A Lightweight Thermal Camera Payload with Georeferencing Capabilities for Small Fixed-Wing UAVs

Frederik S. Leira, Kenan Trnka, Thor I. Fossen and Tor Arne Johansen

Abstract — This paper discusses the design and implementation of a lightweight thermal camera payload for small fixed-wing UAVs with a pan/tilt mechanism and georeferencing based on a simple autopilot's GPS and IMU. Further, an effective and simple method to accurately calibrate a thermal camera is developed and tested. In order to evaluate the accuracy of the georeferencing algorithm, a test flight was conducted. Georeferencing 80 images of an object showed that the average of the 80 images were only 1.6 m from the object's actual position when operating at 50 – 100 m altitude above ground. Although the average georeferencing error was estimated to be around 7 m, this accuracy is considered sufficient for most of the intended object tracking and surveillance applications.

I. INTRODUCTION

In recent years, the increased availability of small light-weight unmanned aerial vehicles (UAVs) has lead to the use of UAVs in many different applications involving inspections of structures, surveillance and tracking of objects. Furthermore, it is seen that more often than not the UAV will also have an on-board camera as visual information can be valuable to the UAV operator. In some cases the visual information can also be processed on-board and be directly involved in the UAV's decision making. This is especially apparent in applications such as autonomous object detection and tracking [1][2]. In this regard, the need to be able to extract the coordinates of interesting pixels in a captured image (georeferencing) has lead to a lot of research on georeferencing in UAVs, e.g [3], [4], [5], [6], [7].

In [5] and [6] two very accurate georeferencing algorithms for UAVs are described. However, [5] requires the use of ground reference points (GCPs) in order to determine the coordinates of pixel coordinates. This is very efficient in rural areas and other environments with recognizable landmarks, but it is for instance not possible to use this technique in maritime environments such as the one described in [1]. Results from [6] indicate an accuracy of 10 cm by using an iterative version of aerial triangulation (AT). However, in addition to having a big computational requirement not suitable for small fixed-wing UAVs, it requires corresponding points of ground features between sequential images. As with [5] this works well in environments containing textures and features, but will lose its efficiency in e.g maritime environments where distinct landmark features are scarce.

[3] describes a real time automatic georeferencing system which does not require the use of GCPs or distinct landmark features. The georeferencing is achieved by synchronizing UAV attitude and altitude measurements with the camera images. Using a RTK-GPS, a high-end IMU and a high resolution camera they are able to achieve a georeferencing error of only 90 cm. However, the camera is mounted in a fixed position, which greatly reduce the flexibility of the system when mounted in a fixed-wing UAV. Furthermore, the total equipment weight used in their UAV is heavy and not suitable for a small fixed-wing UAVs.

[4] describes a complete system for online georeferencing which in addition to not requiring any GCPs also allows for free movement of the camera relative to the UAV. The results and accuracy of the system is promising, but no quantitatively analysis is performed. Hence it is not possible to say anything specific with regard to application specific performance. For instance, the accuracy...
of georeferencing the position of an object pixel location is not known. [4] also propose a novel method for automatic camera calibration. However, this method is not compatible with thermal cameras as it depends on the localization of certain patterns on a printed sheet of paper. This pattern will not show up clearly on a thermal video feed as the paper has an uniform temperature.

The most closely related prior work to our system is described in [7]. They perform georeferencing of a tracked object, and achieve a similar precision on georeferencing as we do in our work. However, their system does not seem to account for the distortions that occurs in most commercial cameras.

The results on georeferencing in the literature emphasize the need to adequately calibrate the on-board camera to increase the georeferencing accuracy. Several suggested solutions for calibration of thermal cameras exist, but often requires either a complex computer vision algorithm ([8]) or a complex mechanical construction ([9], [10]).

In this paper we develop a light-weight payload and an algorithm based on the work of [3] and [4] which is suitable for online georeferencing of images captured on-board UAVs. One of the main advantages of this method compared to results available in the literature is that the camera orientation and position with respect to the UAV are not restricted. This allows the camera to be controlled by an on-board gimbal while the system is still able to georeference pixel coordinates. Furthermore, its focus is that of localization, i.e finding the coordinates of an object based on its pixel location. This is uncommon, as most such system focus on creating mosaics of large areas and placing entire images in a geocentric coordinate frame. The proposed algorithm is not dependent on the determination of GCPs in order to be accurate, as the algorithm use on-board autopilot data (UAV attitude and altitude) to determine in what location and pose an image was captured. In addition, a novel, simplistic method for calibration of thermal cameras is proposed.

The remainder of this paper is organized as follows. First, the payload system components and software are described in detail. Following this, the theory required to perform georeferencing is presented, in addition to some theory on camera calibration and synchronization of data. The accuracy of the georeferencing algorithm is then tested on data gathered with the payload during an UAV flight test. Finally the paper is summarized in a conclusion.

II. PAYLOAD DESCRIPTION

The components of the payload and their interconnections can be seen in Figure 1. The payload includes an autopilot (PixHawk ArduPilot [11]) for flight control, a single board computer (Beaglebone Black [12]) for on-board UAV control and video recording and an analog thermal camera (FLIR Tau2 640 [13]) as a visual sensor. The

![Fig. 1. Overall payload system description. DUNE is the on-board control software.](image-url)

thermal camera has a sampling frequency of 9 frames per second and is sensitive to the long-wave infrared spectral band (7.5 – 13.5µm) with a sensitivity < 50 mK. It has a resolution of 640 × 480 pixels, which is interpolated up to 704 × 480 pixels. Furthermore, to capture the video from the thermal camera there is also an analog to digital converter (Axis M7001 [14]) that converts the analog thermal video feed to a digital video stream. The video stream can also be broadcasted to the ground station by the payload communication link (Ubiquiti Rocket M5 [15]). Finally, the thermal camera is mounted in a retractable gimbal (R-BTC88 [16]) which can rotate along the pan and tilt axes. The payload (including the gimbal) mounted in the airframe of the fixed-wing UAV can be seen in Figure 2.

On the single board computer, the on-board UAV control software DUNE is running. The
Fig. 2. The X8 Skywalker fixed-wing UAV interfaced with a retractable gimbal.

DUNE Uniform Navigation Environment (DUNE) [17] is composed of independent tasks that communicate by sending and subscribing to specific messages (sensor measurements, actuator control etc.). Hence, DUNE is also the software which is used to command the gimbal position. DUNE does this by communicating with the autopilot, which in turn sends the appropriate control signal to the gimbal. The autopilot is running an Extended Kalman Filter [18] (EKF) in order to have filtered estimates of the UAV’s altitude and attitude. The output from this filter is communicated to the single board computer and stored on-board. The autopilot has a sampling rate on telemetry data of 50Hz.

III. GEOREFERENCING

A. Georeferencing Theory

In this paper we seek to find an object’s location in a North-East-Down (NED) frame based on its pixel coordinates in a thermal image. The following approach to this problem is based on the work found in [3] and [4].

Now, before we are able to compute anything in a NED frame, a reference point for the NED frame has to be set using geographic (latitude and longitude) coordinates. To define a NED frame at a reference geographic position, $\lambda_{ref}$ (longitude), $\phi_{ref}$ (latitude) and $h_{ref}$ (height), we first find the ECEF coordinates of the position using the following equations[19]

$$
X^e = (N + h)\cos(\phi_{ref})\cos(\lambda_{ref})
$$

$$
Y^e = (N + h)\cos(\phi_{ref})\sin(\lambda_{ref})
$$

$$
Z^e = [N(1 - e^2) + h_{ref}]\sin(\phi_{ref})
$$

$$
N = \frac{a}{\sqrt{1 - e^2\sin^2(\phi_{ref})}}
$$

$$
f = \frac{(a - b)}{a}
$$

$$
e^2 = 2f - f^2
$$

$(X^e, Y^e, Z^e)$ is a position in the ECEF frame, $a$ is the semi-major earth axis (6378137m) and $b$ is the semi-minor earth axis (635752.3142m). We then apply the following rotation to the ECEF coordinates in order to align it with the North, East and Down axis[19]

$$
R_{ecef}^{ned} =
\begin{bmatrix}
-s\lambda_{ref}c\phi_{ref} & -s\lambda_{ref}s\phi_{ref} & c\lambda_{ref} \\
-s\phi_{ref} & c\phi_{ref} & 0 \\
-c\lambda_{ref}c\phi_{ref} & -c\lambda_{ref}s\phi_{ref} & -s\lambda_{ref}
\end{bmatrix}
$$

where $s$ and $c$ are the trigonometric functions sin and cos.

Now, to find the georeferenced location of an object (in our case the object’s location in the NED frame), the image frame coordinates of the object’s center is calculated. Using these coordinates, combined with the UAV’s attitude and altitude data and the gimbal camera orientation, the location of the object in NED frame coordinates can be related to the image coordinates as follows[3]

$$
\begin{bmatrix}
p'_{u} \\
p'_{v} \\
\lambda
\end{bmatrix} = AG
\begin{bmatrix}
N_{obj} \\
E_{obj} \\
D_{obj}
\end{bmatrix}
$$

where

$$
G = \begin{bmatrix}
R_{camera}^{ned} & -R_{camera}^{ned}\mathbf{C}
\end{bmatrix}
$$

Here, $\lambda$ is a scaling factor; $p'_{u}$ and $p'_{v}$ define the scaled object location in the image frame (pixel coordinates) and $N_{obj}$, $E_{obj}$ and $D_{obj}$ are the NED frame coordinates of the object. In order to find the non-scaled (actual) pixel coordinates $(p_u, p_v)$
we first solve for the left side of (3). Then, $\lambda$ is used to calculate the normalized homogenous form of the image coordinates that was found by solving (3), i.e

$$
\begin{bmatrix}
p_u \\
p_v \\
1
\end{bmatrix} = \begin{bmatrix}
\frac{D}{\lambda} \\
\frac{D}{\lambda} \\
1
\end{bmatrix}
$$

(5)

Furthermore, $A$ is the intrinsic parameter matrix of the camera and $G$ is the extrinsic orientation parameters of the camera (rotation and translation). $C$ is the position of the camera given in the NED frame\cite{4}, i.e

$$
C = \begin{bmatrix}
N_{uav} \\
E_{uav} \\
D_{uav}
\end{bmatrix} + R_{\text{body}}^{\text{ned}} \begin{bmatrix}
X_{GB} \\
Y_{GB} \\
Z_{GB}
\end{bmatrix}
$$

(6)

where $N_{uav}, E_{uav}, D_{uav}$ define the UAV’s position in the NED frame, and $X_{GB}, Y_{GB}, Z_{GB}$ is the gimbal position in the body frame. The relation between the coordinates systems are illustrated in Figure 3.

The rotation matrix $R_{\text{ned}}^{\text{camera}}$ can easily be calculated from the following equation

$$
R_{\text{ned}}^{\text{camera}} = (R_{\text{body}}^{\text{ned}})^{-1} \quad R_{\text{ned}}^{\text{camera}} = \begin{bmatrix}
r_1 & r_2 & r_3
\end{bmatrix}
$$

(7)

where $R_{\text{ned}}^{\text{body}}$ is defined as \cite{19}

$$
R_{\text{body}}^{\text{ned}} = \begin{bmatrix}
\cos \phi \cos \theta & -\sin \phi \cos \psi & \sin \phi \\
\cos \phi \sin \theta & -\sin \phi \sin \psi & \cos \phi \\
-\sin \theta & \cos \theta & 0
\end{bmatrix}
$$

(8)

\(\theta, \phi\) and \(\psi\) is the roll, pitch and yaw, respectively, of the aircraft. Furthermore, letting 0\(^\circ\) pan and 0\(^\circ\) tilt be the position in which the gimbal (hence also the camera) is pointing directly downwards, the rotation matrix describing the rotation between the body frame and the mounted frame ($R_{\text{body}}^{\text{mount}}$) is equal to a 90 degrees clockwise rotation about the z-axis. That is,

$$
R_{\text{body}}^{\text{mount}} = \begin{bmatrix}
0 & 1 & 0 \\
-1 & 0 & 0 \\
0 & 0 & 1
\end{bmatrix}
$$

(9)

Now, in the mounted frame, the gimbal’s pan and tilt movement corresponds to a rotation along the mounted z-axis and the mounted x-axis, respectively. Given a gimbal position $\psi_{gb}$ (pan) and $\phi_{gb}$ (tilt), we have the following rotation matrix to relate the mounted frame to the camera frame.

$$
R_{\text{mount}}^{\text{camera}} = \begin{bmatrix}
c_{\psi_{gb}} c_{\phi_{gb}} & s_{\psi_{gb}} & 0 \\
-s_{\psi_{gb}} c_{\phi_{gb}} & c_{\psi_{gb}} c_{\phi_{gb}} & s_{\phi_{gb}} \\
s_{\phi_{gb}} s_{\psi_{gb}} & -s_{\psi_{gb}} c_{\phi_{gb}} & c_{\phi_{gb}}
\end{bmatrix}
$$

(10)

Finally, in order to find $R_{\text{camera}}^{\text{body}}$ we use the following relation

$$
R_{\text{body}}^{\text{camera}} = (R_{\text{mount}}^{\text{camera}})^{-1} = (R_{\text{mount}}^{\text{camera}} R_{\text{body}}^{\text{mount}})^{-1}
$$

(11)

Now, we will assume, without loss of generality, that every object is located on the ground surface. This implies that $D_{\text{obj}}$ in (3) will be equal to 0. This yields the following relationship between the object’s image frame coordinates $(p_u, p_v)$ and its NED coordinates $(N_{\text{obj}}, E_{\text{obj}}, 0)$

$$
\lambda \begin{bmatrix}
p_u \\
p_v \\
1
\end{bmatrix} = AG_{NE} \begin{bmatrix}
N_{\text{obj}} \\
E_{\text{obj}} \\
1
\end{bmatrix}
$$

(12)

where

$$
G_{NE} = \begin{bmatrix}
r_1 & r_2 & -R_{\text{ned}}^{\text{camera}} C
\end{bmatrix}
$$

(13)

$\lambda$ is still a scaling factor as defined in (3) and (5). If, in general, $D_{\text{obj}} \neq 0$, the system can be
adapted to use electronic maps. This would involve a search for the intersection between $D_{obj}$ and the ray of light going from the object to the camera center. This would increase the complexity of the calculation by a small amount.

From (12) it is readily seen that if the pixel coordinates of an object $(p_u, p_v)$ is known, the object’s position in the NED frame can be found by the following calculation

$$\frac{1}{\lambda} \begin{bmatrix} N_{obj} \\ E_{obj} \\ 1 \end{bmatrix} = G_{NE}^{-1} A^{-1} \begin{bmatrix} p_u \\ p_v \\ 1 \end{bmatrix}$$

(14)

B. Camera Calibration and Distortion Model

From (3) it is seen that in order to relate image pixels to NED coordinates, one has to use the camera intrinsic parameter matrix $A$. $A$ is completely dependent on the camera characteristics, and is defined as

$$A = \begin{bmatrix} k_u f & 0 & u_0 \\ 0 & k_v f & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

(15)

where $u_0$, $v_0$ is the principal point of the camera, $k_u$, $k_v$ are the pixel sizes along the two image axes and $f$ is the focal length of the camera. This matrix can be found in two ways. The first is to use the camera specification sheet to find the focal length ($f$) of the lens, the image sensor size ($w \times h$) of the camera and its pixel resolution ($u \times v$). The following is then an estimate for $A$

$$\hat{A} = \begin{bmatrix} \frac{u}{w} f & 0 & \frac{u}{w} \\ 0 & \frac{v}{h} f & \frac{v}{h} \\ 0 & 0 & 1 \end{bmatrix}$$

(16)

The reason that this is only an estimate of $A$ is that the manufacturer of the camera will not be able to make the camera exactly as specified. E.g. the camera’s principal point might be off by a small margin, making $(\frac{u}{w}, \frac{v}{h})$ a poor estimate for the principal point.

The camera intrinsic parameter matrix is based on an ideal pinhole camera model. In reality, this is not correct for actual cameras. This results in distortions both along the radial and tangential direction of the image frame, hence this needs to be corrected. In order to correct for distortion, we will use the following simplified model for the distortion in a camera [20]

$$u_d = u_u (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2 k_3 u_u v_u r^2 + 2 u_u^2)$$

$$v_d = v_u (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + k_3 (r^2 + 2 v_u^2) + 2 k_4 u_u v_u$$

(17)

Here, $u_d, v_d$ is the distorted image point as projected in the camera frame by the camera lens, while $u_u, v_u$ is the undistorted image point as projected by an ideal pin-hole camera. That is, $u_d, v_d$ is the position in the camera frame that we observe, while $u_u, v_u$ is the (undistorted) position that we want to find. $k_1, k_2, k_3$ are radial distortion coefficients, while $k_3$ and $k_4$ are tangential distortion coefficients. In order to incorporate this distortion model into our georeferencing equations, we now want to use the following mapping between the camera frame and the NED frame [3]

$$\lambda \begin{bmatrix} u_u \\ v_u \\ 1 \end{bmatrix} = G_{NE} \begin{bmatrix} N_{obj} \\ E_{obj} \\ 1 \end{bmatrix}$$

(18)

Here, $\lambda$ is still a scaling factor. Note that since we are mapping NED positions to the camera frame (and not the image frame), we do not need to use the camera’s intrinsic parameter matrix. Now, in order to find the undistorted camera frame coordinates $(u_d, v_d)$ of an object given its pixel location $(p_u, p_v)$ in an image, we use the following relation

$$\begin{bmatrix} u_d \\ v_d \\ 1 \end{bmatrix} = A^{-1} \begin{bmatrix} p_u \\ p_v \\ 1 \end{bmatrix}$$

(19)

Having found the object’s distorted position $(u_d, v_d)$, we now solve for $(u_u, v_u)$ in (17). This is non-trivial because of the non-linearity of the problem, and the reader is referred to [21] for a more in-depth take on the solution.

Having found the undistorted object coordinates $(u_u, v_u)$ in the camera frame, we can easily solve for the NED position of the object using

$$\frac{1}{\lambda} \begin{bmatrix} N_{obj} \\ E_{obj} \\ 1 \end{bmatrix} = G_{NE}^{-1} \begin{bmatrix} u_u \\ v_u \\ 1 \end{bmatrix}$$

(20)
In order to find a better estimate of $A$ than (16), in addition to estimating the distortion co-
efficients for the thermal camera, we go through a process referred to as camera calibration [22]. In our work, the OpenCV Calibration Tool[23] was used in order to calibrate the camera. Usually the calibration is performed by using a chessboard pattern on a sheet of paper and capturing images of this pattern with the camera from several different angles. However, this is difficult to accomplish with a thermal camera, as the chessboard pattern will not show up clearly on the paper because the temperature of the paper is close to uniform. To make the calibration process more in line with that of calibrating a regular camera, a 2mm thick plate with $8 \times 7$ 3mm wide holes 2cm apart from each other was made. After cooling the plate down to $\approx 0^\circ C$, several images of the plate from different angles and positions were taken in room temperature ($\approx 24^\circ C$). An example image is shown in Figure 4. The thermal camera was configured to be sensitive to the temperature range $5^\circ - 15^\circ$, effectively creating an almost binary like image, hence making the plate pattern easily detectable. The result (estimated $A$ matrix and distortion coefficients) from the calibration process can be seen in Figure 5.

C. Synchronization

Ideally, time synchronization of images, GPS and IMU data should be done directly in the hardware and not during post processing. However, since the thermal camera has an analog output, it is not possible to directly control the shutter of the camera. This results in that the exact time stamp of an image is not available, and has to be estimated. Since the thermal video stream coming from the thermal camera has 9 FPS, it was assumed that between each image frame the time elapsed is exactly $\frac{1}{9}$ s. This means that synchronization of the thermal video data and the telemetry data is a matter of deciding the initial offset $\Delta t$ between an image frame and the telemetry data.

In order to manually decide the time delay between the video and telemetry data, the time stamp of the command to retract the gimbal in the autopilot’s log was correlated with the time stamp of the beginning of the retract motion in the thermal video. Although the assumption that the time between each captured image frame is constant might not be a perfect model for the time delay, the time delay was found to be consistent throughout the video. This was ensured by finding the offset between the video and the telemetry data for several retract/deploy commands sent to the
gimbal.

Note that finding an estimate for $\Delta t$ during offline analysis enables us to expand the system to online georeferencing, as the synchronization offset will be exactly the same online as in the offline case.

IV. RESULTS

A field test of the payload was conducted in order to test the functionality of the system and the accuracy of the georeferencing algorithm. Infrared footage and telemetry data was recorded during a 30 min test flight at altitudes varying from 50m to 100m. During the flight, the gimbal was fixed to the 0° pan and 0° tilt (that is, pointing straight downwards in the body frame) position. This was done in order to evaluate the inaccuracy of the system without the gimbal before testing the system with the gimbal at a later point in time. Furthermore, the exact GPS location of the ground base antenna was also saved. This was done in order to have the ”ground truth” of an object location. Figure 6 shows an example image from the video feed during flight.

In order to georeference the GPS antenna located on the ground station, the pixel coordinates of the antenna (red square in Figure 6) were manually marked in each image in which it was visible. Now, to find the pose of the aircraft at the moment that an image was captured, the last received aircraft state (position, attitude and altitude) from the autopilot was used.

In total, 80 instances of the GPS antenna were found and marked manually from the infrared video. Using (14) in the non-calibrated case and (20) in the calibrated case, the position of the GPS antenna was georeferenced. The result from performing the georeferencing for the 80 points without correction for the distortion can be seen in Figure 7. In Figure 9 the results for georeferencing with a better estimate of the intrinsic camera matrix $A$ and distortion correction is shown. In Figure 8, a section of the flight path is shown together with the UAV’s position at the times of detecting the ground GPS antenna.

Comparing Figure 7 with Figure 9, it is readily observed that calibrating the camera has a big impact on the accuracy of the payloads georeferencing capabilities. In the non-calibrated results an average georeferencing error of 10.53m was observed, while the average of the georeferenced positions were 4.03m away from the actual position of the GPS antenna. This is not bad, especially considering that the main point of georeferencing object locations in our case is to keep the object within the field of view of the camera. Considering that the camera has a field of view of $80 \times 60m^2$ when the roll and pitch of the UAV is 0, the object...
will be well within the image frame even though the camera is pointed at the average object position instead of the actual position. However, as is seen

in Figure 7, the spread of the georeferenced points is large. With a standard deviation of 10.4m in the north direction, and 6.6m in the east direction, it is clear that using only a single observation of the object pixel location in order to perform georeferencing will be inaccurate.

Now, with a calibrated camera it is clear that

the accuracy has increased. This emphasizes the need to accurately calibrate the camera. The average projection error with the calibrated camera is 7.09m, while the average of the georeferenced positions were only 1.64m away from the actual position. Furthermore, the standard deviation has now decreased to 7.78m in the north direction and 3.96m in the east direction. This is still quite high if the goal is to determine an object’s position based on only one observation. However, as the average point suggests, the error seems to be unbiased. This means that using several observations of the same object will increase the accuracy of the position estimate.

There are many inaccuracies which can affect the performance of the georeferencing system. A small inaccuracy in the $\Delta t$ used, a small error in the camera calibration process, a small deviation from actual UAV attitude or a bad estimate of the UAV altitude are all factors that will affect the accuracy of the system. An offset of $+5^\circ$ in both the roll and pitch angle (assuming originally $0^\circ$ roll, $0^\circ$ pitch, $0^\circ$ yaw and $100m$ altitude) will for instance change the projection of pixel point $(0, 0)$ from $(31.4m, -41.85m, 0m)$ to $(42.65m, -54.28m, 0m)$ in the NED frame, i.e a change of $11m$ in the north direction and $12.5m$ in the east direction. This means that it is crucial to have a very accurate estimate of the UAV’s pose at the exact time of the image being taken.

In Figure 10, the roll and pitch flight data is shown together with the time instants of detecting the ground GPS antenna. It is seen that the roll
and pitch signals have high frequency components, making it difficult to accurately decide the roll and pitch values at any given time instant. The high frequency components also indicate a need to synchronize the telemetry data with the video data as accurately as possible. Furthermore, in Figure 11, the altitude flight data is shown together with the time instants of GPS antenna detection. Although not as time varying as the roll and pitch data, it is seen that the altitude signal is ranging from 130m to 55m, with steep transitions in between. Flying at a constant altitude might improve the georeferencing accuracy further, as it could yield a more accurate estimate of the UAV's altitude as well as making a perfect time synchronization of telemetry data and video less crucial.

Considering that we are using an affordable open source platform (cheap IMU, GPS and magnetometer) in our payload, the results are reasonable. Using for instance a RTK-GPS and hardware synchronization would indeed improve the accuracy of the system for use in applications that would require better precision.

V. CONCLUSION

This paper discusses a light-weight thermal camera based payload for small fixed-wing UAVs with georeferencing capabilities. In addition, an effective and simple method to accurately calibrate a thermal camera was developed and tested. Furthermore, the georeferencing algorithm was successfully demonstrated and its accuracy was evaluated using data from a test flight conducted using the payload described. Although it was observed that the georeferencing had a standard deviation of up to 7.8m, it is emphasized that the georeferencing is accurate enough to ensure that, if the camera is pointed at the estimated object ground position, objects of interest will be well within the image frame. In other words, the payload system described in this paper is accurate enough for applications such as object tracking and surveillance.

In future work we plan to run the georeferencing algorithm on-board during flight. Furthermore we will test the system together with the automatic detection, classification and tracking system in [1]. This will allow the UAV to autonomously follow and keep track of objects of interest located on the ground.

ACKNOWLEDGMENT

This work has been carried out at the Centre for Autonomous Marine Operations and Systems (AMOS). The Norwegian Research Council is acknowledged as the main sponsor of AMOS. This work was supported by the Research Council of Norway through the Centres of Excellence funding scheme, Project number 223254.

REFERENCES


