Proceedings of the International Conference on
Information Technologies (InfoTech-2017)
20-21 September 2017, Bulgaria

Invited paper

SURVEILLANCE AND PROTECTION BY DRONES

Leon Rothkrantz

Delft University of Technology, Czech Technical University in Prague
e-mail: L.J.M.Rothkrantz@tudelft.nl
The Netherlands, Czech Republic

Abstract: In modern war fare conventional weapons as planes, tanks and soldiers will be replaced by robots and drones. In the civilian domain drones are used for surveillance. Recently we observed many almost collision of airplanes by drones. Surveillance 24/7 by human operators takes too many human and financial resources. In this paper we discuss surveillance by automated drones and drones modelled as kamikaze pilots. The flight behaviour of drones is modelled after human model.
Key words: autonomous drones, flight bots, modelling F-16 pilots, simulation study.

1. INTRODUCTION

A drone is an unmanned aerial vehicle (UAV), equipped with high definition cameras and is able to survey large areas from the air. A drone can be considered as a flying camera system. A drone can be launched from the ground or from a ship (see Figure 1, 2), and is navigated by human operators in the control room. Actions as launching a missile from a drone will be initiated by human operators. Video recordings are transmitted from the drones to the ground station and the semantic interpretation of these recordings in the given context is again done by human operators. So a drone is under control of human operators and many operators are needed. Human operators in the control rooms are supposed to survey video data 24/7 and to give a semantic interpretation of the observed data.

Nowadays many military researchers are involved in projects to design autonomous drones. At this moment most drones are able to navigate autonomously using maps and waypoints on the map. But detection of dangerous situations and starting defensive or aggressive actions is still under human control. An intelligent drone is modelled as an agent. An agent is an autonomous object able to monitor its environment, to detect and track objects and to reason about their behaviour.

Fig. 1. ScanEagle launched from HNMLS Rotterdam (picture Netherlands Defence)

Fig. 2. AeroVironment RQ-11 Raven launched by soldier (picture Netherlands Defence)

An agent has an intelligent module on board to context sensitive reasoning and is able to choose appropriate actions such as attacking, fleeing away or wait and see. Such a drone should also be able to monitor and evaluate its own behaviour if the (planned) actions are effective and efficient. Complex situations arise when a drone itself is attacked by missiles or chased by enemy drones. A battle of drones under remote control is almost impossible because of time delays.

At this moment there are automated pilot systems available for war planes. Such systems are able to control the flight of an aircraft from take-off, landing and flying from one weight point to another, even under bad weather conditions. An automated pilot system gets information from aircraft sensors, radar, flight plan. At this moment automated pilots systems are used to support a pilot. Unmanned warplanes are still under control of operators at a ground station or pilots at a remote safe place. Similar systems could be used to control the flight of drones during surveillance missions. But the control of a drone during a combat is much more complicated. From sensor information the drone has to assess the situational awareness of the opponent to take an optimal decision based on the situational awareness of the drone itself.

The design of autonomous drones is under discussion. Especially if such a drone will be permitted to launch autonomously its own missiles is a topic of ethical debates. But launching a missile hitting the expected target is also a technical challenge. We are used to the battle of drones in war games and agents in cyberwar. But a battle of drones in the real world is still in our dreams and our research findings presented in this paper is a first step. In surveillance tasks usually light weight drones are used unable to carry weapons. A light weight drone is able to chase an enemy drone. To destroy the enemy drone a kamikaze model can be used. Both drones the attacker and the enemy will be destroyed. In case an enemy drone is threatening costly objects or people the loss of a drone is of minor importance.

The outline of the paper is as follows. In the next chapter we present related work. In section 3 we describe the knowledge acquisition. In section 4 the model of our drones and the intelligent modules such as reasoning. In section 5 we describe some experiments performed at the compound of the Netherlands Defence Academy. We end this paper with conclusion and references.

2. RELATED WORK

The Journal of Unmanned Vehicle Systems [1] was established in spring 2013, and is focused on the fast growing area of Unmanned Vehicle Systems. The journal is broadly themed into four main areas of applications of practical and academic interest: civil, environmental, military, and engineering technology.

In [2] selected authors discuss the many aspects of surveillance ranging from the legal aspects, technical challenges, privacy aspects and possible applications.

In [3] Herwitz et al report that in September 2002, NASA’s solar-powered Pathfinder-Plus unmanned aerial vehicle (UAV) was used to conduct a proof-of-concept mission in US national airspace. The UAV was remote controlled by regional air traffic controllers. The imaging system of the UAV was able to take pictures from a large area. These pictures provide information about irrigation problems, and fertilization anomalies and weed outbreaks.

In [4] Puri discussed the application of UAV in traffic surveillance. Compared to sensors localised at specific places, UAV are very flexible and are able to provide an overview of congestion and traffic jams distributed over large areas. The author discusses a variety of multiple and interchangeable imaging devices. The risk of UAV surveillance systems are discussed and some privacy aspects.

Researchers at TUDelft [5, 6, 7] are for many years involved in the design and implementation of UAV’s. One of the latest developments is a hybrid drone which is able to ascend vertically and survey some area in a very short time. On 2017 the TU Delft Micro Air Vehicle Lab (MAVLab) will host the first ever anti-drone competition DroneClash. During this competition participants use their own drone(s) to take down as many other drones as possible. This challenge was our motivation to design our surveillance robot able to attack other drones.

The first prototype of a simulated dogfight agent forms the basis of our current prototype of a drone. These research activities fits in a research project on surveillance systems [8, 9, 10].

3. KNOWLEDGE ACQUISITION

At this moment drones are under control of a remote human operator. But during a combat fight the man-machine communication is too slow. Our drone agent should have context awareness, it should form its own ideas about what is going on. Also, it should be capable of reasoning and making decisions what to do. We used F-16C as our basic model. A multiplayer game of F16 Falcon already exists which was a good example of our prototype. But the main reason to choose F16 was the public domain availability of the specifications of F16. This data was converted in UML and translated to if-then rules. We invited experienced F16 pilots to fly the F16 Falcon simulation game and logged all the flight behaviour. This flight behaviour of human experts provide the necessary knowledge to design the flight and combat behaviour of our agent.

3.1. Specification F16 Falcon

A public manual with detailed specification of the F16 Falcon was available as public domain source. In [5, 6] we converted this information using XML language. JESS, the Java Expert System Shell has an option to convert XML code to if-then rules. In the next Figure 3 we provide an example of a flight plan in XML

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="flightplan"
targetNamespace="flightplan"
xml:lang="en">
<xsd:annotation>
<xsd:documentation>
This is a schema for a flightplan.
</xsd:documentation>
</xsd:annotation>
<xsd:annotation>
<xsd:documentation>
The root tag containing all steerpoint tags.
</xsd:documentation>
</xsd:annotation>
<xsd:element name="flightplan">
<xsd:sequence>
<xsd:element ref="steerpoint" maxOccurs="unbounded" />
</xsd:sequence>
</xsd:element>
<xsd:annotation>
<xsd:documentation>
The steerpoint tag contains information about the steerpoint,
like the coordinates of the steerpoint, the airspeed at which the
pilot should fly to reach the steerpoint in time, the heading
to fly to the steerpoint, the altitude at which the pilot should
fly towards the steerpoint, the TOS (Time Over Steerpoint) which
is the time at which the pilot should be over the steerpoint and
the distance from the previous steerpoint. It also contains two
attributes, one that says what type of steerpoint it is and one
that contains the number of the steerpoint.
</xsd:documentation>
</xsd:annotation>
<xsd:element name="steerpoint">
<xsd:sequence>
<xsd:element ref="coordinates" />
<xsd:element name="airspeed" type="xsd:integer" minOccurs="0"/>
<xsd:element name="heading" type="xsd:integer" minOccurs="0"/>
<xsd:element name="altitude" type="xsd:integer" minOccurs="0"/>
        <xsd:element name="TOS">
<xsd:restriction base="time">
<xsd:pattern value="hh:mm:ss"/>
</xsd:restriction>
</xsd:element>
<xsd:element name="distance" type="xsd:integer" minOccurs="0"/>
<xsd:element name="action" minOccurs="0" maxOccurs="unbounded" type="xsd:string"/>
</xsd:sequence>
<xsd:attribute name="type" type="steerpointType"/>
<xsd:attribute name="number" type="xsd:integer"/>
        </xsd:element>
```

Fig. 3. Part of a schema for a flight plan in XML

3.2 Open interviews

The researchers made a list of 25 open questions. These questions were discussed with pilots individually in open interviews. The interviews were logged and contained much of additional information which was not discussed in the manual for example. As an example we present five questions-answers in Figure 4.

Q: What is the most important instrument that is used to stay on course?

A: The route between two steer points is called a track. When you follow the INS steering to the next steer point you will fly the correct track. The heading indicator cannot be used to stay on this track, because the heading depends on the wind. The nose has to be turned into the wind to fly a straight path. Therefore visual checks of the environment are very important for navigating to the steer point and for evaluating the performance

Q: Might some missiles be fired without a radar?

A: Most missiles can be fired without a radar lock. It is more difficult though because the distance to the target is not known. It might be too far away or too close. Furthermore the radar will calculate all the data compiled in a graphic display on the HUD called a DLZ or dynamic launch zone.

Q: When a LGB attack is performed by two fighters of which one has to lock the target with his laser, what is the task of the supporting fighter?

A: The supporting fighter will circle at a safe distance around the target while he keeps his laser on the target. The laser arm switch is set to ARM at the latest moment possible because otherwise it emits a signal that might be picked up by enemy units. The fighter with the LGB's will toss his bombs in a cone above the target that assures the weapon(s), can ”glide” towards the laser signal.

Q: What does the speed meter display in a deep stall?

A: Airflow is severely disrupted around an airplane in deep stall, therefore are indications on any instrument are highly unreliable. The airspeed should be bigger than zero. It is difficult to say exactly, but it could be plus minus 80 knots.

Q: Should the master lights be switched off when a missile launch is detected?

A: In wartime the lights will always be switched off.

Fig. 4. Examples of questioning-answering from open interview with F16 pilots

4. MODEL

In this section we model a surveillance drone who is able to start an attack with an intruder aerial vehicle. Our starting point is a drone with an automated pilot, able to perform surveillance flights, flying from one waypoint to another. Such an automated pilot was developed after the model of an F16 pilot as modelled in section 3.1. Now we have to extent this model with a software module such that our drone is able to perform a battle such as a dogfight. The design is based on the knowledge from F16 pilots extracted in interviews as described in section 3.2.

During the surveillance flight the video footage is send to the home base. At regular time at the waypoints a picture is taken and this picture is analyzed to detect intruders. When an intruder is detected the human controller at the home base gives permission to start an attack. Our lightweight drone has no weapons onboard and the risk that a missing missile hits unwanted objects is too high. Our drone is assumed to destroy an intruder by hitting it. Our drone will probably be destroyed too but other threatened more valuable objects will be saved. An attack always starts with detection of a hostile drone. After verification the drone will be localised, his heading, speed possible goals will be assessed. Flying directly to our target is too risky that our object will be missed. A better approach is to get behind the enemy and trace and track the enemy (see Figure 5).

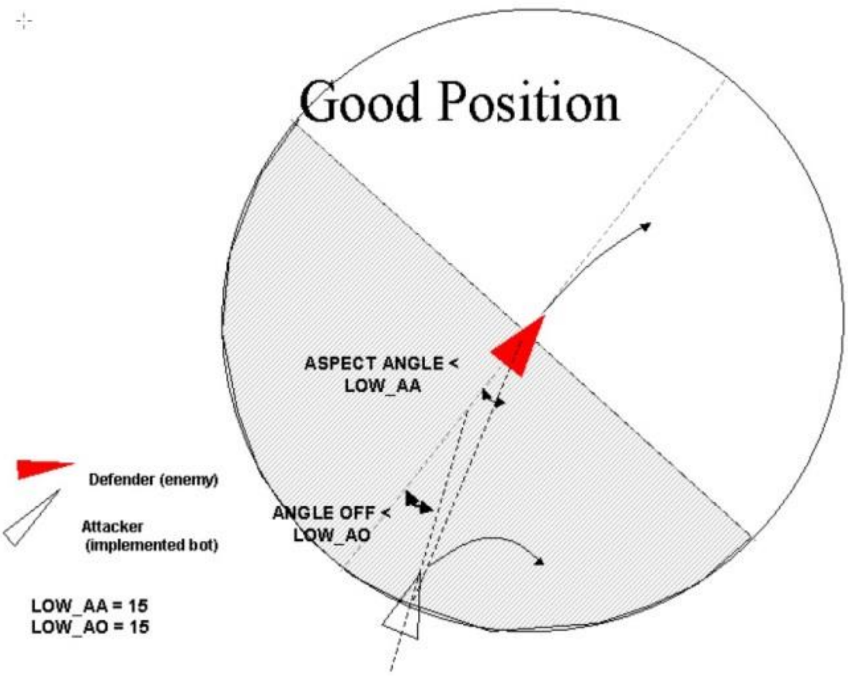

Fig. 5. Compass to navigate the drone

4.1. Object detection

For Object Detection we implemented Background Subtraction. At the waypoints a picture of the scene background is available. If the drone takes a new picture the differences between both pictures are analysed for intruder objects. In case of multiple objects the human operator selects a target object usually. The help of the human operator is needed to prevent attacking friendly objects. Once an object is detected the drone starts chasing and destroying the objects automatically without support of the human operator. But at any moment the operator is able to abort the drone operation.

4.2. Localisation and tracking

Humans are generally very capable of tracking objects, even if they move very fast. Our brain predicts where the object is expected to be in the next moment in time.

Next human observation is very context sensitive, for example a drone cannot move through buildings or trees. For a video surveillance system to work properly, it must be able to track objects too. For this tracking task we use an adapted and improved version of a tool called Predator [11], also known as OpenTLD.

The software tool Predator uses a new technique called P-N learning. The main idea is that the tracker learns appearances of the object (positive examples) and its direct neighborhood (negative examples). The task of the P-N learning algorithm is to learn a classifier that labels each unlabeled sample from a feature set using an a-priori labeled set. A Lucas-Kanade tracker is used to track the object from frame to frame. This tracker is evaluated using the confidence determined by a normalized cross-correlation between the tracked patch and the patch selected in the first frame.

4.3. Reasoning

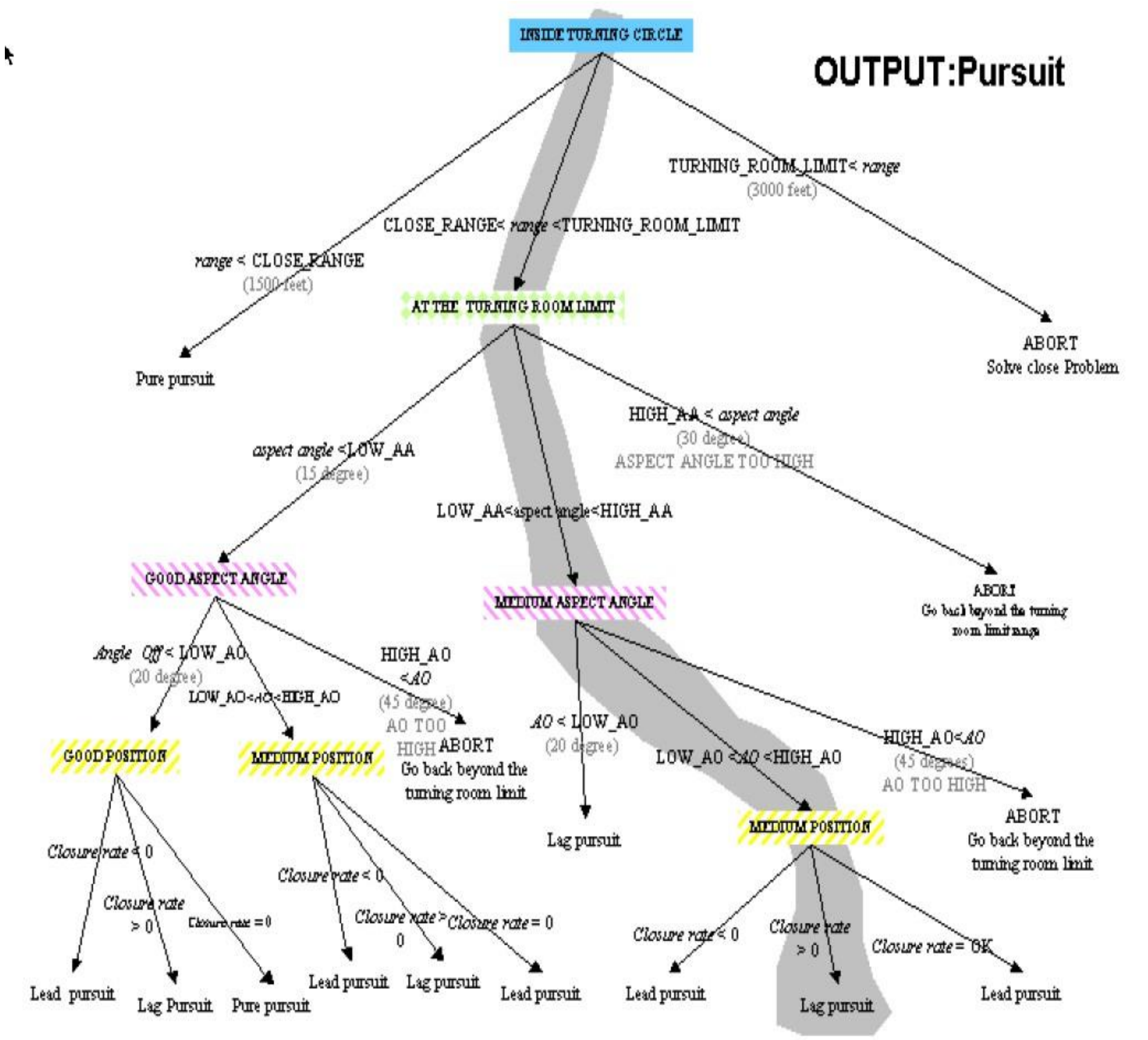

Fig. 6. Example of a pursuit tree

To recognize situations Bayesian belief networks were deployed. To implement a dogfight agent decisions trees were used. A decision tree is actually based on rules, put into the form of a tree. In each node a rule will decide which path must be followed in the tree. After some nodes the path will end in a leave. The leaves contain all actions that can be taken (see Figure 6). By following the right path the system will end up in the leave with the action that is appropriate to the current situation. Decision trees are often used to create the decision-making mechanism in agents.

A rule-based system consists of a lot of rules to describe what to do in predefined situations. It requires a lot of knowledge about all different situations to make explicit rules for the rule based system, for example: 'IF an enemy is on my right side THEN make a turn left'. In our case rules were defined by experts, extracted from combat data or from the F16 flight control manual.

In Figure 7 we display the different levels of skill-based behaviour, rule based behaviour and finally knowledge based behaviour. On the highest level of cognitive reasoning human observers are involved. The final decision to attack should be supported by a human operator. But other ways of knowledge based behaviour is already implemented in the agent. A difficult problem is to assess the impact of the performed actions. A basic level of consciousness and awareness of the situation is implemented in the agent.

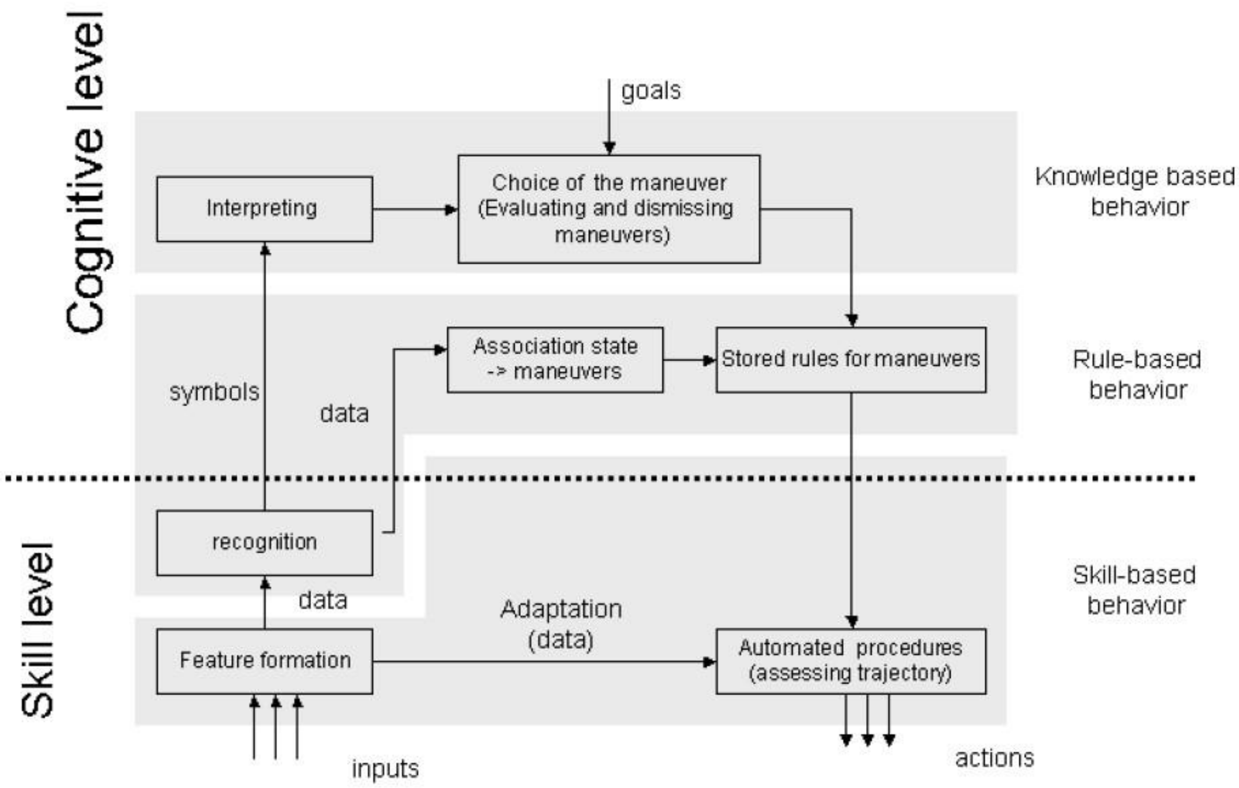

Fig. 7. Adapted model of Rasmussen

5. EXPERIMENTS

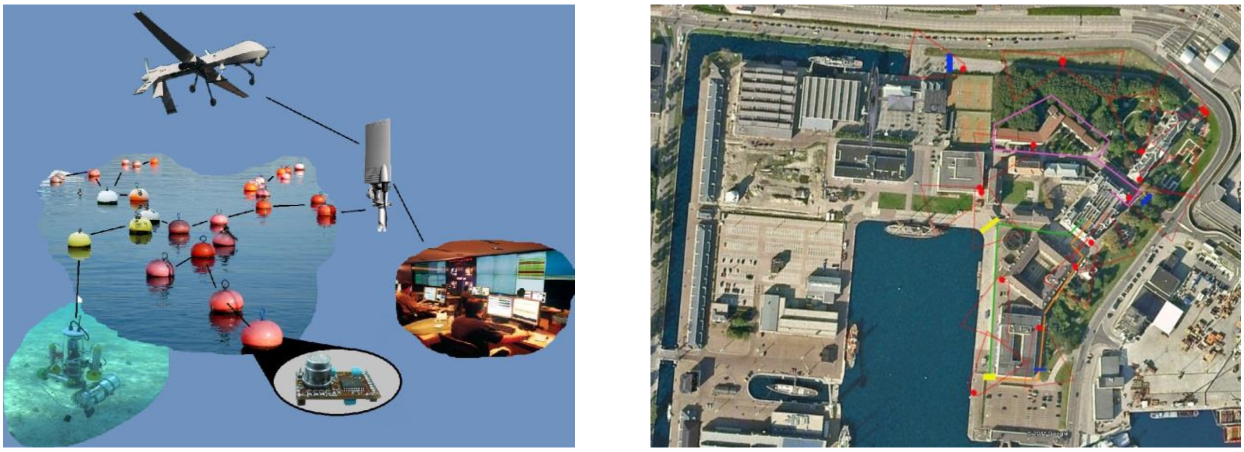

Fig. 8. Surveillance flights of drones over water and military area at Den Helder

As our target area and test area the military area of the Netherlands Defence Academy around the military harbour was chosen (see Fig. 5). The area is surrounded by gates with warnings of high risk areas. The entrances are guarded by safety guards. Nevertheless it happens regularly that drones equipped with cameras survey the harbour taking pictures/video of military vessels which is strictly forbidden. The next step could be a terroristic attack with armed drones or floating mines in the harbour.

Our drone was able to perform surveillance flights at regular times. The route is composed by a sequence of waypoints. During the flight on-board video recorder takes recordings from the environment. At the waypoints a picture is taken and send to the home base. This pictures are analysed automatically and searched for suspicious objects. In case a suspicious object is detected the assistance of a human operator is needed. The operator has to take the decision to survey the object for some time, to perform a shine attack to hunt away the intruders or to destroy the detected enemy drone. Destroying the hostile drone is a kamikaze act. Our designed drone has no weapons on board. It will detect and track the hostile drone and finally try to collapse. A second drone will be used to survey the battle field and direct a safety car of guards.

The combat behaviour of our drone has not been tested in real life yet but only in a simulated environment. Especially the dogfight attacks has good results as reported in [7]. Hostile objects were attacked from behind. That gives good results in tracking and tracing and finally hitting and destroying the enemy drone.

The military harbour borders to open sea. During surveillance it happens that some boats by mistake or on purpose enter the military harbour. It proves that a visible surveying drone circling around an intruder chased away the intruders. Next future more advanced drones will be used to survey the coastline. Surveillance by helicopters or boats takes too much efforts.

6. CONCLUSION

The goal of the paper was to design an agent who was able to navigate a drone and perform combats and destroy intruders. We took F16 Falcon as our model. Thanks to a detailed manual of F16 we were able to model our agent such that he was able to perform surveillance flights along waypoints. Interviews and test flights with F16 pilots in a simulator provided the underlying knowledge to model combat behaviour such as a dog fight.

The first real life test were promising. A problem was the limited reach of the drone because of the limited capacity of the light weight batteries. We expect that the intensive research in the electrical car industry will provide improved batteries next future. It was impossible to fly the drone under bad weather conditions. We expect that more robust drones will provide a solution. At this moment we are involved in the design of a swarm of drones and communication with UAV's on the ground and with the network of surveillance cameras. Next future we expect to test a network of multiple static and mobile agents located at the ground or in the air.

REFERENCES

[1] Journal of Unmanned Vehicle Systems (**2017**). (available at: http://www.nrcresearchpress.com).

[2] Ball, K. Haggerty, D. Lyon (eds) **(2012**). Routledge handbook of surveillance studies. Routledge Taylor & Francis Group, New York.

[3] Herwitz, S.R., Johnson, L.F., Dunagan, S.E., Higgens. R., Sullivan, D.V., Lobitz, J.G. Leung, Gallmeyer, B.A., Aoyagi, M., Slye, R.E., Brass, J.A. **(2004**). Imaging from an unmanned aerial vehicle: agricultural surveillance and decision support Computers and Electronics. In: *Agriculture*, vol 44, issue 1, pp. 49-61.

[4] Puri, A. (**2004).** A Survey of Unmanned Aerial Vehicles (UAV) for Traffic Surveillance. *A Technical Report submitted to CUTR*, USF.

[5] Ehlert, P.A.M., Q.M. Mouthaan, L.J.M. Rothkrantz, (**2003**). A rule based and a probabilistic system for situation recognition in a flight simulator A rule-based and a probabilistic system for situation recognition in a flight simulator. *Proceedings of 4th Int. Conference on Intelligent Games and Simulation (GAME-ON 2003*), Wolverhampton, pp. 201–207.

[6] Mouthaan, Q.M., P.A.M. Ehlert, L.J.M. Rothkrantz (**2003).** Situation recognition as a step to an intelligent situation-aware crew assistant system. *Proceedings of the 15th Belgium-Netherlands conference on Artificial Intelligence (BNAIC 2003*), Nijmegen, The Netherlands

[7] Solinger, D., P.A.M. Ehlert, L.J.M. Rothkrantz (**2005**). A flight simulator dogfight agent for real-time decision-making in the cockpit", *CGAMES*, vol. 7, pp. 129–133.

[8] Rothkrantz, L. (**2014**). Surveillance Angels, *Neural Network World*, vol. 24, pp.1-25.

[9] Hameete, P.S., T. Leysen, T. Van Der Laan, I. Lefter, L. Rothkrantz, (**2012**). Intelligent Multi-Camera Video Surveillance. *International Journal on Information Technologies and Security*, vol. 4, pp. 51-62.

[10] Rothkrantz, L.J.M. (**2014**). Smart Surveillance Systems, Network Topology. In: *Command and Control: Organization, Operation, and Evolution: Organization, Operation, and Evolution*, IGI Global, Hershey, Pennsylvania, pp. 270-280.

[11] Kalal, Z., J. Matas, K. Mikolajczyk (**2010**). P-N Learning: Bootstrapping Binary Classifiers by Structural Constraints. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 49–56, June 2010. 45, 46, 86.