AIAA Guidance, Navigation, and Control Conference
13 - 16 August 2012, Minneapolis, Minnesota

AIAA 2012-4703

Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703

Performance Characteristics of a Complete Vision Only Sense and Avoid System [*]

Bálint Vanek[†‡] Tamás Péni[*] Ákos Zarándy[§]

József Bokor[¶]

Computer and Automation Research Institute Hungarian Academy of Sciences, Budapest, Hungary.

Tamás Zsedrovits[‖] Tamás Roska [**]

Pázmány Péter Catholic University The Faculty of Information Technology.

The present paper investigates the real world feasibility of a purely vision based sense and avoid system, required for small unmanned aerial vehicles (UAV) to routinely access the national airspace. The two distinct functions, sensing and avoidance are integrated into a common framework. No information is exchanged between aircraft, only passive 2-D vision information is available to estimate the encountering traffic. Based on the predicted intruder motion the time of the encounter and the minimum distance are predicted. In case an intruder violates the minimum separation the onboard autopilot initiates an avoidance maneuver. The viability of the system is demonstrated on several estimation approaches, using Extended Kalman filter (EKF) and Unscented Kalman filter (UKF) implementations. Since it is shown that for certain type of observer movements the estimation process remains unobservable in bearings-only problems, the sensitivity of the estimation performance and the resulting avoidance response with respect to different intruder motion is investigated in a Monte-Carlo simulation. The system is tested on a high fidelity Hardware-in-the-Loop (HIL) simulation platform, where flight control algorithms, scene rendering, image processing and estimation algorithms are implemented individually over a network of computers with special emphasis on parallel implementation of computationally intensive tasks. Representative encounter scenarios are presented to provide performance measures, including detection time and achieved miss distance of distinctive approaches to assess the applicability of the results.

I. Introduction

The emerging role of Unmanned Aerial Systems (UAS) for both military and civil operations depends on the ability to gain unrestricted access to national airspace. One of the key issues that must be resolved to open up the skies for UAS is to be able to coexist safely and effectively with current manned operations in the national and international airspace.[1] This includes the ability to perform Sense and Avoid (SAA) functions at an "equivalent level of safety" (ELOS) to manned aircraft while not negatively impacting the existing infrastructure and manned Traffic Alert and Collision Avoidance System (TCAS) that create today's safe airspace.[2,3] The UAS collision avoidance system, or SAA system, needs to operate for hazards within

[*] This work is supported by the Office of Naval Research Global, Grant Number N62909-10-1-7081, Dr. Paul Losiewicz program officer. This work is also supported by the Control Engineering Research Group of HAS at Budapest University of Technology and Economics

[†] senior research fellow, Systems and Control Laboratory

[‡] corresponding author:vanek@sztaki.hu

[§] Associate Professor

[¶] research director, Head of Systems and Control Laboratory

[‖] PhD candidate

[**] Professor

Copyright © 2012 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved.

Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703

a defined volume of airspace surrounding the UAS. The system needs to detect a hazard, determine if a maneuver is required, communicate and execute that maneuver in time to achieve a specified miss distance. The elements of SAA represent segments of system design that must be studied together, with their performance allocated between the elements so as to achieve statistically safe operation in the planned operating environment. A purely camera based SAA system, which is applicable for tactical, mini and micro size vehicles, would provide cost and weight advantages against radar based solutions currently under research.[4,5] Feasibility and technical characteristics of such system are unknown, since solely electro-optical sensor based approach has not yet been demonstrated on small scale vehicles, only on the Global Hawk UAS.[6] Solutions for the individual building blocks of a SAA system exist, but only ad-hoc approaches have been proposed to build a complete system. Hence, legislation lacks the necessary in-depth knowledge about the interaction of multidisciplinary system components and currently no solutions or standards exist for the non-cooperative SAA problem.

Research aiming towards better understanding the limits and applicability of vision only SAA systems would be greatly beneficial for the UAS community. It is of similar interest to provide trade-off analysis on collision avoidance as a function of vehicle size, velocity and maneuver types and to clearly understand the interaction of navigation, control and sensing-computing systems to provide guidelines for improved, highly effective SAA systems applicable for smaller sizes UAVs. Based on these general results lightweight, vision only, highly integrated SAA solution can be proposed for a limited class of airspace users, based on fusion of novel sensing, navigation and control methods.

First the standalone estimation algorithms have to be compared to show the feasibility of the problem. Even when no avoidance maneuver is required, the system has to track and propagate the motion estimates of the surrounding traffic. The standard way of estimating unknown parameters in linear systems is to use a Kalman filter.[7] In the special case where both the dynamic and observation models are nonlinear but the noises are additive and Gaussian the Extended Kalman filter (EKF)[8] has been the standard technique usually applied. But, for severe nonlinearities, the EKF can perform poorly or even go unstable. Hence it is important to compare the EKF with other methods, to select the best possible approach used in the SAA problem. The system components required for this comparison serves as a core for SAA system development, and denoted consecutively as “Detection”, “Data Association & Tracking” and “Motion Prediction” on Figure 1. The consecutive tasks depending on the estimation are mainly related to avoidance, and denoted consecutively as “Collision Risk Estimation & Decision”, “Trajectory generation”, and “Flight Control”.

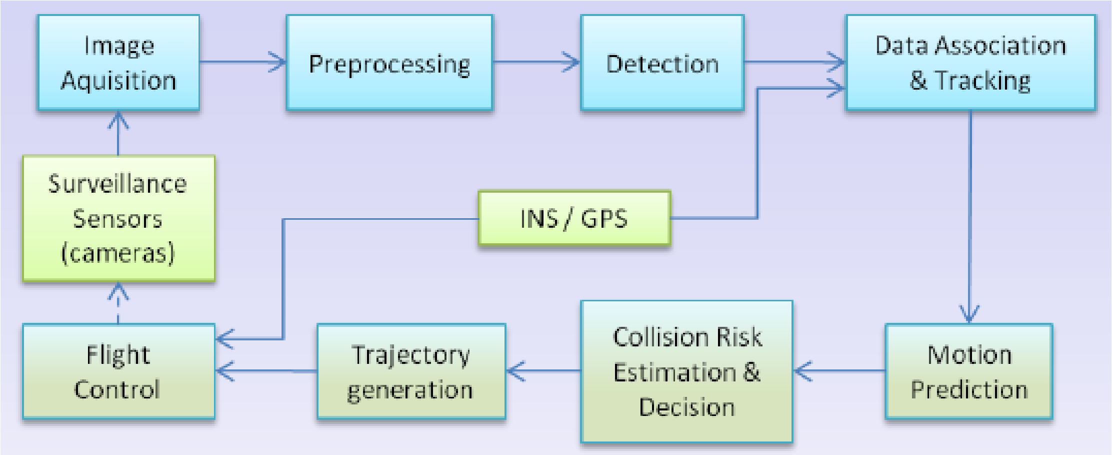


Figure 1. System interconnection of the SZTAKI SAA research platform

The way the motion model is described in the problem setup can also have impact on the solution. The dynamics of the intruder can be described with a simple point mass model. Including higher fidelity dynamics, where wing level measurement is included or flight envelope constraints are enforced, might lead to significantly higher computational load on the solution side, but possibly they can guarantee certain benefits on the estimation and trajectory generation tasks. It is also not straight forward in which frame of reference the problem has to be posed. Different state variables in the estimators, not conventional coordinates in the Earth frame, may lead to computationally better posed problems.

The article is organized in the following way, Section II details the estimation problem using Kalman

Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703

filtering methods, the first task of the system, gathering visual data and analyzing them is described in Section III , the following avoidance tasks are detailed in Section IV. The system performance is analyzed in detail using a Hardware-in-the-loop test environment, from which experimental results are described in V. The article is concluded in Section VI, where future research directions are described towards onboard implementation of the proposed scheme.

II. Kalman filtering based intruder detection

We assume there is only one intruder to be detected, so the case of multiple threats is not considered in the paper. The detection of the intruder is formulated as a state estimation problem, where the dynamics are the relative motion of the intruder to our aircraft. The measured output contains all information that can be extracted from the camera images. Since the camera projects the 3D view onto a 2D plane, which is a nonlinear mapping, the measured outputs are nonlinear functions of the relative position. Depending on the model used[a] either the dynamics or the output equations will be nonlinear. Even if the motion of the aircrafts are modeled by a linear system, the nonlinearity of the output equation makes it necessary to apply Extended (EKF) or Unscented Kalman Filters (UKF).[9, 10] A fundamental “flaw” of the EKF is that the distributions (or densities in the continuous case) of the various random variables are no longer normal after undergoing their respective nonlinear transformations. Alternatively UKF uses a deterministic sampling technique known as the unscented transform to pick a minimal set of sample points (called sigma points) around the mean. These sigma points are then propagated through the non-linear functions, from which the mean and covariance of the estimate are then recovered. The result is a filter which more accurately captures the true mean and covariance. In addition, this technique removes the requirement to explicitly calculate Jacobians, which for complex functions can be a difficult task.

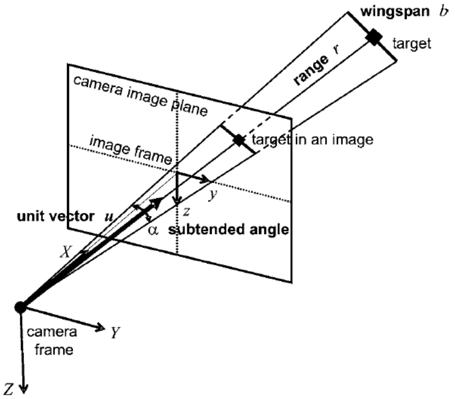


Figure 2. Subtended Angle Relative State Estimation (SARSE) methods

A. Coordinate systems

The following three coordinate systems are used in the paper:

- North-East-Down (NED) - Fixed in one (latitude,longitude) point. We assume flat Earth and the flying distance is short, where axes are: X positive in North, Y positive in the East and Z positive towards the center of Earth (perpendicular to X-Y plane).
- Body - fixed to the c.g. of the aircraft. X_b positive forward, through the nose of the aircraft, Y_b positive to starboard, Z_b axis - positive downwards, perpendicular to X-Y plane.

[a] If the relative position, relative velocity and acceleration are chosen as state variables, the dynamics are linear, but the output equations will be nonlinear. If the the inter-aircraft distance, direction vector and their derivatives constitute the state vector (like in this paper, see section REF) then the dynamics are highly nonlinear, but the measurement is a subset of states.

- Camera - a pan, tilt camera is assumed onboard, which is directed in the intended flight direction, which is fixed throughout the entire flight. (The intended flight direction is the direction in which the own aircraft would normally fly if the intruder did not come.) The camera has offset r' from the c.g., and in the current paper the axes X_c, Y_c, Z_c are chosen to coincide with the axes of NED if the aircraft flies from West to East.

B. Equations of motions

To simplify the filter design, the vehicles (intruder and own aircrafts) are modeled in the NED frame by simple point-mass dynamics. The intruder is assumed to fly along a straight path at constant velocity vector. Let $p_{\text{int}}(t)$, v_{int}, denote the position and velocity of the intruder and let $p_{\text{own}}(t)$, $v_{\text{own}}(t)$, $a_{\text{own}}(t)$ be the motion parameters of the *camera*. (Note that, the camera is not in the c.g., so a coordinate transformation has to be applied to obtain ($p_{\text{own}}(t)$,$v_{\text{own}}(t)$,$a_{\text{own}}(t)$) from the aircraft's motion vector expressed relative to the center of gravity.) If we introduce

$$p = p_{\text{int}} - p_{\text{own}}, \quad r = \|p\|, \quad u = \frac{p}{r}, \quad v = v_{\text{int}} - v_{\text{own}} = \dot{(ur)} = \dot{u}r + u\dot{r}$$

then the equations of motion in the relative coordinates can be expressed as follows[11, 12] :

$$\begin{aligned}
\dot{r} &= \frac{1}{\|p\|} \cdot v^T p = v^T u \\
\ddot{r} &= -a_{\text{own}}^T u + v^T \dot{u} = -a_{\text{own}}^T u + (r\dot{u}^T + \dot{r}u^T)\dot{u} \\
&= -a_{\text{own}}^T u + r\dot{u}^T\dot{u} + \dot{r}u^T\dot{u} = -a_{\text{own}}^T u + r\|\dot{u}\|^2 \\
\frac{d}{dt}\left(\frac{1}{r}\right) &= -\frac{\dot{r}}{r^2} = -\frac{\dot{r}}{r} \cdot \frac{1}{r} \\
\frac{d}{dt}\left(\frac{\dot{r}}{r}\right) &= \frac{\ddot{r}r - \dot{r}^2}{r^2} = \ddot{r} \cdot \frac{1}{r} - \left(\frac{\dot{r}}{r}\right)^2 = -a_{\text{own}}^T u \cdot \frac{1}{r} + \|\dot{u}\|^2 - \left(\frac{\dot{r}}{r}\right) \\
\ddot{u} &= -a_{\text{own}} \cdot \frac{1}{r} - 2\left(\frac{\dot{r}}{r}\right)\dot{u} - p\ddot{r} \cdot \frac{1}{r} = -a_{\text{own}} \cdot \frac{1}{r} - 2\left(\frac{\dot{r}}{r}\right)\dot{u} - \frac{p}{r}\ddot{r} \cdot \frac{1}{r^2} \\
&= -a_{\text{own}} \cdot \frac{1}{r} - 2\left(\frac{\dot{r}}{r}\right)\dot{u} - u \cdot \frac{1}{r}(-a_{\text{own}}^T u + r\|\dot{u}\|^2) \\
&= -a_{\text{own}} \cdot \frac{1}{r} - 2\left(\frac{\dot{r}}{r}\right)\dot{u} + \frac{1}{r}u \cdot a_{\text{own}}^T u - u \cdot \|\dot{u}\|^2
\end{aligned}$$

or in state space form:

$$x = \begin{bmatrix} u \\ \dot{u} \\ 1/r \\ \dot{r}/r \\ b \end{bmatrix}, \quad \dot{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} \dot{u} \\ -a_{\text{own}} \cdot \frac{1}{r} - 2\left(\frac{\dot{r}}{r}\right)\dot{u} + \frac{1}{r}u \cdot a_{\text{own}}^T u - u \cdot \|\dot{u}\|^2 \\ -\frac{\dot{r}}{r} \cdot \frac{1}{r} \\ -a_{\text{own}}^T u \cdot \frac{1}{r} + \|\dot{u}\|^2 - \left(\frac{\dot{r}}{r}\right) \\ 0 \end{bmatrix} + w = f(x, a_{\text{own}}) + w \tag{1a}$$

where w is an additive noise vector representing the effects of modeling inaccuracies and external disturbances.

By locating and tracking the intruder on the image plane the image processing unit can determine the direction vector $u_b(t)$ in the camera frame, which is transformed into the direction vector in the NED frame $u(t)$ using the Direction Cosine Matrix (DCM),[13] notice that the Euler angles used to form the DCM are also subjected to measurement noise. The fourth measurement is the subtended angle $\phi(t) = 2\arctan\frac{b}{2r}$ under which the target is seen. (b is the unknown wingspan of the target, which is also to be estimated by the filters). These parameters constitutes the measurement outputs of the system:

$$z(t) = \begin{bmatrix} u(t) \\ \frac{b}{2\|p(t)\|} + \nu \end{bmatrix} = h(x(t)) + \nu \tag{1b}$$

Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703

where ν is the measurement noise. The estimation task has to reconstruct the state vector x in order that the motion parameters of the intruder could be determined. To implement a suitable EKF or UKF based state estimator the model has to be discretized with given sampling time T_s:

$$x_{k+1} = x_k + T_s f(x_k, a_{vk}) + w_k \tag{2}$$

$$z_k = h(x_k) + \nu_k \tag{3}$$

Since the EKF and UKF formulations are fairly standard, given the system equations in (1) and (3), we refer to [9,10,14] for further details on them.

C. Attributes of the estimation task

It is not difficult to see that the system in (1) is not observable for all input trajectories. Moreover, even if the observability condition holds a less exciting control input may lead to ill-conditioned estimation problem, which fails to converge due to numerical problems or results in unpredictably large estimation errors. Consequently, the selection of a suitable control policy is a key point in the estimation task.[15] Since the control input is the acceleration a_{own}, the input design means finding a suitable trajectory for the own aircraft. In the current paper, an open-loop strategy is employed, with periodic up-down-left-right maneuvering of the own platform as shown on Figure 9, which provides persistent excitation for the estimation problem.

It may often occur in a real application that the measurement data becomes corrupted by some unexpected disturbance or due to some sensor failure. This results in wrong filter inputs, which are inconsistent with the underlying dynamics. The wrongly feeded filter produces false estimation, which can lead to false alarms or even collision. A simple method to detect - at least a part of - these situations is to run a bank of differently initialized filters in parallel. If all filters converge to a same (similar) output we can accept the estimation, otherwise the occurrence of a failure can be presumed and the estimation can not be used. On the other hand, the nonlinearity of the model can also result in the divergence of filters, especially of those, which use linearized dynamics. With careful selection of the model we use, the nonlinearity in itself does not impair the convergence of the estimators, i.e. their outputs tend to the same values irrespective of the (rationally chosen) initial state. Simulations that prove this observation are not shown in the last paper due to space.

The final goal is to determine all parameters from which the motion of the intruder can be reconstructed and precisely estimated. Exploiting the assumption that the intruder flies at a constant velocity, it is clear that two constant parameter vectors (each having three elements) are enough: the velocity v_{int} and the position $p_{\text{int},0}$ where the intruder was detected for the first time. From the estimated $\hat{x}$ the velocity and the current position can be directly obtained:

$$v_{\text{int}} = v_{\text{own}} + \hat{x}_2\hat{x}_3 + u\frac{\hat{x}_4}{\hat{x}_3}, \quad p_{int} = p_{own} + u\hat{x}_3 \tag{4}$$

By definition, $p_{\text{int}} = p_{\text{int},0} + v_{\text{int}}t$ with $t = [0, T_{\text{sense}}]$, where T_{sense} denotes the time available for estimation. To suppress the noise remained on the estimated signals, $p_{\text{int},0}, v_{\text{int}}$ are computed from p_{int} and v_{int} by Least-Square method.

III. Image processing

The main task of image processing is to determine the direction $u(t)$ and the subtended angle $\phi(t)$ of the intruder on the raw camera image frames. In the previous paper[16] it is assumed, that the image sequence is obtained by a gimballed, 2-axis inertial stabilized camera platform which points towards a fixed direction in the NED frame, however this assumption is dropped in the present paper and the image sequences are obtained by a body fixed camera. The image processing algorithm obtains images from the camera, which is fixed to the nose of the aircraft. Its offset relative to the c.g. in the body frame is r', which is defined as shown on Figure 3. The Field of View of the camera is 120 deg, the maximum currently achievable with the FlightGear engine, but in the hardware platform we are using an array of 5 cameras covering the recommended 220 deg FOV.[17]

The raw images, from the image acquisition tool are sent to the processing algorithm, the flowchart schematics is shown on Figure 4. The input images of the algorithm are at least one megapixel. As shown on Fig. 4 the first step is a space variant adaptive threshold[18] to filter out the slow transitions on an image.

Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703

Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703

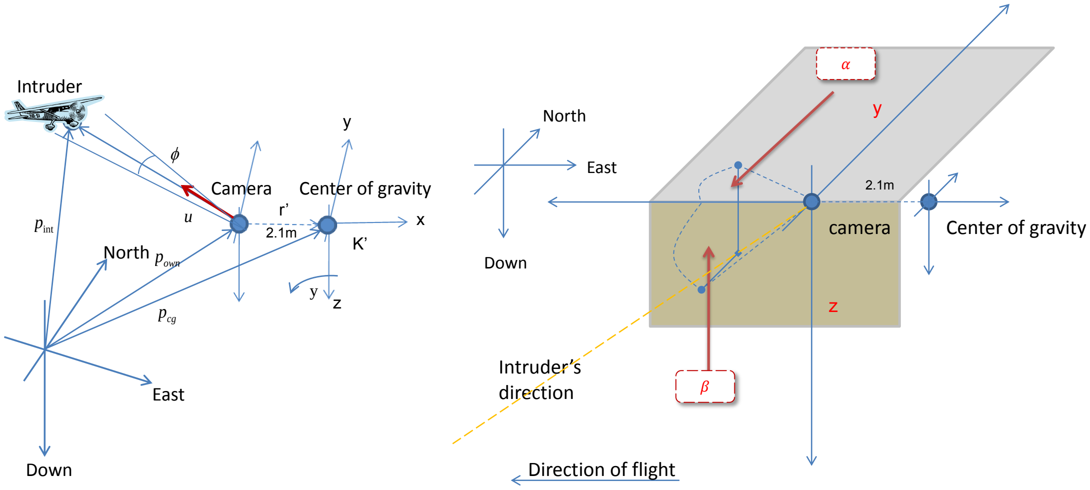
This can be the entire raw image or a smaller sub-image of it, according to that the current picture is the first on which an object is detected or it is part of a sequence where the object is present. To reduce the input image size and speed up the computation, a window containing the intruder airplane according to the previous location and size is cut. In this way the information calculated by the previous step to the location of the other aircraft on the image plane is used. The adaptive threshold results a binary image containing some of the points of the aircraft. On this binary image a centroid calculation[18] is applied. The center of Region of Interest (ROI) is determined by the centroid co-ordinates. The size of the ROI is determined by the previously calculated wingspan plus 20 pixels in each direction. In that way two images are cut: one from the original picture (colored ROI image) and one from the result of the adaptive threshold (binary ROI image).

The aircraft is composed of darker and brighter pixels than the intensity mean value of the original picture. On the colored ROI image two thresholds are ran. The first one is calculated on the inverse picture of the grayscale image created from the red channel of the colored ROI image. With this threshold the pixels brighter than the intensity mean value of the original picture are found. The result is a binary image with the brighter pixels. The other threshold is calculated on the blue channel of the colored ROI image and with it the darker pixels are found. We used the blue channel because the sky is blue and the difference between the background and foreground pictures is the largest in this channel. The result is a binary image with the darker pixels. A logical OR is applied for the two threshold images. The result is a binary picture with the found pixels of the aircraft and with some other pixels. In some cases the parts of the airplane are not connected in this picture. A closing[18] is applied to connect the components. From the binary ROI picture we have an approximation for the aircraft and from the previously calculated picture we have the pixels of the whole airplane with some noise. A recall[19] is applied according to the binary ROI for the double threshold image.

A small picture with the shape of the airplane is obtained in this way. On this picture the centroid in pixels and the pixel count between the left and right wingtip of the aircraft is determined. Based on the outputs provided by this algorithm the direction of the incoming aircraft can be determined in radians in the following way:

$$\alpha = \arctan \frac{x}{x_{\max}} \tan 40^\circ, \quad \beta = \arctan \frac{y}{y_{\max}} \tan 15^\circ \tag{5}$$

where x, y is the position of the centroid in pixels, $x_{\max}, y_{\max}$ is the horizontal and vertical resolution of the image (e.g. 2208×600), 40° and 15° are the half of the FoVs in the two directions. The angles α and β are measured between the direction vector pointing towards the intruder and the y-z (horizontal) and x-z (vertical) plane of the camera, as it is depicted in Fig. 3. In the possession of the coordinate transformation between the camera frame and NED, the direction vector u can be determined from α and β.



(a) The fixed North-East-Down (NED) and the camera frame x-y-z.

(b) Measured variables

Figure 3. Coordinate systems and measured variables

Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703

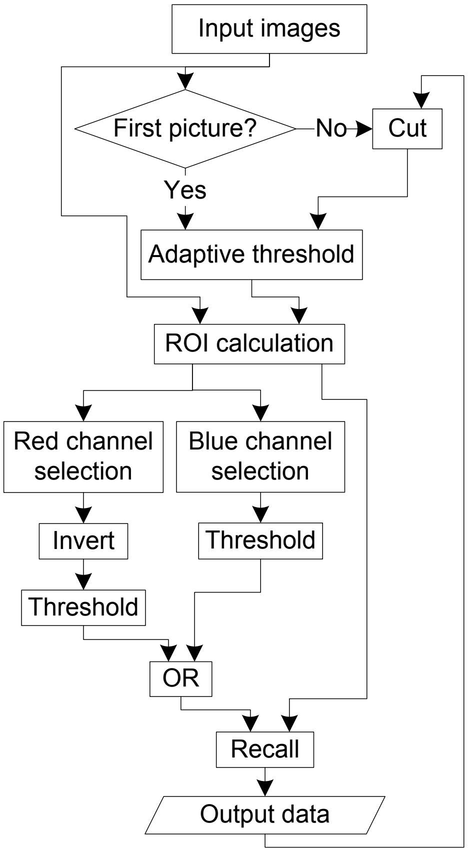


Figure 4. Diagram of the image processing algorithm

Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703

IV. Avoidance

The avoidance maneuvering is executed by supplying new target waypoints to the flight control system, hence it is assumed that the UAV is equipped with a proper autopilot, which can receive such commands.

A. Estimating the time to collision and miss distance

To reliably detect the risk of collision, we need the position and time when the intruder flies closest to our aircraft. Since the own aircraft tracks a rapidly varying trajectory (to maintain the persistency of excitation) it has to be clearly defined what do we mean by this closest point. In this paper we apply the following concept: at each time instant we connect the actual position of the own craft with a remote future waypoint on the intended path and consider a straight line trajectory that our aircraft would fly if it would immediately finish the estimation and would start to return with constant velocity $\bar{v}_{own}$ to the intended path along this straight line, as shown in Figure 5. Since the intruder flies also at constant speed the problem of determining the 'time to collision' t^*, the time needed to reach the encounter point, and 'miss distance' d^*, the distance between aircrafts at encounter point, boils down to the minimization of a quadratic function:

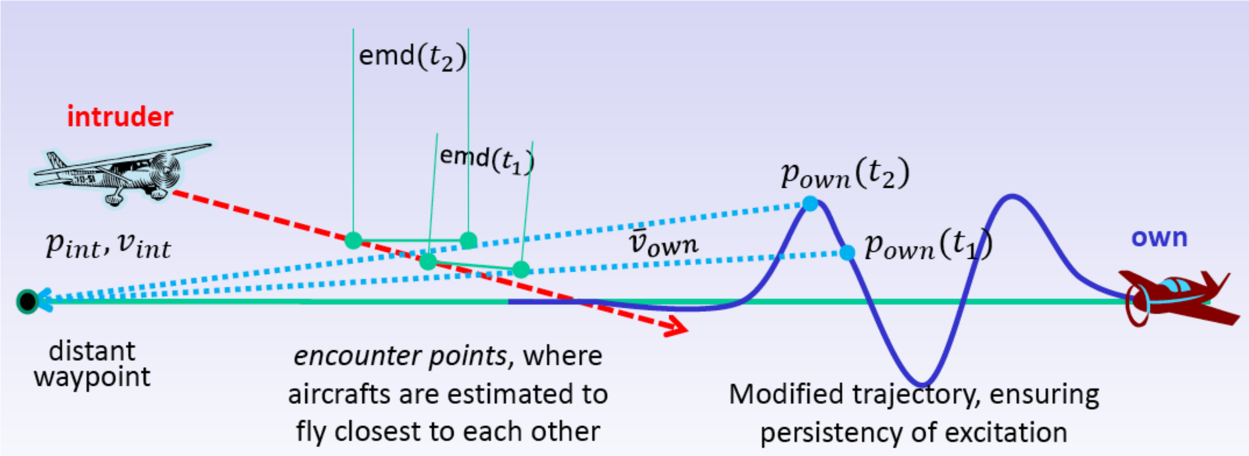

emd(t_2)
intruder
emd(t_1)
$p_{own}(t_2)$
p_{int}, v_{int}
$\bar{v}_{own}$
$p_{own}(t_1)$
own
distant waypoint
encounter points, where aircrafts are estimated to fly closest to each other
Modified trajectory, ensuring persistency of excitation

Figure 5. Closest Encounter Calculation (Time To Collision and Est. Miss Distance)

$$t^* = \arg\min_t V(t), \qquad d^* = V(t^*)^{1/2}, \qquad V(t) = \|p_{\rm int} + v_{\rm int}t - p_{\rm own} - \bar{v}_{\rm own}t\|^2$$

By taking the gradient vector of $V(t)$ and equating it to 0 the parameters t^* and d^* can be computed directly, without on-line optimization. In the current investigation it is assumed that the intruder reaches the encounter point of $X_i = [0, 0, -20000]$ m (North, East, Down reference frame) at 75 s, while the future waypoint is located at the same height as the encounter point at $X_{wp,n} = [0, -150, -20000]$ m, 150 m ahead of the own aircraft and the same distance behind the intruder aircraft. But due to obvious reasons this waypoint can be selected differently, not to take into account the unknown time of encounter.

B. Decision Making Algorithm and Connection to the Flight Control System

Due to common air traffic rules and to simplify the task of the decision making algorithm, four pre defined avoidance maneuvers are available in case the onboard decision making algorithm detects a potential collision. Selection is based on the estimated miss distance calculated above: in case the average over a period of 10 s of the estimated miss distance d^* exceeds 100 m, no action is taken, on the other hand if it is below this threshold the following actions are taken: 1.) the vector d^* is projected onto the future predicted normal plane of the own craft path, 2.) the resulting projection on the normal plane can lie in four different quadrant, as shown in Figure 7, and based on this the direction of encounter is determined, 3.) new waypoints are issued to the flight control system: the first one has 100 m offset from the middle point between the current own position and the predicted own position at timet^* , when the closest encounter occurs, the second waypoint has the same offset (either towards left, right, up, or down) but from the predicted own position at

Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703

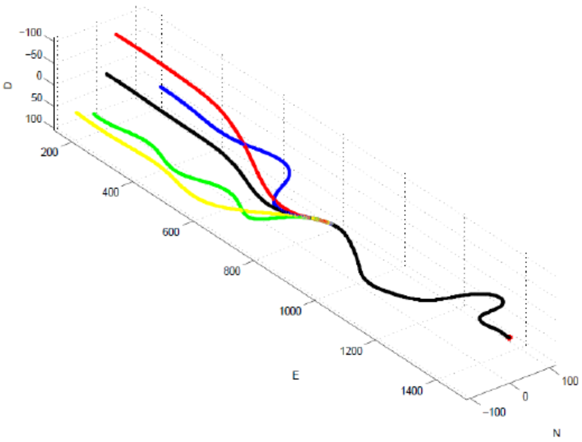


Figure 6. The original path and the corresponding avoidance trajectories, in straight level flight towards West

timet^* as shown in Figure 6. This forces the UAV, that if time is available for avoidance maneuvering, that the minimum separation is met within $0.5(t - t^*)$, allowing a sufficient safety factor. The command issued by the logic is calculated based on the offset (as shown in Figure 7), in case the predicted collision scenario involves the intruder flying above the own craft, the issued command is to fly towards down, in case the encounter is from the left, than the issued command is to move towards righ, an similarly in the other cases.

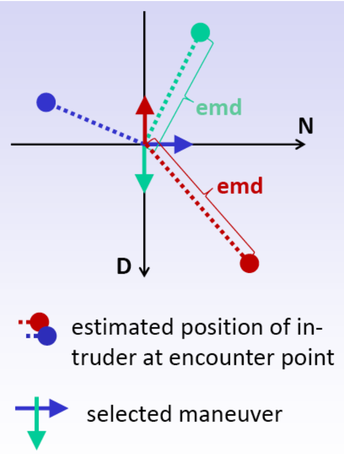


Figure 7. Decision Making Logic (Based on the projection of the Est. Miss Distance)

V. Simulation

A. Simulation environment

To analyze and demonstrate the functionality of a vision-based system it is important to have real or at least realistic camera images available. In indoor tests these images are generated by an appropriate simulator, which is able to provide realistic 3D views from the flight scenarios. The FlightGear simulator is chosen for this purpose, since it has a flexible interface with SIMULINK via the Aerospace blockset and the software is open source, hence the interface with the image processing algorithm can be customized as well.

For the sake of calculating precise input data for the estimation algorithm the FlightGear program has to be calibrated. To asses the rendering capability of Flightgear, first the Field of View (FoV) and aspect ratio settings are measured. A Cessna 172P aircraft model was used for these measurements due to the popularity of this light weight airplane type. It is foreseen that small UAVs will share airspace with similar type of aircrafts, and most of them have no radar and use visual sensing for collision avoidance. The wingspan of

Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703

Cessna $172P$ is 11 m. The FoV of the rendered image from the following model is calculated

$$FoV = \frac{2\arctan\frac{5.5}{d}}{w_a} w$$

where where FoV is in degree, d is the distance of the two aircrafts in meters, w_a is the measured width of the aircraft in pixels, w is the width of the rendered image in pixels. From the measurements it turned out, that two regions can be defined from rendering point of view: a far region (d>20m), where this model can be used and a close region (d<20m), where distortions of this model are observed. The images can be used without post filtering or additional compensation, since the far region is of interest in our case, and no emergency situation is considered yet. We have to detect the other aircraft far enough to do the avoiding maneuver. It also turns out that FlightGear does not take care about the aspect ratio parameter. If geometry is not 1:1, the FoV is set to the bigger size and the image is cropped by FlightGear. According to the measurements that are not detailed here, it can be asserted that the geometry used by FlightGear is linear perspective.

In Fig. **??**. the diagram of the simulation environment is shown. The flight control is running on hardware in the loop system, shown at the upper left corner. The aircrafts are simulated by MATLAB/SIMULINK. For the own aircraft a high fidelity mathematical model has been identified using the measurement data collected from the Ultrastick unmanned aircraft.[20] The intruder is modeled as a simple double integrator. For the own aircraft a trajectory tracking controller has been designed, which runs on an MPC5200 embedded microprocessor.

The flight simulator PC communicates with the image processing computer via Ethernet. On the image processing PC a modified FlightGear is running, which contains the image processing package as an embedded component. The information extracted from the image is then sent to an FPGA unit via USB. The FPGA realizes a Kalman filter and calculates the Motion Prediction data required by the control block. These data are forwarded to the control block by the image processing PC via Ethernet, (see Fig. **??**). The decision logic is than implemented in the Aircraft PC, which than sends the new waypoints to the autopilot realized on an embedded micro controller (MPC5200).

Our aim is to implement the image processing algorithm on the FPGA in a later stage of the project, to reduce the power consumption together with the mass and volume of the system and to mature it for onboard implementation.

B. Simulation Results

The performance of the EKF and UKF based estimation algorithms are compared in different simulations, and the UKF based solution proved to be more stable and robust for the initial conditions and it provided tighter covariance bounds on the estimates, $\pm 1 - \Sigma$ bounds are only $10 - 15m$ 25s after the start of the estimation with the UKF as shown in Figure 8, after performing a few maneuvers exciting the system dynamics. The EKF based solution provided very similar position estimate, but the associated covariance is significantly $(5 - 10$ times) higher.

42 different encounter scenarios with different intruder motions are investigated in the simulations, as shown on Figure 9. The own aircraft flies an up-down-left-right path towards West, starting from $X_{0,own} = [0, 1500, 20000]$ m to ensure the persistency of excitation. The intruder aircraft (Cessna 172P) comes from West towards East, from different initial locations, as shown in Figure 9. The trajectories are determined by two parameters, at time $t = 75$ s the intruder is at $X_{t,int} = [0, 0, 20000]$ m, at the ideal collision point, and secondly the constant velocity vector of the intruder is rotated with azimuth angle β an elevation angle α, where $\alpha = [-10 : 5 : 15]$ deg and $\beta = [-45 : 15 : 45]$ deg, which determines the initial coordinate as well, (see Fig. 9). α and β are selected according to the size limitation of the FlightGear window.

It is also assumed, that the intruder is approaching with $20m/s$, while the nominal speed of the own UAV is also $20m/s$, but can vary with the trajectory. The resolution of the images captured from the FlightGear simulator are 2208×600. Applying this resolution an intruder of size of a Cessna can be reliably detected from a range of 3000m. In the examined scenarios the intruder was therefore visible from the very beginning of the simulation.

The accuracy of the measurements provided by the camera and the image processing unit are analyzed in Fig. 10. The true and measured values of the unit vector coordinates are compared in two representative scenarios. The figures show that the measurement follows the true values with acceptable precision, the

Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703

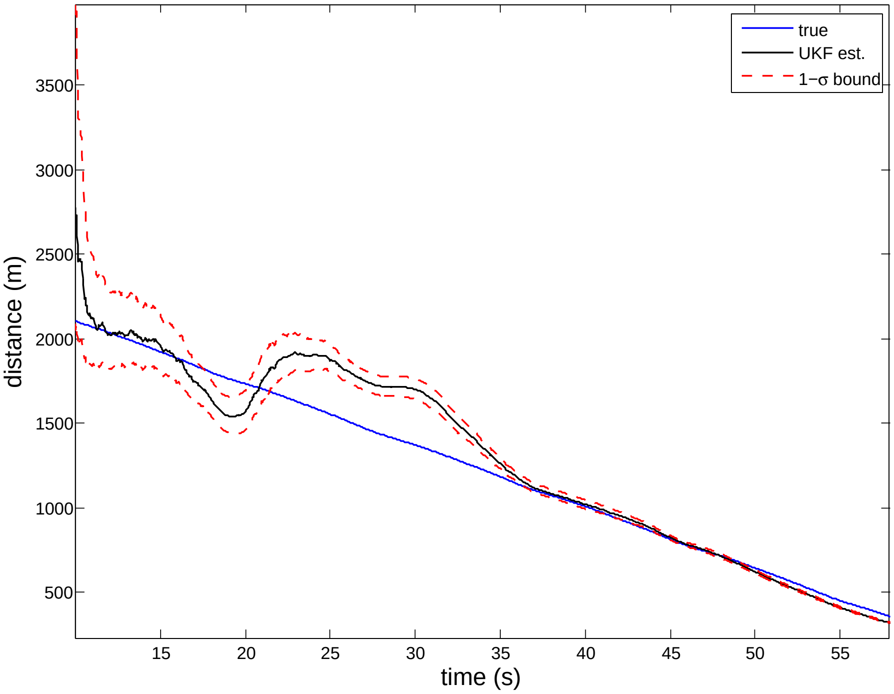


Figure 8. Simulation of the UKF based estimation with covariance bounds, single run.

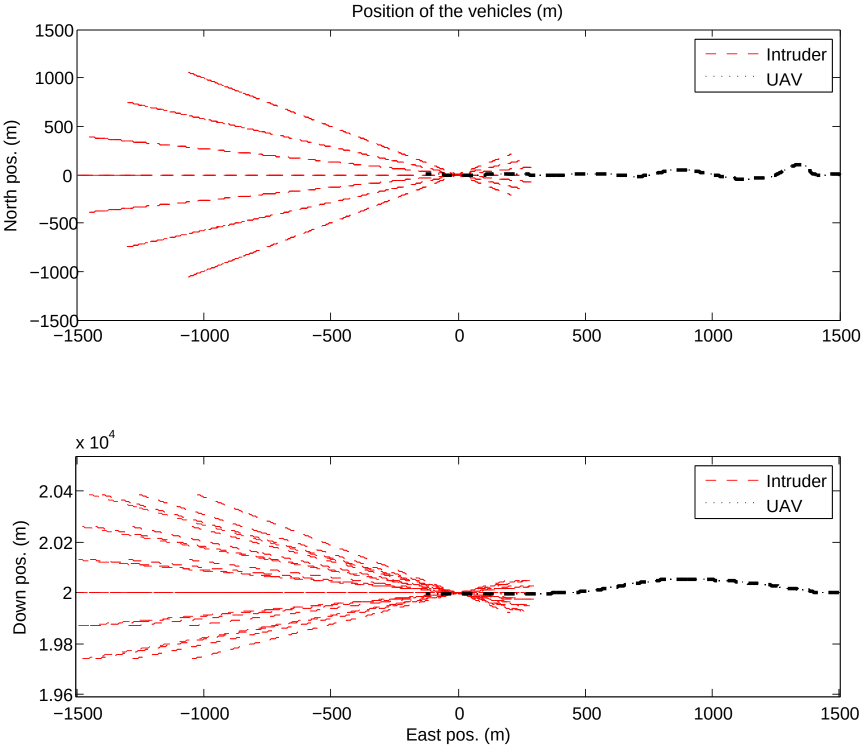
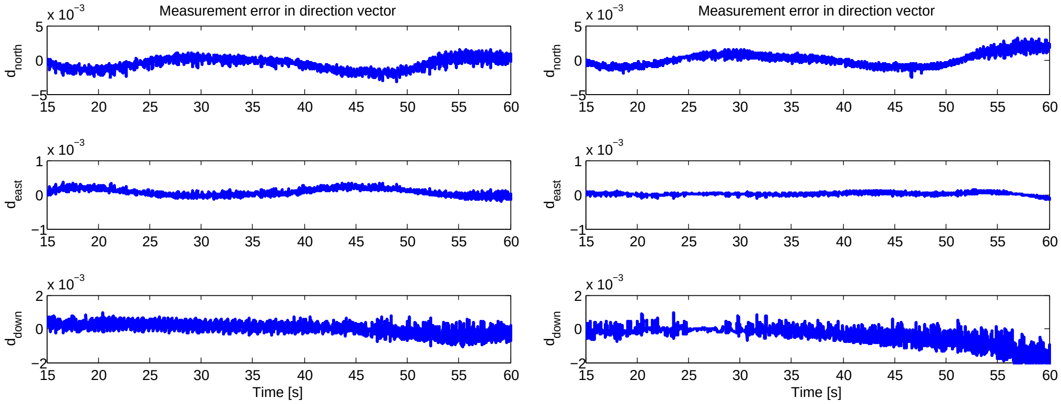


Figure 9. Encounter scenarios example

Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703



(a) SCN2

(b) SCN4

Figure 10. Coordinates of the direction vector - true values and their measured counterparts generated by the image processing unit.

maximal error is only $4 \cdot 10^{-3}$ radian, detected in the north coordinate of u in the first scenario.

The EKF and UKF filters were run with the following parameters:

Initial state (EKF,UKF)	$u(0) =$ actual value from im. proc., $\dot{u}(0) = 0$, $r(0) = 1000, \dot{r}(0) = -20,\ b(0) = 15$
State covariance (P_0)	$0.1 * \mathrm{diag}([0.01,\ 0.01,\ 0.01,\ 0.01,\ 0.01,\ 0.01,\ 0.01,\ 0.01,\ 1])$
Process noise covariance	$\mathrm{diag}([0.03,\ 0.00015,\ 0.003])$
Observation noise covariance	$\mathrm{diag}([0.0001,\ 0.0001,\ 0.0001,\ 0.1])$

These initial values corresponds to physical quantities, taking the worst case the initial estimate for range is 1000 m, the wingspan is 15 m, while the closing speed between the two vehicles is 20 m/s, the velocity of the own craft. It was demonstrated previously[16] that based on the measurement data from the image processing unit, corrupted by delay and quantization, the filters (both completed with an LS estimator) are able to produce estimates for the position and velocity of the intruder.

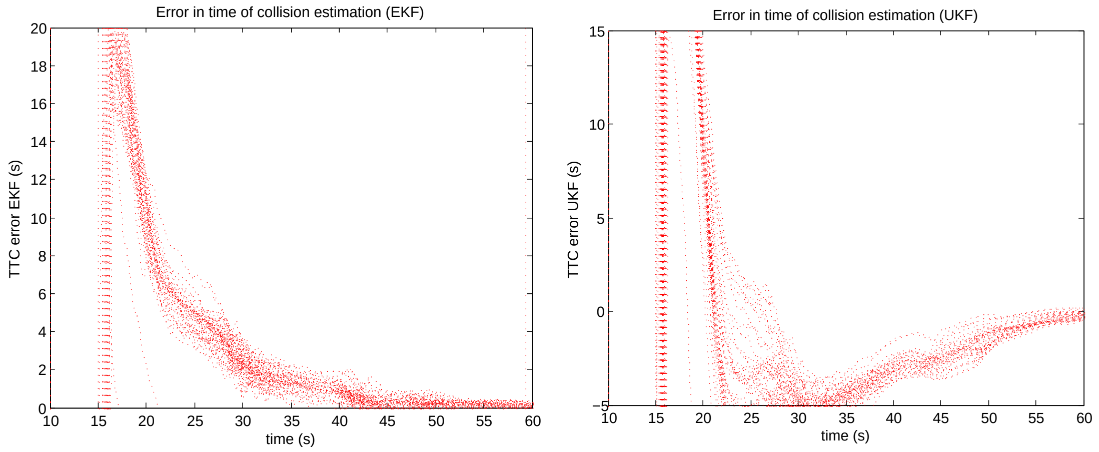
The more important quantities here are the time to collision and estimated miss distance, which are calculated from the estimated position and velocity components. The difference between the true and predicted time to collision values are depicted in Fig. 11. It can be seen that both EKF and UKF filter based predictions provide reasonable good estimates, and the values are within 2 s from each other. On the other hand, it has to be noted, that the EKF based solution converges faster and the solutions are less spread out than in the case of the UKF. Moreover the filters converge in all cases less than 35 seconds, note that the estimation starts at 10 s, not form the beginning of the simulation. At the time of acceptable convergence the distance between the two aircrafts is at least 1000m, therefore 25 seconds remains to launch and execute the avoidance maneuver (if needed).

The difference between the calculated and predicted miss distance values are depicted in Fig. 12. It is important to notice, that the error in predicting a close encounter is less than ±20 m, even when the aircrafts are almost 2000 m apart. In the EKF case one estimate converges slower, which is due to the fact that the image of the intruder plane leaves the rendering window, and held constant until found again.

Based on the calculated miss distance the decision logic is issuing avoidance commands in different directions, depending on the encounter type (see Fig. 13). In the current investigation the decision is made always at 40 s, given a 10 s long averaging preclude it, and based on our experience it takes at least 20 s for the estimation algorithms to converge.

The predicted and calculated miss distances of the individual simulations are depicted in Figure 14. It can be seen, that all encounter scenarios, without the avoidance mode activated are violating the required 100 m minimum separation distance. On the other hand, when the avoidance mode is active 41 out of the 42 simulation cases are outside the required minimum range, with the only exception of the case when the image of the plane goes out of the FlightGear window.

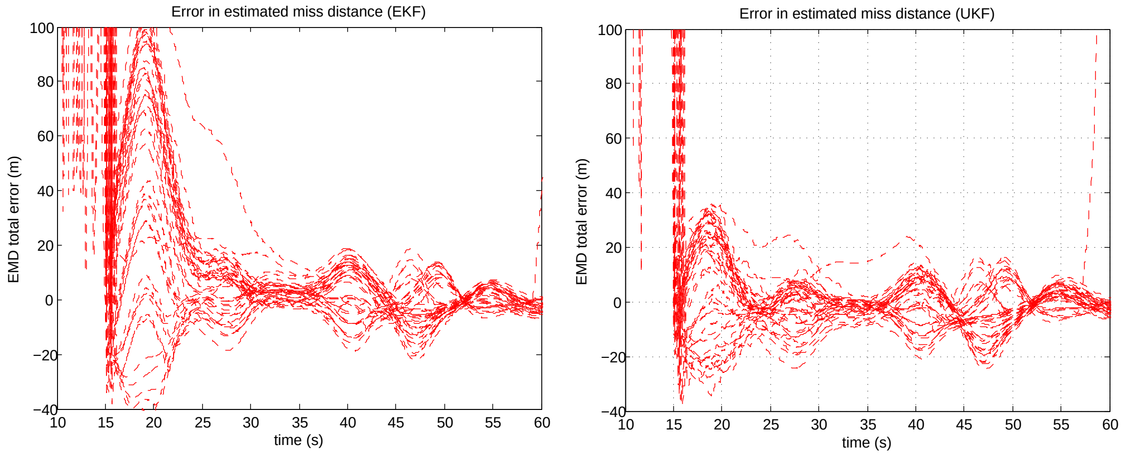
Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703



(a) EKF

(b) UKF

Figure 11. Time to collision estimation error



(a) EKF

(b) UKF

Figure 12. Estimated miss distance error

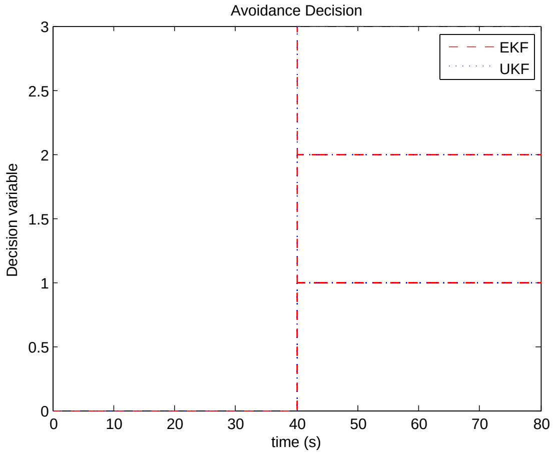
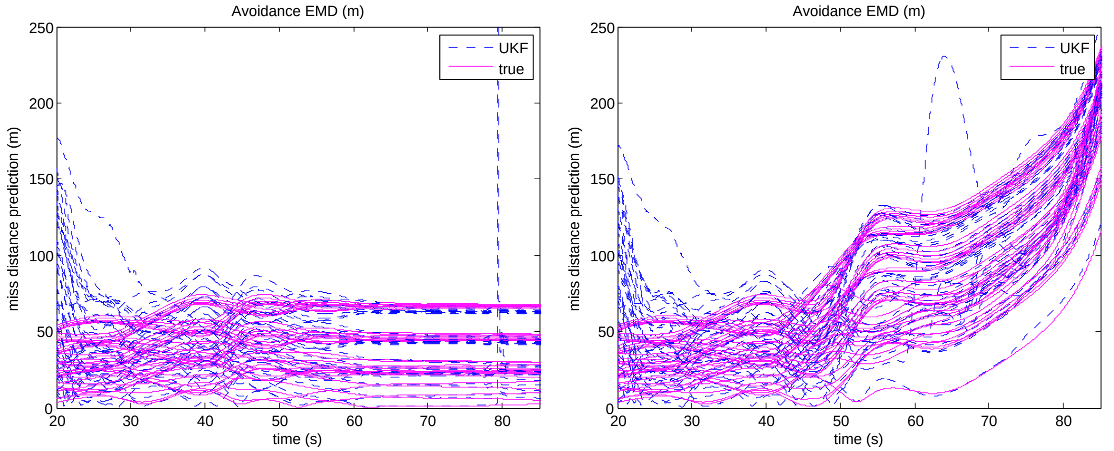


Figure 13. Decision variable of the avoidance subsystem

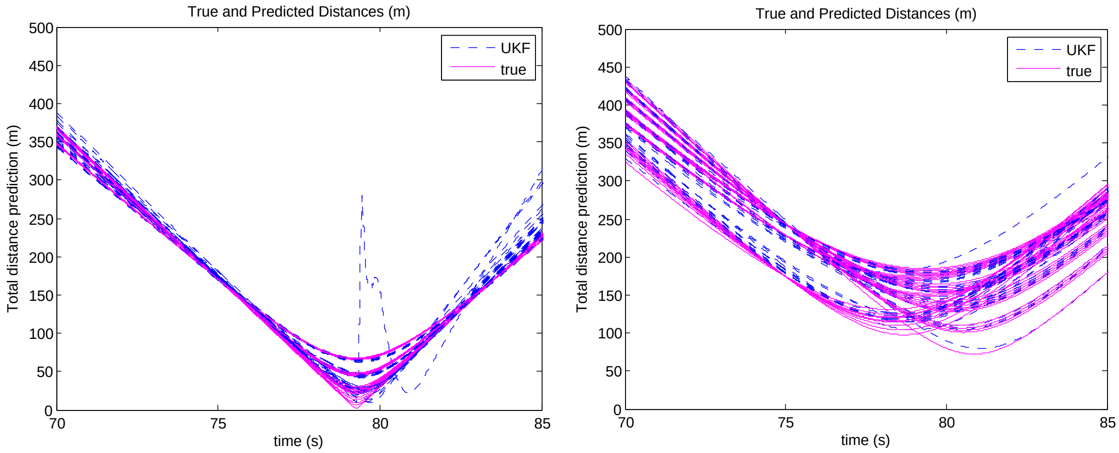


(a) Open loop, without avoidance

(b) Closed loop, avoidance active

Figure 14. Predicted and true miss distance between vehicles

The miss distance prediction is further strengthened with Figure 14, where the total distance between the two aircrafts can be seen as a function of time. This leads to the same conclusion, that in all but one situations the avoidance system is providing adequate separation, while supporting also the applicability of the proposed method of calculating the predicted miss distance and time of collision.



(a) Open loop, without avoidance

(b) Closed loop, avoidance active

Figure 15. Predicted and true total distance between vehicles

The simulation results confirmed the feasibility of the vision only sense and avoid system, even when the measurements are obtained from a realistic scene rendering engine. Moreover, the estimator performance showed sensitivity to the ownship maneuvers. A promising way of assessing the estimator performance is by minimizing the trace of the state estimator covariance matrix by appropriate choice of guidance inputs. Further research is necessary on this topic to address the problem in a noisy environment, when convergence times are important to leave sufficient time for the avoidance maneuver. It is also demonstrated that a relatively simple avoidance logic, coupled to the autopilot of the aircraft can provide reasonably good avoidance maneuvers. While the problem of constrained field of view, and loosing the track of the encountering aircraft remains an issue.

Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703

Downloaded by Balint Vanek on January 27, 2014 | http://arc.aiaa.org | DOI: 10.2514/6.2012-4703

VI. Conclusion

Development of a sense and avoid system consist of multiple individual problems, worth considering independently, but the system has to be evaluated as a whole. In the present article the feasibility of a vision only sense and avoid system is presented. Moreover, the coupling between estimator performance and flight trajectory is highlighted here, which indicates the need for strong coupling of estimation and flight control onboard the future autonomous UAS. To build a feasible SAA system the potential strengths and weaknesses of all system components has to be understood, hence a comprehensive HIL simulation platform is also laid out in the article. This platform helps to find the best possible SAA architecture compromise, which is computationally feasible for implementation onboard a UAV, while provides the necessary performance required for safe separation in the airspace.

References

[1] Dempsey, M., "U.S. Army Unmanned Aircraft Systems Roadmap 2010-2035," Tech. rep., U.S. Army UAS Center of Excellence, 2010.

[2] DeGarmo, M. T., "Issues Concerning Integration of Unmanned Aerial Vehicles in Civil Airspace," Tech. rep., MITRE Center for Advanced Aviation System Development, 2004.

[3] Cox, T. H., Nagy, C. J., Skoog, M. A., Somers, I. A., and Warner, R., "Civil UAV Capability Assessment," Tech. rep., NASA Dryden Flight Research Center, 2004.

[4] Hutchings, T., Jeffryes, S., and Farmer, S. J., "Architecting UAV sense & avoid systems," *Proc. Institution of Engineering and Technology Conf. Autonomous Systems*, 2007, pp. 1–8.

[5] Fasano, G., Accardo, D., Forlenza, L., Moccia, A., and Rispoli, A., "A multi-sensor obstacle detection and tracking system for autonomousuav sense and avoid," *XX Congresso Nazionale AIDAA, Milano*, 2009.

[6] Shakernia, O., Chen, W.-Z., and Raska, M. V. M., "Passive Ranging for UAV Sense and Avoid Applications," *In fotech Aerospace*, 2005.

[7] Kwakernaak, H. and Sivan., R., *Linear optimal control systems.*, Wiley, 1972.

[8] Anderson, B. D. O. and Moore, J. B., *Optimal Filtering*, Prentice Hall, 1979.

[9] Lefebvre, T., Bruyninckx, H., and Schutter, J. D., "Kalman Filters for nonlinear systems: a comparison of performance." Tech. rep., International Journal of Control, 2001.

[10] Julier, S. J. and Uhlmann, J. K., "A New Extension of the Kalman Filter to Nonlinear Systems," 1997, pp. 182–193.

[11] Watanabe, Y., Johnson, E. N., and Calise., A. J., "Stochastically Optimized Monocular Vision-Based Guidance Design." *AIAA Gudance, Navigation and Control Conference,*, 2007.

[12] Watanabe, Y., *Stochastically optimized monocular vision-based navigation and guidance*, Ph.D. thesis, Georgia Institute of Technology, 2008.

[13] Stengel, R. F., *Flight Dynamics*, Princeton University Press, 2004.

[14] Bar-Shalom, Y. and Li, X. R., *Estimation and tracking: principles, techniques and software*, Artech House, 1993.

[15] Hernandez, M. L., "Optimal Sensor Trajectories in Bearings-Only Tracking," Tech. rep., QinetiQ, 2004.

[16] Vanek, B., Péni, T., Zsedrovits, T., Zarándy, A., Bokor, J., and Roska., T., "Performance analysis of a vision only Sense and Avoid system for small UAVs." *AIAA Gudance, Navigation and Control Conference,*, 2011.

[17] Bryner, M., "Developing Sense & Avoid For All Classes of UAS," *UAV 2007 Conference, Paris*, 2007.

[18] Pratt, W. K., *Digital Image Processing: PIKS Inside*, PixelSoft Inc., 2001.

[19] T.Roska, Kék, L., Nemes, L., and Zarándy, A., *CNN Software Library, (Templates and Algorithms)*, Comp. and Auto. Ins. of the Hung. Acad. of Sci., 1997.

[20] Chai, P. Y., *Synthesis and Validation of Flight Control for UAV*, Ph.D. thesis, University of Minnesota, 2009.