(g^n\) is the gravity vector in \{N\}. The matrix \(J\) is the output injection term, whose design is inspired by the TRIAD algorithm [55] and defined as

\[
\begin{align*}
\dot{J}(\nu^b_{cv}, \nu^n_{b/n}) &= \dot{A}^b \nu^n_{b/n} - \dot{\hat{R}}^n_{b} A^b T \\
A^b &= [J^b_{\text{imu}} \times \nu^b_{cv}, J^b_{\text{imu}} \times (f^b_{\text{imu}} \times \nu^b_{cv})] \\
\hat{A}^b &= [\dot{f}^n_{b/n} \times \nu^n_{b/n}, \dot{f}^n_{b/n} \times (\dot{f}^n_{b/n} \times \nu^n_{b/n})]
\end{align*}
\]

(13a)

(13b)

(13c)

The system Σ₁−Σ₂ is a feedback interconnection, as illustrated by Fig. 4.
D. Stability Proof

The error dynamics of the nonlinear observer can be written in a compact form as

\[
\Sigma_1 \begin{cases} 
\dot{\hat{R}}^n_b = R^n_b \left[ \omega^n_{b/n} \right]_x - \dot{R}^n_b \left[ \omega^n_{imu} - \dot{\hat{b}}^n_{gyro} \right]_x - \sigma K_p \dot{J} \\
\dot{\hat{b}}^n_{gyro} = -\text{Proj}(\dot{\hat{b}}^n_{gyro}, -k_I \text{vex}(\hat{P}_a(R^n_b K_p \hat{J})))
\end{cases}
\]

\[
\Sigma_2 \begin{cases} 
\dot{\hat{w}} = (A_w - K_w C_w) \hat{w} + B_w \hat{d}
\end{cases}
\]

where \( \hat{w} = [(\dot{\hat{R}}^n_b)\times, (\dot{\hat{v}}^n_{b/n})^T, (\dot{\hat{a}}^n_{b/n})^T]^T \) collects the estimated position, velocity and acceleration vectors, \( \hat{d} = (R^n_b \left[ \omega^n_{b/n} \right]_x - \dot{R}^n_b \left[ \omega^n_{imu} - \dot{\hat{b}}^n_{gyro} \right]_x) \dot{\hat{R}}^n_b + (R^n_b - \dot{R}^n_b) \dot{\hat{a}}^n_{b/n} \), and the four matrices in (14b) are defined as

\[
A_w = \begin{bmatrix} 0_{6\times3} & I_6 \\ 0_{3\times3} & 0_{3\times6} \end{bmatrix}, \quad B_w = \begin{bmatrix} 0_{6\times3} \\ I_3 \end{bmatrix},
\]

\[
C_w = \begin{bmatrix} I_6 & 0_{6\times3} \end{bmatrix}, \quad K_w = \begin{bmatrix} K_{pp} & K_{pv} \\ K_{vp} & K_{cv} \\ K_{xp} & K_{xv} \end{bmatrix}.
\]

The following theorem can be stated about the stability of the nonlinear observer (10)-(12), if assuming that \( \dot{\hat{b}}^n_{gyro} \) is kept constant in (12).

**Theorem 1. (Stability of the CEOF observer)** Let \( \sigma \) be chosen to ensure stability according to Lemma 1 in [5] and define \( H_K(s) = (I_s - A_w - K_w C_w)^{-1} B_w \). There exists a set \((0,c)\) such that, if \( K_w \) is chosen such that \( A_w - K_w C_w \) is Hurwitz, and \( \|H_K(s)\|_{\infty} < \gamma \), for \( \gamma \in (0,c) \), then the origin of the error dynamics (10)-(12), provided Assumptions 1-4, is USGES when the initial conditions satisfy \( \|\dot{\hat{b}}^n_{gyro}(0)\| \leq L_b \).

**Proof.** Proof is based on Theorem 1 in [3], where \( M \) is being replaced with the new computer vision subsystem from (12). It must be shown that \( \dot{v}^b_{cv} \) is uniquely defined. Then it follows from Theorem 1 in [3] that the origin of the error dynamics (10)-(12) is USGES.

Moreover if the sign of \( v_x \) is known, then \( \dot{v}^b_{cv} = \frac{v^b_{cv}}{\|v^b_{cv}\|} \). From Assumption 3 \( v_x > 0 \), hence the uniqueness of \( \dot{v}^b_{cv} \) can be shown by the uniqueness of \( v_x \). \( v_x = [1,(A^T b)^T]^T \) has a unique solution if and only if the rank of \( A \) is full [53]. Given that the computer vision algorithm extracts features such that Assumption 4 is not violated, then \( A \) has full rank, and \( v_x \) is uniquely determined. Hence \( \dot{v}^b_{cv} \) is uniquely determined, and it follows from Theorem 1 in [3] that the system is USGES.

\[ \square \]

V. Experimental Results

An experiment is carried out to validate the theory in practice. The UAV employed is a UAV Factory Penguin-B, equipped with a custom-made payload that includes all the necessary sensors. The IMU is a Sensonor STIM300, a low-weight, tactical grade, high-performance sensor that includes gyroscopes, accelerometers, and inclinometers, all recorded at a frequency of 300 Hz. The chosen GPS receiver is a uBlox LEA-6T, which gives measurements at 5 Hz. The video camera is an IDS GigE uEye 5250CP provided with a 8mm lens. The camera is configured for a hardware-triggered capture at 10 Hz. The experiment has been carried out on 6 February 2015 at the Eggemoen Aviation and Technology Park, Norway, in a sunny day with good visibility, very little wind, an air temperature of about -8°C. The terrain is relatively flat and covered with snow.

The observer is evaluated offline with the flight data gathered at the experiment. It is implemented using first order forward Euler discretisation with a time-varying step depending on the interval of the data acquisition of the fastest sensor, namely the STIM300, and it is typically around 0.003 seconds. The gyro bias is initialized by averaging the gyroscope measurement at stand still before take-off. The position estimate is
initialized by using the first GPS measurement, while the NED velocity is initialized by the difference between the first two consecutive GPS measurements. The various parameters and gains are chosen as \( L_b = 2^°/s \), \( L_{\dot{b}} = 2.1^°/s \), \( \sigma = 1 \), \( K_P = \text{diag}[0.08, 0.04, 0.06] \), \( k_I = 0.0001 \), \( K_{pp} = 30I_{3 \times 3} \), \( K_{pv} = 2I_{3 \times 3} \), \( K_{vp} = 0.01I_{3 \times 3} \), \( K_{\dot{v}v} = 20I_{3 \times 3} \), \( K_{\xi \dot{p}} = I_{3 \times 3} \), and \( K_{\xi v} = 50I_{3 \times 3} \).

The reference provided for the attitude, position, and velocity is the output of the EKF of the autopilot mounted on the Penguin-B. A reference for the gyro bias is not available.

All the images are processed with a resolution of 1600×1200 (width×height) pixels and in their original state, without any pre-processing. The lens distortion of the camera is not accounted for, and no correction is applied to the images. SIFT is implemented with the open source computer vision library (OpenCV) [56] with default settings. Each match is tagged with a value indicating the accuracy of the match, and the smallest of these values is considered to be the best match. To increase the reliability of the OF vectors, each match is compared to the best one. Every match with an uncertainty more than double the uncertainty of the best match is removed. Also the template matching algorithm is implemented with OpenCV. The size of the templates is chosen to be 120×90 pixels and a correlation of 99% is required in order for a template match to be considered reliable and not removed.

In addition to the CEOF and GTOF observer, a nonlinear observer without CV is implemented. This is done by removing the CV subsystem in (12) from the nonlinear observer, and approximating the body-fixed linear velocity measurement by \( v^b = [1, 0, 0]^T \). The nonlinear observer without CV is denoted NoCV. Although Theorem 1 does not cover feedback of the gyro bias estimate to CV in the CEOF nonlinear observer, this feedback is implemented. This is assumed to increase the accuracy without being destabilizing, as the bias estimator is tuned to have slow dynamics.

\section*{A. Results}

The results presented here refer to a complete flight of the Penguin-B, from take-off to landing. The time on the x-axis is the elapsed time since the data logging began, and only the significant part involving the flight is presented. The maneuvers performed include flights on a straight line and turns with a large and small radius of curvature, approximately 200 m and 100 m. Preliminary experimental results for the GTOF nonlinear observer were reported in [3], while here experimental results for both the GTOF and CEOF nonlinear observers are presented and compared.

Fig. 5 shows the measured body-fixed velocity from the GTOF CV. The measurements are contaminated by noise. The mean values are close to the reference, although the mean forward velocity (\( u \)) is slightly greater than the reference. The measured crab and flight path angle of the UAV are shown in Fig. 6. It is seen that both the GTOF and CEOF CV succeeds in measuring the correct direction, but GTOF has a larger noise level than CEOF.

Fig. 7 illustrates the estimated attitude. It can be seen that all observers need approximately 60 seconds to converge. The estimates of the roll angle are fairly similar for NoCV, GTOF and CEOF. The estimated pitch angle has a small offset for all nonlinear observers throughout the entire flight. The yaw angle estimate is almost identical for the NoCV, GTOF and CEOF. Fig. 8 and Fig. 9 illustrates the estimated velocity and position in \{N\}, and shows small differences for NoCV, GTOF and CEOF. The estimated gyro bias is seen in Fig. 10. No bias reference is available, but the estimated bias is close to equal for NoCV, GTOF and CEOF. The flight terrain is relatively flat and the UAV has small crab and flight path angle during the flight. Therefore the weaknesses of the GTOF and NoCV observer are not significant in the results. However the experimental results show that the nonlinear observers yield small deviations from the reference EKF, and that the CV give reasonable estimates of normalized body-fixed velocity.

\section*{VI. Simulation Results}

In order to evaluate the NoCV, GTOF and CEOF observer representations in the presence of more rugged terrain and to compare with an exactly known reference, a simulator is implemented in Matlab. An elevation profile of a coastline is generated, and a UAV flight is simulated.

The following parameters and gains are chosen identical for the NoCV, GTOF and CEOF observer: \( L_b = 2^°/s \), \( L_{\dot{b}} = 2.1^°/s \), \( \sigma = 1 \), \( K_{pp} = \text{diag}[5.5, 5.7] \), \( K_{pv} = \text{diag}[50, 50, 50] \), \( K_{vp} = \text{diag}[0.01, 0.01, 0.01] \), \( K_{\dot{v}v} = 10I_{3 \times 3} \), \( K_{\xi \dot{p}} = 0.1I_{3 \times 3} \), and \( K_{\xi v} = 5I_{3 \times 3} \). For the GTOF and CEOF observer \( K_P = I_{3 \times 3} \) and \( k_I = 0.03 \) are chosen. The NoCV is tuned with \( K_P = \text{diag}[1, 0.2, 0.1] \) and \( k_I = 0.01 \).
Figure 5: Measured and estimated body-fixed velocity by GTOF and autopilot EKF respectively.

Figure 6: Measured and estimated crab and flight path angle.

Figure 7: Estimated attitude by the observers.

Figure 8: Estimated velocity by the observers.

Figure 9: Estimated position by the observers.

Figure 10: Estimated gyro bias by the observers.
All observers are initialised with \( \hat{p}_b^n = I_{3 \times 3}, \hat{b}^b_{\text{gyro}} = 0_{3 \times 1} \). \( \hat{b}^b_{\text{gyro}} \) and \( \hat{v}^b_{n/b} \) are initialised as the first GNSS position and velocity measurement respectively.

### A. UAV Path

Linear and angular velocity, \( \nu^b_{n/b} \) and \( \omega^b_{n/b} \), are generated over a time interval of 200 sec. Wind directed straight north with magnitude 5 m/s is simulated causing the UAV to have a crab angle. As the camera measures the velocity relative to the ground, one does not have to consider the sideslip angle. Kinematic equations are used to generate positions and attitude of the UAV.

\[
\nu^b_{n/b} = R^b_n(\Theta_{b/n})v^b_{b/n} \\
\omega^b_{n/b} = R^b_n(\Theta_{b/n})(v^b_{b/n} + \omega^b_{b/n} \times v^b_{b/n}) - \nu^n \\
\dot{\Theta}_{b/n} = T_\Theta(\Theta_{b/n})\omega^b_{b/n}
\]


### B. Sensor Data

Sensor data are generated before running the observer. A gyroscope, accelerometer, inclinometer, GNSS and CV are simulated. The GNSS is simulated to measure \{N\} position and velocity, and CV is simulated to measure the OF. The gyroscopic, accelerometer, inclinometer are configured to output measurements with a rate of 100 Hz. The GNSS is configured to output measurements at 5 Hz. The noise on the position measurement from GNSS is modelled as a Gauss-Markov process by \( \nu(k + 1) = e^{-K_{\text{GNSS}}\Delta T}\nu(k) + \eta_{\text{GNSS}} \), with noise parameters given in Table 1.

<table>
<thead>
<tr>
<th>Direction</th>
<th>Std. dev. ( \eta_{\text{GNSS}} ) [m]</th>
<th>( 1/K_{\text{GNSS}} ) [s]</th>
<th>( \Delta T_{\text{GNSS}} ) [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>North</td>
<td>0.21</td>
<td>360</td>
<td>0.2</td>
</tr>
<tr>
<td>East</td>
<td>0.21</td>
<td>360</td>
<td>0.2</td>
</tr>
<tr>
<td>Down</td>
<td>0.4</td>
<td>360</td>
<td>0.2</td>
</tr>
</tbody>
</table>

The camera is simulated to capture 25 frames per second. The camera extracts features and calculates OF as described in Appendix. White noise is added to the IMU, inclinometer, camera and velocity from GNSS sensor data by the multivariate normal random noise-function, \texttt{mvnrnd}, in Matlab. Inclinometer measurements are denoted \( \Theta^b_{\text{incl}} = [\phi, \theta]^T \). The following mean and covariance are used:

\[
\begin{align*}
\nu_{\text{imu}} & \sim \mathcal{N}(0_{3 \times 1}, \Sigma_{\nu_{\text{imu}}}) = (0.135^\circ/s)^2 I_{3 \times 3} \\
\omega_{\text{imu}} & \sim \mathcal{N}(0_{3 \times 1}, \Sigma_{\omega_{\text{imu}}}) = (1.29 \cdot 10^{-3} g)^2 I_{3 \times 3} \\
\nu_{\text{incl}} & \sim \mathcal{N}(0_{2 \times 1}, \Sigma_{\nu_{\text{incl}}}) = (0.18^\circ)^2 I_{2 \times 2} \\
\nu_{\text{GNSS}} & \sim \mathcal{N}(0_{3 \times 1}, \Sigma_{\nu_{\text{GNSS}}}) = (0.21 m/s)^2 I_{3 \times 3}
\end{align*}
\]

No bias on the accelerometer is assumed, and a constant bias is assumed on the gyroscope. The gyroscope is simulated with the following bias

\[
\hat{b}^b_{\text{gyro}} = \begin{bmatrix}
0.1^\circ/s \\
-0.3^\circ/s \\
-0.35^\circ/s 
\end{bmatrix}
\]
White noise is also added to the OF data from the simulated camera. Every extracted feature is given white noise with variance, $\sigma^2_{dr} = \sigma^2_{ds} = \sigma^2_{d} = (4.5 \cdot 10^{-5})^2$. As two corresponding features are needed to get an OF vector, the resulting noise of the OF vector has variance $\sigma_{OF}^2 = \sigma_d^2 I_{2 \times 2}$. On a camera chip with 1600 × 1200 pixels and dimension 7.2 × 5.4 mm, this would yield a small variance of $(\sqrt{2} \cdot 0.01 \text{px})^2$ for the OF vector noise.

C. Terrain Simulation

In order to evaluate the performance of the GTOF and CEOF observer representations with a realistic environment, a terrain model is generated. The terrain model is a matrix, $Z$, with values corresponding to the elevation profile of the terrain. It is also called the elevation profile of the terrain, as it describes the elevation of the terrain. The terrain model is made to mimic a coastline, and has a resolution of 1m × 1m meter. The covered area is 1km × 1km. At position $x, y$ of the matrix the elevation $h$ of the terrain at $x$ meters North and $y$ meters East is found. A point on the surface of the terrain will have NED coordinate $x, y, -h$. Fig. 11 displays the simulated UAV path and the terrain model.

D. Results

Fig. 12 shows the crab angle error and the flight path angle error in the measured normalized body-fixed velocity from CV. It can be seen that the GTOF fails to produce correct measurement of the body-fixed velocity when the terrain is non-planar (at time 110-220 seconds). Any crab and flight path angle of the UAV causes NoCV to fail as it assumes pure forward motion.

Fig. 13 and Fig. 14 show the attitude estimates and the error in the estimates. The NoCV observer fails to produce accurate estimates of the attitude. It is seen that the accuracy of the GTOF observer is heavily reduced when flying over the non-planar area. The CEOF observer on the other hand is not limited by the rugged terrain, and provides accurate estimates during the entire flight. The estimated and real gyro bias is displayed in Fig. 15. It is seen that the bias values from NoCV does not converge to the correct value. The shortcomings of the GTOF observer is again illustrated when the UAV flies over the non-planar area.

Fig. 16 and 17 show the real and estimated velocity and position. The estimates are close to the reference and quite similar for GTOF and CEOF. This is expected as the velocity and position measurements from GNSS have the largest influence on these estimates.

Table 2 provides numerical evaluation of the observers in means by the Root Mean Squared (RMS) error. The CEOF observer has lower RMS in the estimates of the attitude than the GTOF observer. NoCV has the least accurate estimates in attitude, and is outclassed by CEOF. There are no major differences in estimated position and velocity. However CV seem to slightly increase the accuracy in estimated position. The estimated gyro bias is most accurate with the CEOF and least accurate with NoCV. The crab angle and flight path angle error are reduced significantly with CV. This is because NoCV assumes zero crab and flight path angle, which is not the case.
Overall the CEOF observer proves to be much more reliable than GTOF and NoCV, with a robust and accurate performance. The GTOF performs better than NoCV, which supports the use of CV in the observer. However the validity of the GTOF observer is restricted to horizontal planar terrain, which limits the range of use in practice. CEOF is not restricted by the same limitations and thus more applicable in practice.

Table 2: RMS values for the estimated states in the different cases using the ground truth optical flow (GTOF) and the continuous epipolar optical flow (CEOF) observer representation. $\chi$ and $\gamma$ are the crab angle- and flight path angle error in the body-fixed velocity measurement from the computer vision (CV), given in degrees. The gyro bias converges after approximately 100 second, hence the RMS values of the attitude and bias is considered from 100 seconds after start.

<table>
<thead>
<tr>
<th>Nonlinear Observer</th>
<th>NoCV</th>
<th>GTOF</th>
<th>CEOF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roll ($^\circ$ RMS)</td>
<td>1.1255</td>
<td>0.48963</td>
<td>0.16426</td>
</tr>
<tr>
<td>Pitch ($^\circ$ RMS)</td>
<td>1.2942</td>
<td>0.3906</td>
<td>0.15134</td>
</tr>
<tr>
<td>Yaw ($^\circ$ RMS)</td>
<td>16.1414</td>
<td>4.9359</td>
<td>0.31014</td>
</tr>
<tr>
<td>$p_{b/n}^n$ (m RMS)</td>
<td>9.5084</td>
<td>6.9281</td>
<td>6.9506</td>
</tr>
<tr>
<td>North (m RMS)</td>
<td>4.4114</td>
<td>4.2439</td>
<td>4.2623</td>
</tr>
<tr>
<td>East (m RMS)</td>
<td>1.2397</td>
<td>0.79603</td>
<td>0.80051</td>
</tr>
<tr>
<td>Down (m RMS)</td>
<td>0.63969</td>
<td>0.1526</td>
<td>0.15025</td>
</tr>
<tr>
<td>$v_{b/n}^n$ (m/s RMS)</td>
<td>0.32982</td>
<td>0.10317</td>
<td>0.084081</td>
</tr>
<tr>
<td>North (m/s RMS)</td>
<td>0.1619</td>
<td>0.15283</td>
<td>0.15257</td>
</tr>
<tr>
<td>East (m/s RMS)</td>
<td>0.029879</td>
<td>0.018871</td>
<td>0.0054807</td>
</tr>
<tr>
<td>Down (m/s RMS)</td>
<td>0.063454</td>
<td>0.021199</td>
<td>0.0062694</td>
</tr>
<tr>
<td>$b_{gyro}^b$ ($^\circ$/s RMS)</td>
<td>0.095493</td>
<td>0.10302</td>
<td>0.0088266</td>
</tr>
<tr>
<td>Roll ($^\circ$/s RMS)</td>
<td>12.1701</td>
<td>4.7667</td>
<td>0.46633</td>
</tr>
<tr>
<td>Pitch ($^\circ$/s RMS)</td>
<td>6.8836</td>
<td>2.2066</td>
<td>0.16025</td>
</tr>
<tr>
<td>Yaw ($^\circ$/s RMS)</td>
<td>6.8836</td>
<td>2.2066</td>
<td>0.16025</td>
</tr>
</tbody>
</table>

VII. Conclusions

In this paper two different vision-aided nonlinear observers, and one nonlinear observer without CV, for estimation of position, velocity and attitude, have been evaluated on real experimental data obtained by flying a fixed-wing UAV with a custom-made payload of sensors. The nonlinear observers have also been tested on simulated data to compare the performance of the observers in the presence of non-planar terrain and with an exact known reference for comparison. The results show that using CV increases the accuracy of the nonlinear observer, especially in estimated attitude. This is because CV provides useful information about the direction of the body-fixed velocity. Furthermore the CEOF nonlinear observer has shown to be a more robust option than the GTOF nonlinear observer, as it is terrain independent.

Acknowledgements

The authors are grateful for the assistance provided by the UAV engineers at NTNU and Maritime Robotics AS, in particular Lars Semb and Carl Erik Stephansen. Significant contributions to the construction of the UAV payload was made by the rest of the navigation team at NTNU, in particular Sigurd M. Albrektsen, Jakob M. Hansen and Kasper T. Borup.

References

Figure 12: Error in crab (\(\tilde{\chi}\)) and flight path angle (\(\tilde{\gamma}\)) for the measured normalized body-fixed velocity.

Figure 13: Estimated attitude. When the UAV flies over the rugged terrain, the GTOF observer fails to produce correct estimates of the attitude.

Figure 14: Error in estimated attitude. When the UAV flies over the rugged terrain, the GTOF observer fails to produce correct estimates of the attitude. The NoCV observer fails to estimate correct pitch angle.

Figure 15: Estimated gyro bias together with the real gyro bias. After 100 sec the gyro bias has converged. When flying over the rugged terrain the GTOF observer produces erroneous gyro bias estimates, while the CEOF observer is unaffected.

Figure 16: Estimated velocity.

Figure 17: Estimated position.


Appendix

Parameter Projection

The parameter projection $\text{Proj}(\cdot, \cdot)$ is defined as:

$$\text{Proj}(\hat{b}, \tau) = \begin{cases} 
\left( I - \frac{c(\hat{b})}{\|\hat{b}\|^2} \hat{b} \hat{b}^T \right) \tau, & \|\hat{b}\| \geq L_b, \quad \hat{b}^T \tau > 0 \\
\tau, & \text{otherwise}
\end{cases}$$

where $c(\hat{b}) = \min \{1, (\|\hat{b}\|^2 - L_b^2)/(L_b^2 - L_b^2)\}$. This operator is a special case of that from Appendix E of [54].
Consider only one of the FOV features, and denote this FOV feature at time \( t \) as \( p \). Let the displacement of a projected point in the image plane between time \( t_k \) and \( t_k+1 \), that is \( dr = r(t_k) - r(t_k-1) \) and \( ds = s(t_k) - s(t_k-1) \). Lets first consider how one can choose features to project given the UAVs attitude, position and a elevation profile of the terrain. These features are the one that one wish to find the OF of.

At a time \( t_k \) a ray is drawn in the camera z-axis as shown in Figure 18. The ray intersects the ground plane at a point \( t^n_{\text{centre}} = [x^n_{\text{centre}}, y^n_{\text{centre}}, 0]^T \) or expressed in \( \{C\} \) \( t^n_{\text{centre}} = (T^n_{c})^{-1}x^n_{\text{centre}}, \ T^n_{c} \) being the homogeneous transformation matrix relating \( \{C\} \) and \( \{N\} \). The point \( t^n_{\text{centre}} \) is named the "centre ground point".

Points are chosen deterministically around the centre ground point, \( t^n_{\text{centre}} \), distributed on a plane perpendicular to the ray from \( \{C\} \) to the centre ground point. This plane is named the field of view (FOV) plane. The points are distributed on the FOV plane, ranging from the centre ground point \(-30 \) to \( 30 \) meters in camera x-direction \(-40 \) to \( 40 \) meters in camera y-direction, separated with \( 10 \) meters in both dimensions. Lets call these points "FOV features" and denote them by \( p^n_{\text{FOV}} \). The FOV features in camera coordinates is then defined as \( p^n_{\text{FOV}} \in t^n_{\text{centre}} + [x, y, 0]^T, x \in [-30, -20, \ldots, 20, 30], y \in [-40, -30, \ldots, 30, 40] \). Lets now consider only one of the FOV features, and denote this FOV feature \( p^n_{\text{FOV}} \).

The FOV feature \( p^n_{\text{FOV}} \) is transformed to \( \{N\} \) by \( p^n_{\text{FOV}} = T^n_{c}p^n_{\text{FOV}} \). Let the FOV feature be defined as \( p^n_{\text{FOV}} = [x^n_{\text{FOV}}, y^n_{\text{FOV}}, z^n_{\text{FOV}}] \). The FOV feature is then projected onto the terrain by using \( x^n_{\text{FOV}}, y^n_{\text{FOV}} \) and the elevation \( h \) at the \( x^n_{\text{FOV}}, y^n_{\text{FOV}} \) coordinate of the elevation profile. The projected point is then \( p^n = [x^n_{\text{FOV}}, y^n_{\text{FOV}}, h]^T \), which is called a "feature".

Now that the feature location in \( \{N\} \) is found, it is in our interest to find the projection of this feature at time \( t_k \) and \( t_k-1 \). The camera moves between \( t_k-1 \) and \( t_k \), meaning the homogeneous transformation matrix \( T^n_{c} \) is time variant. The feature can then be transformed to \( \{C\} \) by \( p^n = (T^n_{c})^{-1}(t_k)p^n \) and \( p^n = (T^n_{c})^{-1}(t_k-1)p^n \). The points \( p^n = (T^n_{c})^{-1}(t_k)p^n \) represents the feature on the surface of the terrain given in camera coordinates at time \( t_k \) and \( t_k-1 \) respectively.

The feature at time \( t_k-1 \) and \( t_k \) can then be projected onto the image plane by the pinhole camera model from (1), yielding \( x^n(t_k) = [r(t_k), s(t_k)]^T \) and \( x^n(t_k-1) = [r(t_k-1), s(t_k-1)]^T \). The discrete OF can then be found as \( dr = r(t_k) - r(t_k-1) \) and \( ds = s(t_k) - s(t_k-1) \).

Fig. 18 illustrates the relationship between the "centre ground point" \( t^n_{\text{centre}} \), "FOV features" \( p^n_{\text{FOV}} \), and "features" \( p^n \).

![Figure 18: Features on the surface of the terrain are chosen based on the attitude and position of the UAV. A ray along the camera z-axis intersects the ground plane at \( t^n_{\text{centre}} \). A plane denoted field of view (FOV) is constructed perpendicular to the ray. FOV features are distributed along the FOV plane. Features are constructed with \( z^n \)-component from the elevation profile and \( x^n, y^n \) coordinate from the corresponding FOV feature. Features \( p^n \) are projected onto the image plane by the pinhole camera model to find the image plane coordinate \( x^n = [r, s] \). This is done at time \( t_k \) and \( t_k+1 \) with the same features, \( p^n \), to get the discrete OF \( dr \) and \( ds \).]