A UAV Ice Tracking Framework for Autonomous Sea Ice Management

Frederik S. Leira, Tor Arne Johansen, Thor I. Fossen
Centre for Autonomous Marine Operations and Systems
Department of Engineering Cybernetics
Norwegian University of Science and Technology, Trondheim, Norway
Email: frederik.s.leira@ntnu.no

***Abstract*—This paper describes an unmanned aerial vehicle (UAV) ice tracking framework for use in sea ice management applications. The framework is intended to be used in an ice management scenario where the UAV should detect and track the movement of icebergs and ice floes in an Arctic environment, and seeks to enable the UAV to do so autonomously. This is achieved by using an occupancy grid map algorithm and a locations of interest generator coupled with a Model Predictive Control (MPC) UAV path planner. The main contribution of this paper is interfacing the occupancy grid map algorithm with a machine vision object detection module in order to enable the UAV to generate an occupancy grid map of a pre-defined search area in real-time using on-board processing of UAV sensor data. Further, the paper presents a locations of interest generator module which generates locations that the UAV should investigate based on the generated occupancy grid map. These locations of interest are then used by an MPC path planner in order to make the UAV autonomously investigate and track ice features at said locations. Furthermore, the paper verifies the use of the developed ice tracking framework for autonomously detecting and tracking ice features based on thermal images captured with a UAV, as well as verifying the usefulness and role of UAVs in ice management scenarios by conducting two flight experiments.**

I. Introduction

In order to conduct marine operations in ice-infested regions in a safe manner, accurate information about the environment is needed. E.g, when operating offshore installations such as oil rigs, it is often important to be able to detect and track icebergs close to these offshore installations with sufficient confidence and in a timely manner [1]. Further, using the Northern Sea Route for shipping activities are of interest due to the possible favourable economic properties, as it may drastically reduce the travelling distance required for the shipping [2]. However, when doing so, an ice management system is needed in order to keep the vessel safe while travelling through Arctic waters. Ice management is here defined as the sum of all activities where the objective is to reduce or avoid actions from any kind of ice features [3]. This could for instance be the detection, tracking and forecasting of sea ice, ice ridges and icebergs. Note that in the two ice management scenarios described above, real-time processing of sensor data could be useful in order to achieve timely detection of nearby ice threats to the offshore platform or shipping vessel is crucial.

In recent years there has been a drastic increase in the commercial availability of small unmanned aerial vehicles (UAVs) which demonstrate flexibility in geographical coverage, spatial and temporal resolution. These are all abilities important in sensor platforms which are to be used in ice management scenarios [4]. The use of aerial sensors and UAV platforms in ice management has gained a lot of interest in recent years, and some proposed applications and solutions already exists. [4] presents a possible structure for a general ice observer/ice management system. The presented ice observer system structure is made to be able to collect, analyse and employ ice intelligence during operations in ice management, and UAVs are presented as a viable sensor platform for these operations. Moreover, it illustrates how UAVs can be equipped with sensors that provide information about the environment, including wind velocity, iceberg and ice floe distribution and ice velocity. This is also demonstrated in [5] where a small UAV is equipped with a camera and an ice monitoring experiment is conducted. Using a simple thresholding technique, images from a UAV flight is segmented into ice and non-ice regions, effectively demonstrating the UAV's usefulness in ice management settings.

[6] presents an algorithm that is able to use stereoscopic optical imaging from UAVs in order to image sea ice surface 3D structure. The algorithm uses structure from motion [7] in order to generate a 3D model of the sea ice structure, and the resulting model can be utilized in the process of finding a safe passage for a boat through a region with thick ice. Structure from motion is also used in [8] to generate 3D models of a glacier by the use of image data capture with a UAV. Although the structure from motion algorithm is too computationally expensive to be executed on-board and in real-time, this work further proves that the UAV is a viable and useful platform in the area of ice management.

[9] and [10] both presents path planning algorithms for UAVs in ice management based on optimization techniques. [9] develops an algorithm to find a path pattern that covers a predefined region of interest. The region is chosen based on the location of a boat and an ice flow estimate, yielding a region where potential icebergs and ice floes headed directly towards the boat are expected to be found. The region is divided into a grid of cells and the UAV's visitation sequence for the grid cells is found by solving a Mixed Integer Linear

Programming (MILP) problem seeking to minimize the time required to cover the total grid. [10] assumes that a set of n icebergs have been detected and located, and that the UAV is given an estimate of each of the iceberg's positions. It is further assumed that the uncertainty of the position estimate for the location of each iceberg is reset to 0 when the UAV is in its vicinity and able to observe its position. A MILP problem is then defined and solved, yielding a visitation sequence that minimize a weighted cost function of total distance travelled by the UAV, and the total uncertainty in the position estimates of the icebergs. Both of these algorithms present useful applications of UAVs in the area of ice management, but does not describe the remainder of the components, such as a detailed description on incorporating sensor measurements into their solutions, that would be necessary in order to have a completely autonomous ice management system for UAVs.

[11] presents an ice concentration and flow monitoring system that is able to create feasible and collision-free paths for a set of UAVs, where the paths are generated in such a way as to minimize the uncertainty in the state estimate of the ice flow. The problem is formulated as an optimal control problem which use a kinematic model for UAVs and a finite element discretization of a uniformly drifting sea ice concentration field. The system's measurements are segmented images of the sea ice captured from a camera on-board the UAVs, which then maps the concentration of pixels segmented as sea ice onto a tessellated map in world coordinates. The approach is very useful for ice management, as it can be used to perform automated mapping of sea ice and ice flow in a predefined region. However, simulations show that the algorithm becomes increasingly computationally demanding as the search region increases in size. A way to deal with the scalability issue would have to be implemented in order to enable the algorithms developed in [11] to be applicable in a realistic ice management scenario for UAVs.

A similar but less computationally demanding approach to the problem of mapping the ice concentration in a predefined region is presented in [12]. This algorithm uses an occupancy grid map method for mapping a predefined search region for ice features based on images captured from a UAV and image processing. The occupancy grid map is a collection of grid cells where each cell is given a probability of being occupied by ice or an object of interest. The occupancy probability for each grid cell is then updated by segmenting images into ice and non-ice regions. Generating an occupancy grid map is useful in many ice management scenarios, for instance when a boat is to safely traverse a region with icebergs and ice floes. However, this would require the UAV to not only generate an occupancy grid map, but the UAV should also be able to keep track of the movement of the ice features found which are in the vicinity of the boat's planned route.

In this paper, the focus is on the development, implementation and integration of a UAV iceberg detection and tracking framework for autonomous sea ice management applications. The framework is focused on being able to perform real-time processing of UAV sensor data in order to identify both entire regions of interest (big icebergs) and a set of many smaller objects of interest (ice floes). This is achieved by using the algorithm developed in [12] together with an object detection module developed in [13] which is based on machine vision. In this paper, the output from the object detection module is used as a basis in order to create an occupancy grid map of an area of interest. Further, this paper extends the work presented in [12] by bridging the gap between the occupancy grid map and the UAV autopilot. This is achieved by developing an algorithm which uses the occupancy grid map to generate locations of interest. The resulting locations of interest is interfaced with a UAV path planner based on Model Predictive Control (MPC) developed in [14], in order to close the loop and enable the UAV to autonomously investigate regions of interest further. Enabling the UAV to operate autonomously is a key aspect of this paper, as autonomy of sensor agents in ice management applications reduces the stress on the UAV operator as well as enabling beyond line of sight operations. This paper also includes results from initial testing of the framework in an autonomous sea ice management application, conducted with a fixed-wing UAV equipped with a gimballed thermal camera and an on-board computer for real-time processing.

The remainder of the paper is organized as follows. First, the overall ice tracking framework and its submodules are described in Section II. Further, the occupancy grid map algorithm coupled with a measurement update model from [12] is presented in Section III. Third, a locations of interest generation module which uses the occupancy grid map as input is developed in Section IV. Section V covers the details of two field tests conducted in order to test the overall ice tracking framework using the occupancy grid map module and the locations of interest generator module presented in this paper, while Section VI covers the results of the conducted flights. Finally, the paper is concluded in Section VII.

II. Framework Overview

The overall proposed ice tracking framework is illustrated in Figure 1. The path controller is an MPC as described in [14], which seeks to find an optimal turning rate or waypoint for the UAV, as well as the optimal gimbal attitude given by the pan and tilt angles. The control outputs found by the MPC algorithm are given to the UAV's flight controller (autopilot), which in turn has its own controllers to control the low-level dynamics of the UAV. In this work, the UAV flight control system is assumed to be already implemented to stabilize the UAV's dynamics. The output of the MPC is optimal in the sense that it tries to optimize the path of the UAV and the gimbal attitude with respect to camera time. Camera time is here defined as the time a location of interest is in the field of view of the camera. That is, the MPC is responsible for sending the control inputs (UAV position and gimbal attitude) to the UAV's autopilot which maximize location of interest's camera time. Note that although not ideal, in this paper,

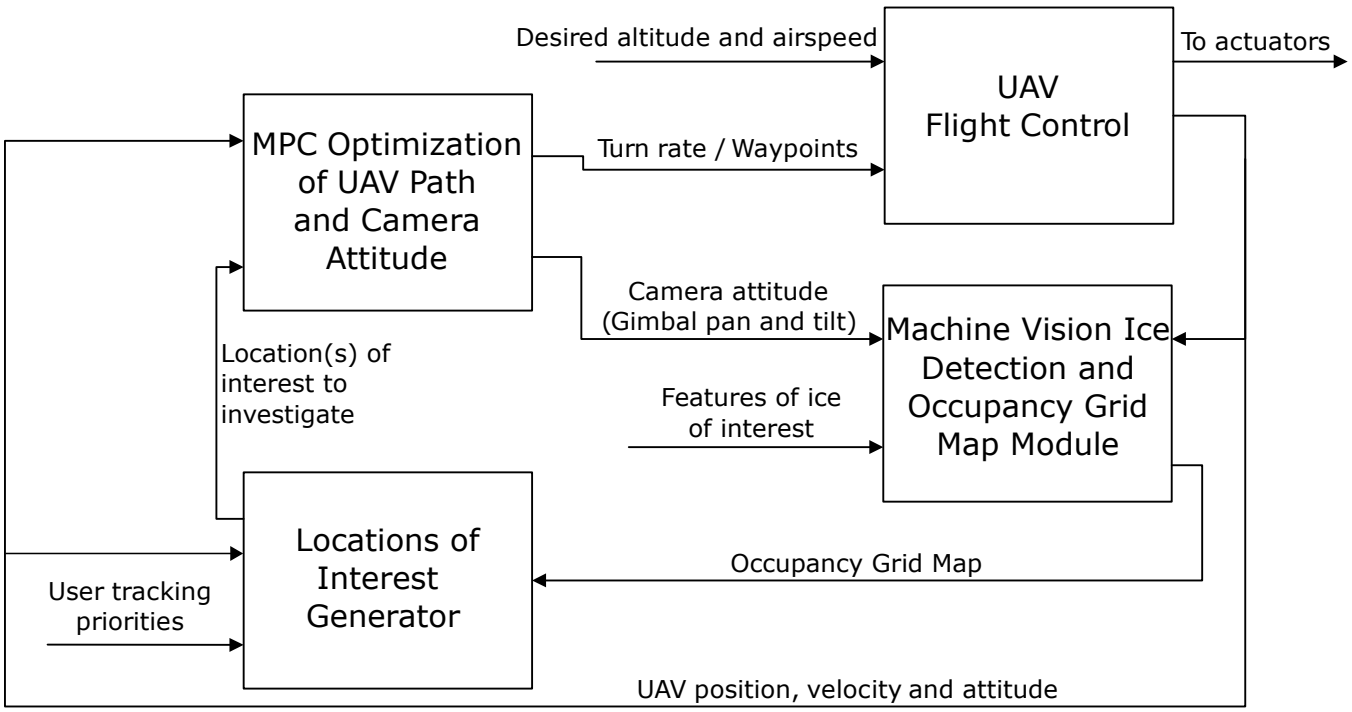

Fig. 1. Overall ice tracking system description. A path planner module (MPC) is combined with a machine vision ice detection and occupancy grid map module and a locations of interest generator in order to detect and track ice features of interest. The UAV's flight controller is assumed already implemented in the present work.

the MPC module was implemented in the ground station, communicating control inputs to the UAV via a wireless link.

The machine vision ice detection and occupancy grid map module, also included in Figure 1, is primarily a machine vision (MV) module running on-board the UAV supplying the locations of interest generator with an occupancy grid map. The occupancy grid map can be calculated using different methods. The most autonomous method, which is implemented in this paper, is to have an on-board computer analyse the images coming from the on-board camera, automatically detecting objects of interest (icebergs and ice floes) over a series of images, and generating the occupancy grid map by combining the visual data with the on-board telemetry data as described in the following section. An example of a machine vision algorithm which can be used in order to detect icebergs and ice floes from the UAV's camera images can be found in [13]. Note that this approach to generating the occupancy grid map is applied without any prior knowledge of the location and number of icebergs and ice floes, but the user will typically input to the MV module some features of the ice features that are of interest (e.g size, anticipated temperature and shape). Another scenario is the case where one already knows the location of the ice features that one want to track, using for instance synthetic aperture radar (SAR) images gathered from a satellite pre-flight. In this scenario it might be useful to do some verification and/or surveillance of the ice features of interest. In this case, the machine vision module is initialized with an occupancy grid map based on the SAR data, and the locations of interest generator and path planner (MPC module) will then make sure that locations of interest will be observed by the UAV. This approach can be combined with the previous mentioned autonomous approach, making the machine vision module continuously update the occupancy grid map in-flight based on the UAV's sensor data. It is also possible with semi-autonomous operations where object detection or recognition is, if needed, assisted by a UAV data analyst. This could in many cases increase the accuracy of the MV module and yield an overall better ice feature detection and tracking performance.

The locations of interest generator module is the module that is responsible for generating and prioritizing which location(s) that should be tracked at any given time. The inner workings of this module may vary depending on the nature of the ice features that are being tracked, but the resulting behaviour is based on the in-flight generated occupancy grid map. Often discrete optimization such as graph search (Travelling Salesman Problem [15]) and MILP [16] can be used as an alternative to find an optimal prioritization of the visiting sequence of locations of interest by some predefined criteria (e.g shortest total flight distance). However, a simple approach where the UAV is made to investigate the location of interest closest to the current position of the UAV can also be effective in many applications. The main concept of this system is that after the MPC algorithm has given the location received by the locations of interest generator a set amount of camera time, the locations of interest generator module will decide the next location(s) that the UAV should prioritize for tracking. An example of a locations of interest generator is developed in Section IV.

III. Occupancy Grid Map

Occupancy grid maps are world fixed grid maps that tries to estimate the probability of whether the grid locations in the map are occupied or free. This is a powerful approach to mapping objects or regions of interest in a predefined search area, especially in scenarios such as ice management where there often are too many objects to track the position and movement of each object of interest individually. A brief overview of an occupancy grid map algorithm developed in [17] and expanded to include the possibility of a dynamic map in [12] is given in this section.

When using an occupancy grid map, the purpose is to estimate the posterior probability density over all possible maps given past measurements and states, i.e

$$p(m_t|z_{1:t}, x_{1:t}) \tag{1}$$

where m_t is the map representation, $z_{1:t}$ are all measurements/observations of the state of the occupancy grid map (described in more detail in Section III-A) and $x_{1:t}$ are the UAV's position and attitude for all time steps from 1 to t.

Now, representing the map as a 2D grid map

$$m_t = \{\mathbf{m}_{i,t}\} \tag{2}$$

where $\mathbf{m}_{i,t}$ are grid elements with an associated (binary) occupancy value, we are interested in knowing the probability of $p(\mathbf{m}_{i,t} = \text{occupied})$ and $p(\mathbf{m}_{i,t} = \text{not occupied})$ for each grid cell $\mathbf{m}_{i,t}$. The term 'occupied' will in our case mean that at least one iceberg or ice floe occupies grid cell i. An assumption made in [17] in order to calculate Equation (1) is that the grid cell densities are independent to each other. That is,

$$p(m_t|z_{1:t}, x_{1:t}) = \prod_i p(\mathbf{m}_{i,t}|z_{1:t}, x_{1:t}) \tag{3}$$

In some cases this could be an inaccurate assumption as the probability of locating icebergs or ice floes in one grid could indicate a higher probability of finding icebergs or ice floes in neighbouring grid cells. However, incorporating this conditional probability would greatly increase the complexity of the problem, while not necessarily significantly increase the occupancy grid map's accuracy [17].

Furthermore, assuming that $\mathbf{m}_{i,t}$ and x_t are Markov, i.e

$$p(\mathbf{m}_{i,t}|z_{1:t-1}, x_{1:t-1}) = p(\mathbf{m}_{i,t}|\mathbf{m}_{i,t-1}) \tag{4}$$

combined with the fact that estimating the state of an individual grid cell is a binary estimation problem (occupied or not occupied), it is possible to estimate the entire grid state by using a binary Bayes filter to estimate the state of each grid cell individually [17].

The reader is referred to [12] for a detailed description of the development of the binary Bayes filter, but the main concept is to exploit that the probability of grid cell $\mathbf{m}_{i,t}$ not being occupied, $p(\neg\mathbf{m}_{i,t})$, can be given as

$$p(\neg\mathbf{m}_{i,t}) = 1 - p(\mathbf{m}_{i,t}) \tag{5}$$

and defining the following odds ratio

$$\frac{p(\mathbf{m}_{i,t}|z_{1:t}, x_{1:t})}{p(\neg\mathbf{m}_{i,t}|z_{1:t}, x_{1:t})} \tag{6}$$

[12] then develops an additive update equation for each grid cell $\mathbf{m}_{i,t}$ by taking the logarithm of Equation (6). This yields the following additive update equation

$$\begin{aligned} l_{i,t} = &\log_{10}\frac{p(\mathbf{m}_{i,t}|\mathbf{m}_{i,t-1})}{1-p(\mathbf{m}_{i,t}|\mathbf{m}_{i,t-1})} + \log_{10}\frac{p(\mathbf{m}_{i,t}|z_t, x_t)}{1-p(\mathbf{m}_{i,t}|z_t, x_t)} \\ &- \log_{10}\frac{p(\mathbf{m}_{i,t})}{1-p(\mathbf{m}_{i,t})} \end{aligned} \tag{7}$$

where the first term will be $l_{i,t-1}$ if the map is static. Using for instance the system developed in [18], an estimate of ice drift can be found by analysing SAR images from a satellite. This estimate could be supplied to the UAV ice management system and incorporated in the derivation of this update equation, introducing map dynamics (ice drift) into Equation (7). $p(\mathbf{m}_{i,t}|z_t, x_t)$ is referred to as the inverse measurement model and is covered in the next subsection, while the last term can be interpreted as the value returned by the inverse measurement model when no information is supplied by a measurement.

Note that in order to get the actual grid map probabilities we have to calculate

$$p(\mathbf{m}_{i,t}|z_{1:t}, x_{1:t}) = \frac{exp(l_{i,t})}{1 + exp(l_{i,t})} \tag{8}$$

A. Measurement Update

Looking at the additive logarithmic update equation developed in the previous section, i.e (7), it is readily seen that the inverse measurement model has to be decided in order to calculate $l_{i,t}$. That is, the term

$$p(\mathbf{m}_{i,t}|z_t, x_t) \tag{9}$$

has to be defined. This term is the probability of the map grid cell i being occupied given an image z_t and a UAV position and attitude x_t. There are many possible choices for this probability function, with the only requirement being that $p(\mathbf{m}_{i,t}|z_t, x_t) \mapsto [0, 1]$.

Ideally this probability function should be found in a systematic way using empirical data, such as images of the objects or regions of interest captured by a UAV. However, since such a data set was not available, a simple probability function assuming that an image has been segmented into foreground (objects or regions of interest) and background (non-interesting regions) was used for demonstration purposes. More specifically, assuming that each pixel $s_{k,t}$ of the image z_t is segmented into foreground and background, [12] propose using

$$n_{k,t} = \begin{cases} \log_{10}\frac{P_1}{1-P_1} & s_{k,t} = \text{Object of Interest} \\ \log_{10}\frac{P_2}{1-P_2} & s_{k,t} = \text{Non-interesting Region} \end{cases} \tag{10}$$

where $n_{k,t}$ is a partial measurement model. Note that P_1 should be a parameter reflecting the object detection algorithm's robustness against false positives. That is, if the object

detection algorithm has a low percentage of false positives, P_1 could be set close to 1 because in such cases a pixel segmented into the "object of interest" category means that this pixel is very likely to originate from an actual object of interest. If the object detection algorithm has a high percentage of false positives, P_1 should reflect this by being set to a lower, non-zero value. Similarly, P_2 reflects the object detection module's percentage of false negatives. I.e, if the object detection module rarely fails to detect objects of interest if they are actually present, P_2 could be set to 0 since a pixel segmented into the non-interesting region is in such cases very likely to actually be a non-interesting region. On the other hand, if the object detection module regularly fails to detect objects of interest when they are present, P_2 should be set higher than 0 to reflect that a pixel segmented into the non-interesting region not necessarily originated from a non-interesting region but in fact an object of interest.

To have a complete measurement model, the UAV's position and attitude x_k has to be used in order to map a pixel coordinate in the image frame to a grid cell in the map frame. The occupancy grid map can be represented by a $N \times M$ matrix, and each cell in the grid map can be represented with homogeneous coordinates $\mathbf{p}^m = [x^m, y^m, 1]$. $x^m \in \{0, ..., N-1\}$ denotes the index of the cell in the horizontal direction, and $y^m \in \{0, ..., M-1\}$ denotes the index of the cell in the vertical direction. Defining the grid map to be in the positive quadrant of the $x-y$ plane in the map frame, the center of a grid cell in the map frame can be found by

$$\mathbf{q}^m = \begin{bmatrix} D_x & 0 & 0 \\ 0 & D_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^m \\ y^m \\ 1 \end{bmatrix} = \mathbf{M}\mathbf{p}^m \tag{11}$$

where $\mathbf{q}^m$ is the center of grid cell $\mathbf{p}^m$ given in map frame coordinates, and D_x and D_y are metric scaling factors describing the size of a grid cell in the x and y direction. Now, the center of a given grid cell $\mathbf{p}^m$ can found in a local North-East-Down (NED) coordinate frame with the following equation

$$\begin{aligned} \mathbf{q}^{ned} &= \mathbf{R}_m^{ned}\mathbf{M}\mathbf{p}^m - \mathbf{R}_m^{ned}\mathbf{C}_{ned,m} \\ &= (\mathbf{R}_m^{ned}\mathbf{M} + \begin{bmatrix} \mathbf{0} & \mathbf{0} & -\mathbf{R}_m^{ned}\mathbf{C}_{ned,m} \end{bmatrix})\mathbf{p}^m := \mathbf{P}_m^{ned}\mathbf{p}^m \end{aligned} \tag{12}$$

where $\mathbf{R}_m^{ned}$ is the rotation matrix from the map frame to the NED frame and $\mathbf{C}_{ned,m}$ is the position of the origin of the NED frame given in the map frame. The reader is referred to [12] for a more detailed development of this equation. Having found the location of a grid cell's center in the NED frame, the location of a grid cell's center in the image frame, $\mathbf{p}^{img}$, can be calculated. That is

$$\begin{aligned} w_s\mathbf{p}^{img} &= \mathbf{A}(\mathbf{R}_{ned}^{cam}\mathbf{P}_m^{ned}\mathbf{p}^m - \mathbf{R}_{ned}^{cam}\mathbf{C}_{cam,ned}) \\ &= \mathbf{A}(\mathbf{R}_{ned}^{cam}\mathbf{P}_m^{ned} + \begin{bmatrix} \mathbf{0} & \mathbf{0} & -\mathbf{R}_{ned}^{cam}\mathbf{C}_{cam,ned} \end{bmatrix})\mathbf{p}^m \\ &:= \mathbf{H}_m^{img}\mathbf{p}^m \end{aligned} \tag{13}$$

where $\mathbf{R}_{ned}^{cam}$ is the rotation matrix from the NED frame to the camera frame and $\mathbf{C}_{cam,ned}$ is the position of the camera given in the NED frame. w_s is a scaling factor describing the scaling ambiguity in camera images, and $\mathbf{A}$ is the intrinsic parameter matrix for the on-board camera. This matrix can be estimated through a process called camera calibration, and an example method of doing this with thermal cameras is described in [19]. $\mathbf{H}_m^{img}$ will then be the perspective transformation between the map frame and the image frame. This means that a segmented pixel $\tilde{\mathbf{p}}^{img}$ in image z_t can be mapped to a grid cell coordinate in the map frame, $\tilde{\mathbf{p}}^m$, by the following equation

$$\frac{1}{w}\tilde{\mathbf{p}}^m = (\mathbf{H}_m^{img})^{-1}\tilde{\mathbf{p}}^{img} = \mathbf{H}_{img}^m\tilde{\mathbf{p}}^{img} \tag{14}$$

Using this equation with the partial measurement model in (10) and the additive update rule in (7), the occupancy grid map can be updated to reflect the observations made by the object detection algorithm for every segmented image.

IV. Generating Locations of Interest

An initial occupancy grid map for a given region can made be by having a UAV search through a pre-defined search region in a systematically manner using the occupancy grid map method with the proposed measurement update described in the previous section. However, the initial occupancy grid map could also be generated from other types of data, such as for instance data from a SAR satellite. An example of an initial occupancy grid map resulting from such methods is shown in Figure 2. The map shows a total of 6 regions of interest where the occupancy grid map has cells with a probability of being occupied larger than 0. In order to make this occupancy grid map useful for the ice tracking framework described in Section II, a method to generate locations for the MPC to track has to be established. There are many possible algorithms for generating these locations, however, in this paper the focus is on an approach which is directly compatible for the overall ice tracking framework.

The general approach to incorporate the occupancy grid map into the system is to use the map to generate locations which the MPC should investigate further. For illustration purposes a synthetically generated occupancy grid map (shown in Figure 2a) will be used as an example. The first step in the process of generating these locations is to perform a thresholding technique on the occupancy grid map using a certain threshold $T_i \in [0, 1]$. The reason for doing this is that when automatic image processing is used in order to perform the measurement update step of the occupancy grid map, false positives could cause the occupancy grid map to have (small) regions (a small number of cells) which has non-zero probability, while still not being occupied by any objects of interest. Since the false positives usually are not as consistent as the true positive detections made by the machine vision algorithm, the idea is that regions with a non-zero probability caused by false positives will still have a smaller probability than regions with a non-zero probability caused by true positives. Thresholding the occupancy grid map will yield a binary map where each cell is either a 1 or a 0. This binary map can optionally be filtered to remove blobs

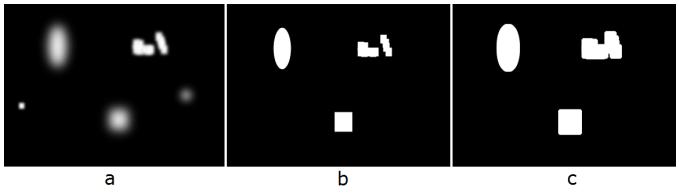

Fig. 2. A synthetically generated example occupancy grid map (a). The occupancy grid map can be manipulated to make generating locations of interest based on the map easier. This is done by by treating the map as an image and performing thresholding (b) before dilating the result of the thresholding (c). This will remove some non-interesting regions of the map while simultaniously emphasizing and expanding actual regions of interest.

which are larger or smaller than a certain size, depending on what kind of regions of interest the system is looking for. An example of a binary map made by thresholding and filtering the blobs based on their size from the original occupancy grid map shown in Figure 2a is illustrated in Figure 2b. Note that one region has been removed because its size was too small (far left), and one region has been removed since non of the grid cells in this region had a value higher than the threshold value T_i.

Furthermore, in some ice management applications, it could be beneficial if regions which are estimated to be occupied by objects of interest which are close to each other are viewed as one connected region. This is especially useful for the scenario of ice management, where it is useful to look at several icebergs and ice floes simultaneously when trying to estimate their general velocity and movement. In order to make regions of interest close to each other appear as one connected region, the binary map generated in the previous step can undergo a process called dilation. This is a process which probes and expands the shapes contained in an input image (in this case a thresholded occupancy grid map) using a predefined geometric form (for instance a circle). The result of dilating the binary map in Figure 2b is shown in Figure 2c. Note how the previously disconnected regions on the right side of the map now is connected. It should also be noted that since that by dilating an image and expanding the shapes contained in it, the dilated thresholded occupancy grid map will set grid cells which are not likely to contain any ice features as occupied cells, care should be taken when the UAV is sent to investigate these regions. That is, the dilated thresholded occupancy grid map can be used in order to decide where the UAV should go and in which order each region should be visited, but ultimately the tracking process (e.g the control of the gimbal) should be based on the originally thresholded occupancy grid map shown in Figure 2b.

Having generated a binary occupancy grid map of an area of interest, the locations for the MPC to track is simply generated by finding the centroid of each of these regions. Feeding the Object Handler and the MPC with these points will work as a replacement for an estimate of the position of rigid objects, as the object detection, recognition and tracking module from the previous chapter was implemented to supply to the remainder of the system. Using the centroid of the regions found to be occupied by objects of interest is effective as long as the regions of interest can be completely captured within one image frame. However, if this is not the case, a more complex algorithm generating several locations per occupied region might be necessary. An example of an algorithm which could be used to generate several loitering points that will completely cover such regions can be found in [20].

Fig. 3. Launch of the X8 UAV at Ny-Aalesund. Picture courtesy of Kjell Sture Johansen.

V. UAV Field Test

In order to test the performance of the object tracking framework described in the previous sections, several field tests were conducted. The system was implemented on the X8 Skywalker [21]. It is a small, light-weight fixed-wing UAV made out of styrofoam with the focus of supplying a cheap expendable aircraft for short-duration (≤ 1 hour) UAV flights. The X8 was equipped with a payload similar to the one described in [19]. This payload consists of the open-source ArduPilot [22] autopilot, a pan-tilt gimbal, a thermal camera and an on-board single board computer capable of performing real-time image processing. The single board computer used in the flight tests conducted in this work is an ODROID U3 [23], which has an ARM-based 1.7 GHz quad core CPU and 2 GB RAM. The thermal camera used is a FLIR Tau2 640 [24]. It has a sampling frequency of 9 frames per second (upsampled to 30 frames per second for analog output) and is sensitive to the long-wave infrared spectral band ($7.5 - 13.5\mu m$) with a sensitivity of $< 50mK$. It has a resolution of 640×480 pixels and a field of view of $32^\circ \times 26^\circ$. The camera is placed inside a retractable R-BTC88 [25] gimbal. The gimbal can be controlled either automatically by the on-board single board computer, or from a manual controller located in the ground station.

An initial field test of the occupancy grid map and the locations of interest generator integrated with the overall ice tracking system as described in Section II was tested with the X8 UAV payload at Ny-Aalesund, Svalbard in the spring of 2016. Figure 3 shows how the X8 UAV platform is launched at the airport of Ny-Aalesund using a catapult.

Fig. 4. The flight plan for the initial search phase for iceberg detection and tracking.

Two field tests were conducted, both at approximately 100 meter altitude. The purpose of the first field test was to gather example thermal video footage of icebergs and ice floes floating in the fjord just outside Ny-Aalesund. This was necessary in order to see if the segmentation algorithm developed in [13] could be used to segment an image into ice and non-ice regions required for the measurement update step described in Section III-A. More specifically, this means that both the thresholding value for the algorithm in [13] and the size of the icebergs and ice floes that were to be detected had to be tuned according to the thermal video data gathered during the first flight test. Note that this is a process which could, and ideally should, be automated, or alternatively implemented so that the parameters could be tuned on-board and in real-time. This would be cruicial in order to increase the autonomy of the system, as weather conditions and environment temperatures will continously change. However, with limited thermal data containing sea ice available, the possibility of performing the parameter tuning autonomously was not investigated further at this time.

After having tuned the ice detection machine vision algorithm according to the thermal video data gathered, the second flight test could be conducted. The purpose of this flight test was to illustrate how the overall ice tracking system can be used as a UAV ice management platform. Unfortunately, at the day of the testing there was only one iceberg within the reach of the X8 UAV platform, hence a full scale test of the system for ice management has to be tested at a later time.

In order for the occupancy grid map to be established and contain the iceberg located within the reach of the X8, the flight plan illustrated in Figure 4 was utilized. The iceberg was estimated (by human observation) to be somewhere inside the red square marked on the map. As seen from Figure 4, using the shown flight plan to find an occupancy grid map will not yield a complete occupancy grid map over the fjord outside Ny-Aalesund. However, since there only was one iceberg within the reach of the UAV and the flight time of the X8 is somewhat limited, it was decided that finding an occupancy grid map for the search pattern illustrated was sufficient.

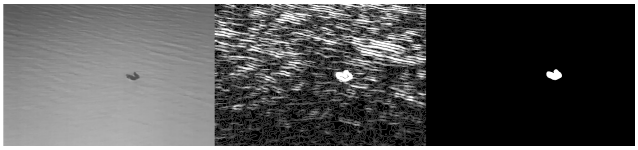

Fig. 5. The iceberg detection process using machine vision. On the left is the original thermal image of an ice floe. In the middle is the processed image, where a varation of an edge detector is used to identify objects of interest. The image to the right show the result of thresholding the resulting image from the edge detector in order to extract the region of the image where the ice floe is located.

The following is the results of the conducted flight tests, which is a fully autonomous test of the ice tracking framework in the sense that the machine vision algorithm directly communicates its on-board, real-time generated occupancy grid map to the locations of interest generator, which in turn communicates a location of interest to the path planner (MPC module). Note however that once the MPC was activated with a generated location of interest for the detected iceberg, the occupancy grid map was not further updated based on new detections of the tracked iceberg.

VI. Results

Using the object detection algorithm developed in [13] on the thermal video data gathered during the first flight test proved to be very successful at segmenting the thermal images into iceberg/ice floe regions and non-interesting regions. An example of such a segmentation is shown in Figure 5. This figure is an example on how, when choosing appropriate values for a tunable threshold parameter, T_g, and the size of blobs that should be filtered, an iceberg is segmented into foreground while the rest of the ocean surface is segmented into background. $T_g = 230$ was found to be a threshold value that yielded a robust iceberg and ice floe detector. Furthermore, filtering out all blobs that consisted of less than 100 pixels was found to remove small blobs which could occur from time to time because of waves and ripples in the ocean surface.

Having tuned the iceberg and ice floe segmentation algorithm, the second flight test could then be conducted and the occupancy grid map algorithm and the locations of interest generator could be tested with the overall system. The occupancy grid map was initialized as a 600×1200 m^2 map where each cell was 10×10 m^2 in size. The initial logarithmic probability, $l_{i,0}$ was set to 0, which means that each grid cell was assumed to have an equal probability of being occupied or not. P_1 was set to 0.8 and P_2 was set to 0.2. Flying 1 round of the flight path shown in Figure 4 while updating the occupancy grid map based on the captured segmented thermal images yielded the development of the occupancy grid map that is shown in Figure 6. Note that for

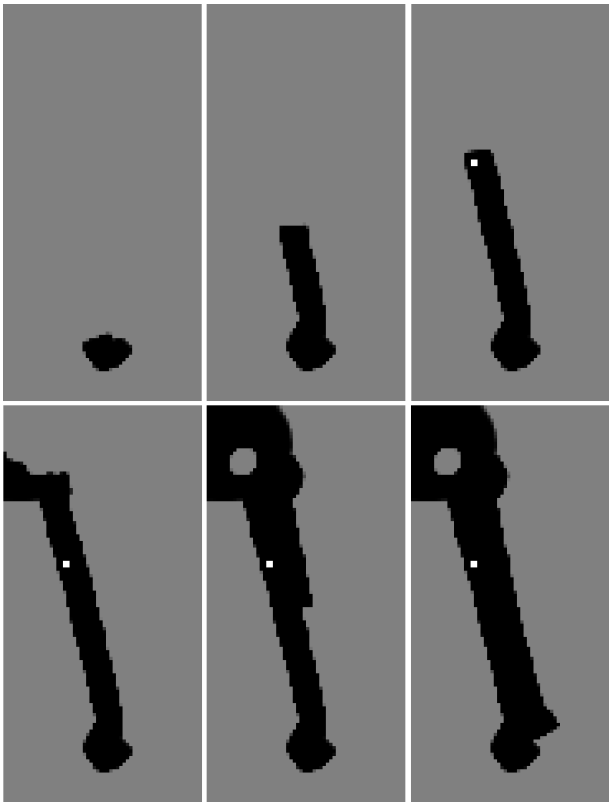

Fig. 6. The state of the occupancy grid map at different points in time during the second flight test conducted at Ny-Aalesund. The top left occupancy grid map is the grid map state at the beginning of the search, while the bottom right occupancy grid map is the state of the grid map towards the final stages of the initial search.

simplification purposes, the gimbal on-board the X8 UAV platform was set to point directly downwards in the UAV body frame. This was done to reduce the uncertainty in the projection of pixels into the map frame as described in Section III-A.

Using the occupancy grid map shown in Figure 6, a location of interest was generated by thresholding this map with $T_i = 0.8$ after the UAV had completed 1 round of the flight plan. This yielded the thresholded occupancy grid map shown in Figure 7, with the centroid of the blob located at -225 m North and 60 m East in the North-East plane. This point was then autonomously given to the MPC path planner, and the MPC was activated with a reference loiter radius of 125 m. The resulting iceberg tracking behaviour is illustrated in Figure 8, while the distance from the X8 UAV to the estimated location of the iceberg is shown in Figure 9.

Looking at Figure 8 it is readily seen that the MPC is successful at keeping the UAV in the wanted tracking range of $\sim$ 125 m to the located iceberg for most of the time. However, on the right hand side of the loiter a more unstable tracking pattern is observed. The unstable tracking pattern is also visible in Figure 9, where it is observed that the UAV's distance to the estimated iceberg location occasionally dips $10 - 30$ m below the 125 m reference range. Although

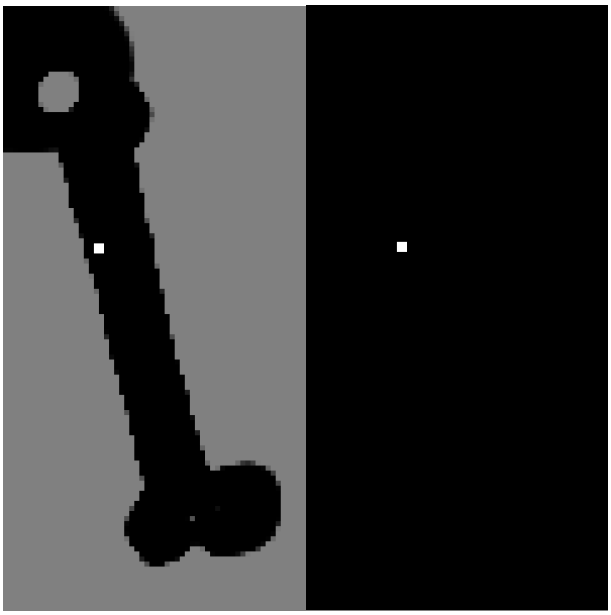

Fig. 7. The complete (left) and thresholded (right) occupancy grid map for the second flight test.

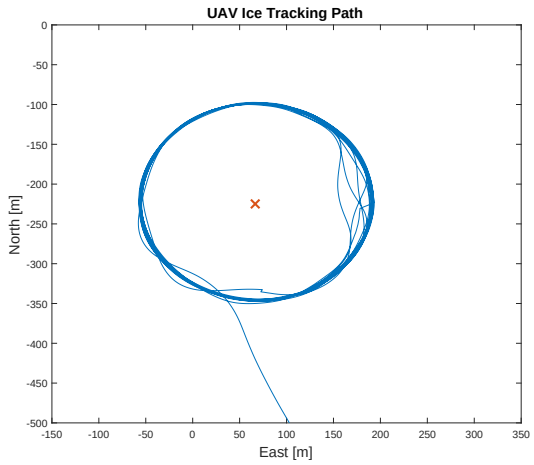

Fig. 8. UAV path while using an MPC to track an iceberg.

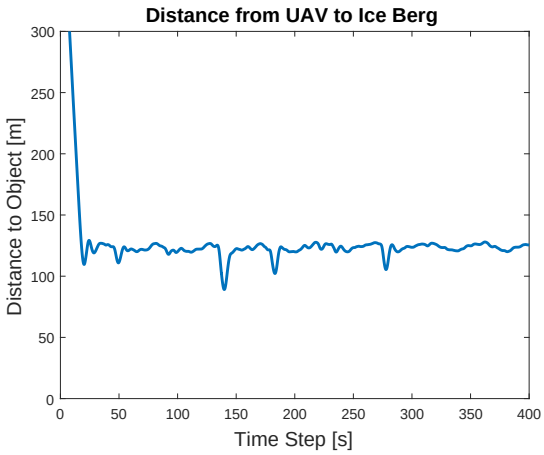

Fig. 9. Distance from the UAV to the estimated position of the iceberg.

the data of the UAV's signal strength to the ground station is not available, a temporary loss of signal was sometimes observed when the UAV was traversing the right hand side of the loiter. Since the MPC was implemented in the ground station, a loss of communication between the UAV and the ground station will cause the ground station to not receive updated telemetry data from the UAV. Hence, the ground station MPC believes that the UAV is standing still during the time that the communication is lost since it will assume the most recent received telemetry message is the current state of the UAV. This causes the MPC to generate a waypoint which is compliant with the belief that the UAV is at a different position than what it really is. When the communication link between the UAV and the ground station is re-established, this wrongly calculated waypoint is the first waypoint the UAV receives. However, the ground station will quickly receive a telemetry update from the UAV, which enables the ground station MPC to calculate a more reasonable waypoint, correcting the UAV course so that it can continue to follow the ideal loiter pattern. This problem could be mitigated by implementing the MPC completely on-board and getting rid of the MPC's ground station component.

VII. Conclusions

In this paper, a UAV ice tracking framework for autonomous sea ice management was developed. The framework is focused on enabling a UAV to perform real-time processing of on-board sensor data and autonomous decision making in ice management applications. This is achieved by combining a machine vision algorithm for object detection with an occupancy grid map algorithm. Furthermore, in order to enable autonomy, these algorithms are coupled with a locations of interest generator and an MPC UAV path planner. By doing this, the UAV is able to automatically identify regions of interest based on the detection of ice features of interest during an initial search period, before autonomously investigating these regions of interest further using the MPC path planner.

The object detection algorithm developed in [13] was proven to be successful in segmenting the UAV's thermal images into ice and non-ice regions by tuning the algorithms parameters. Furthermore, using on-board navigation data to get the UAV's and camera's attitude and altitude, the on-board computer was able to automatically generate and update an occupancy grid map over an area of interest, where each cell in the grid map is given a probability of being occupied by an iceberg or ice floe. The generated occupancy grid map can in turn be used in order to generate locations of interest, which are locations that the UAV should investigate and keep track of.

The presented ice tracking framework was implemented on-board a UAV payload, and was tested in a UAV flight test conducted in the fjord outside Ny-Aalesund at Svalbard. It was found that the system was able to successfully create an occupancy grid map based on automatically segmenting thermal images on-board and in real-time into ice and non-ice regions. Furthermore, using this occupancy grid map, a single location of interest was detected and its centroid was calculated and passed to the MPC path planner developed in [14]. The MPC was successful at keeping the UAV in the tracking range of the estimated iceberg location for most of the time. However, some challenges regarding communication delay and having the MPC implemented off-board in the ground station was observed. Although not a major issue, this definitely emphasize the need to implement the MPC in a more robust way, e.g completely on-board, for ice management missions to be conducted in the future.

Although nominal testing indicates that the proposed framework is suitable for ice management scenarios, a more complete test where there are more icebergs and ice floes should be conducted in order to better assess the performance of the ice tracking framework as a UAV ice management platform. The accuracy of the resulting occupancy grid map should also be assessed in further testing.

Acknowledgment

The authors would like to thank Lars Semb and Pål Kvaløy for their technical support and flawless execution of the practical aspects of the field experiments. This work has been carried out at the NTNU Centre for Autonomous Marine Operations and Systems (NTNU AMOS). The Norwegian Research Council is acknowledged as the main sponsor of NTNU AMOS. This work was supported by the Research Council of Norway through the Centres of Excellence funding scheme, Project number 223254.

References

[1] J. McClintock, R. McKenna, and C. Woodworth-Lynas, "Grand banks iceberg management: Perd/chc report 20-84," in *Report prepared by AMEC Earth & Environmental, St. John's, NL, RF McKenna & Associates*. Wakefield, QC, and PETRA International Ltd Cupids, NL, 2007.

[2] O. M. Johannessen, V. Alexandrov, I. Y. Frolov, S. Sandven, L. H. Pettersson, L. P. Bobylev, K. Kloster, V. G. Smirnov, Y. U. Mironov, and N. G. Babich, *Remote sensing of sea ice in the Northern Sea Route: studies and applications*, 1st ed. Springer Science & Business Media, Berlin Heidelberg, 2007.

[3] K. Eik, "Review of experiences within ice and iceberg management," *Journal of Navigation*, vol. 61, no. 04, pp. 557–572, 2008.

[4] J. Haugen, L. Imsland, S. Løset, R. Skjetne *et al.*, "Ice observer system for ice management operations," in *The Twenty-first International Offshore and Polar Engineering Conference*. International Society of Offshore and Polar Engineers, 2011.

[5] I. Lesinskis and A. Pavlovics, *Marine Navigation and Safety of Sea Transportation: Navigational Problems*, 1st ed. CRC Press, London, UK, 2013, ch. Monitoring of Ice Conditions in the Gulf of Riga Using Micro Class Unmanned Aerial Systems, pp. 167–172.

[6] T. Eltoft, A. P. Doulgeris, C. Brekke, S. Solbø, S. Gerland, and A. Hanssen, "Imaging sea ice structure by remote sensing sensors." Proceedings of the 23rd International Conference on Port and Ocean Engineering under Arctic Conditions June 14-18, 2015 Trondheim, Norway, 2015.

[7] M. Westoby, J. Brasington, N. Glasser, M. Hambrey, and J. Reynolds, "Structure-from-motion photogrammetry: A low-cost, effective tool for geoscience applications," *Geomorphology*, vol. 179, pp. 300–314, 2012.

[8] J. C. Ryan, A. L. Hubbard, J. E. Box, J. Todd, P. Christoffersen, J. R. Carr, T. O. Holt, and N. Snooke, "UAV photogrammetry and structure from motion to assess calving dynamics at store glacier, a large outlet draining the greenland ice sheet," *The Cryosphere*, vol. 9, no. 1, pp. 1–11, 2015. [Online]. Available: http://www.the-cryosphere.net/9/1/2015/

[9] A. Stalmakou, "UAV/UAS path planning for ice management information gathering," June 2011, Master Thesis, NTNU, Department of Engineering Cybernetics.
[10] A. Albert and L. Imsland, "Mobile sensor path planning for iceberg monitoring using a milp framework," in *Informatics in Control, Automation and Robotics (ICINCO), IEEE 12th International Conference on*, vol. 1, 2015, pp. 131–138.
[11] J. Haugen, "Autonomous aerial ice observation," 2014, Doctoral Thesis, NTNU, Department of Engineering Cybernetics.
[12] A. L. Flåten, "Experimental monitoring of sea ice using unmanned aerial systems," June 2015, Master Thesis, NTNU, Department of Engineering Cybernetics.
[13] F. S. Leira, T. A. Johansen, and T. I. Fossen, "Automatic detection, classification and tracking of objects in the ocean surface from UAVs using a thermal camera," in *Proceedings of the IEEE Aerospace Conference*, 2015, pp. 1–10.
[14] E. Skjong, S. A. Nundal, F. S. Leira, and T. A. Johansen, "Autonomous search and tracking of objects using model predictive control of unmanned aerial vehicle and gimbal: Hardware-in-the-loop simulation of payload and avionics," in *Proceedings of the IEEE International Conference on Unmanned Aircraft Systems (ICUAS)*, 2015, pp. 904–913.
[15] D. L. Applegate, R. E. Bixby, V. Chvatal, and W. J. Cook, *The traveling salesman problem: a computational study*. Princeton university press, 2011.
[16] A. Albert and L. Imsland, "Mobile sensor path planning for iceberg monitoring using a MILP framework," in *Informatics in Control, Automation and Robotics (ICINCO), IEEE 12th International Conference on*, vol. 1, 2015, pp. 131–138.
[17] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. MIT press, 2005.
[18] S. Muckenhuber, A. A. Korosov, and S. Sandven, "Open-source feature-tracking algorithm for sea ice drift retrieval from sentinel-1 sar imagery," *The Cryosphere*, vol. 10, no. 2, pp. 913–925, 2016. [Online]. Available: http://www.the-cryosphere.net/10/913/2016/
[19] F. S. Leira, K. Trnka, T. I. Fossen, and T. A. Johansen, "A ligth-weight thermal camera payload with georeferencing capabilities for small fixed-wing UAVs," in *Proceedings of the IEEE International Conference on Unmanned Aircraft Systems (ICUAS)*, 2015, pp. 485–494.
[20] E. J. Forsmo, T. I. Fossen, T. A. Johansen *et al.*, "Optimal search mission with unmanned aerial vehicles using mixed integer linear programming," in *Unmanned Aircraft Systems (ICUAS), IEEE International Conference on*, 2013, pp. 253–259.
[21] (2016) X8 skywalker. http://www.airelectronics.es/products/solutions/x8/.
[22] (2016) APM autopilot suite. http://www.ardupilot.com/.
[23] (2015) ODROID U3. http://www.hardkernel.com/main/products/prdt_info.php?g_code=g138745696275.
[24] (2011) FLIR Tau2 640. http://www.flir.com/cores/display/?id=54717.
[25] (2011) Retractable BTC-88. http://microuav.com/btc-88.html.