The WITAS Unmanned Aerial Vehicle Project

Patrick Doherty[1] and **Gösta Granlund**[2] and **Krzystof Kuchcinski**[3] and **Erik Sandewall**[4] and
Klas Nordberg and **Erik Skarman** and **Johan Wiklund**

Abstract. The purpose of this paper is to provide a broad overview of the WITAS Unmanned Aerial Vehicle Project. The WITAS UAV project is an ambitious, long-term basic research project with the goal of developing technologies and functionalities necessary for the successful deployment of a fully autonomous UAV operating over diverse geographical terrain containing road and traffic networks. The project is multi-disciplinary in nature, requiring many different research competences, and covering a broad spectrum of basic research issues, many of which relate to current topics in artificial intelligence. A number of topics considered are knowledge representation issues, active vision systems and their integration with deliberative/reactive architectures, helicopter modeling and control, ground operator dialogue systems, actual physical platforms, and a number of simulation techniques.

In W.Horn (ed.): ECAI 2000. Proceedings of the 14th European Conference on Artificial Intelligence, IOS Press, Amsterdam, 2000.

1 Introduction

The WITAS[5] Unmanned Aerial Vehicle Project is a long term basic research project located at Linköping University (LiU), Sweden. The project is multi-disciplinary in nature and involves cooperation with different departments at LIU, and a number of other universities in Europe, the USA, and South America. In addition to academic cooperation, the project involves collaboration with a number of private companies supplying products and expertise related to simulation tools and models, and the hardware and sensory platforms used for actual flight experimentation with the UAV. Currently, the project is in its second phase with an intended duration from 2000-2003.

This paper and the invited talk are intended to provide a status report and overview of the project. During the invited talk, a number of video segments will be shown demonstrating various aspects of the project. The paper is structured in the following manner. In section 2, an overview of the project is provided which includes a discussion concerning research methodology; section 3 describes the physical UAV platforms intended to be used in the project; section 4 considers the actual flight areas used for mission flights; section 5 describes the intelligent vehicle control software architecture used for command and control of the UAV; section 6 describes research with helicopter modeling and low-level control; section 7 describes the core vision module used for image processing; section 8 considers knowledge representation issues; section 9 describes the dialogue system and ground operator interface to the UAV; section 10 describes the simulation architecture and integration of the modules described in the previous sections; and section 11 concludes with some references to other university UAV projects.

2 Project Overview

The long term goal of the WITAS UAV Project is the development of the technologies and functionalities necessary for the successful deployment of a fully autonomous UAV operating over road and traffic networks. While operating over such an operational environment, the UAV should be able to navigate autonomously at different altitudes (including autonomous take-off and landing), plan for mission goals such as locating, identifying, tracking and monitoring different vehicle types, and construct internal representations of its focus of attention for use in achieving its mission goals. Additionally, it should be able to identify complex patterns of behavior such as vehicle overtaking, traversing of intersections, parking lot activities, etc.

The achievement of such an ambitious goal involves dealing with a combination of complex practical and basic research issues together with the integration of research results with existing and newly developed hardware and software technologies used in the project. Successful completion of the project involves (at the very least),

- development of reliable software and hardware architectures with both deliberative and reactive components for autonomous control of UAV platforms;
- development of sensory platforms and sensory interpretation techniques with an emphasis on active vision systems to deal with real-time constraints in processing sensory data;
- development of efficient inferencing and algorithmic techniques to access geographic, spatial and temporal information of both a dynamic and static character associated with the operational environment;
- development of planning, prediction and chronicle recognition techniques to guide the UAV and predict and act upon behaviors of vehicles on the ground; and
- development of simulation, specification and verification techniques and modeling tools specific to the complex environments and functionalities associated with the project.

We will touch upon each of these functionalities in the following sections of the paper.

As stated in the introduction, this is a basic research project with a focus on identifying and developing the algorithmic, knowledge

---

[1] Dept. of Computer and Information Science, Linköping University, SE-581 83 Linköping, Sweden, email: patdo@ida.liu.se

[2] Dept. of Systems Engineering, Linköping University, SE-581 83 Linköping, Sweden, email: gosta@isy.liu.se

[3] Dept. of Computer Science, Lund Institute of Technology, Lund University, SE-221 00 Lund, Sweden, email:Krzysztof.Kuchcinski@cs.lth.se

[4] Dept. of Computer and Information Science, Linköping University, SE-581 83 Linköping, Sweden, email: erisa@ida.liu.se

[5] The Wallenberg Laboratory for Information Technology and Autonomous Systems (Pronounced *Vee-Tas*).

representation, software, hardware and sensory functionalities necessary for deploying an autonomous UAV in and over the traffic and road network operational environment. This particular operational environment was chosen because it is sufficiently complex to require both deliberative and reactive behavior on the part of the UAV and a challenging set of issues for the active vision system, but at the same time, it still contains a great deal of structure, such as the road system, to help deal with some of the complexity of the environment.

Though the project does not include development of a commercial product as a major focus, there are a number of practical applications of potential value associated with this and similar operational environments. One can view a UAV with the stated capabilities as an emergency services assistant capable of guiding fire, police or ambulance personnel to, or through, the scene of a catastrophe, monitoring the current situation and relaying real-time video or still photos, perhaps interpreted, to ground personal. The UAV could also be viewed as a mobile sensory platform in a real-time traffic control systems network, moving to areas of congestion, interpreting the reasons for the congestion and relaying video and information to the traffic control center. Finally, the UAV could be used by police and custom services officials for reconnaissance and monitoring.

When choosing experimental UAV platforms, there are essentially two classes of vehicles to choose from, fixed-wing or vertical take-off and landing systems (VTOL). We have chosen to experiment with VTOL systems due to the nature of the operational environment and mission goals. That part of the sensory platform associated with the vision system currently consists of a digital video camera in a gimbaled housing, but will eventually consist of bore calibrated infrared and digital video cameras in a specially designed housing.

2.1 UAV Research and the WITAS UAV Project

Ground robotics has been an essential part of artificial intelligence research for some time. The use of UAVs as a testbed for both artificial intelligence and on-board image processing research is quite recent and there are only a handful of universities around the world currently doing research in this area. The WITAS UAV project distinguishes itself from many of the other projects in terms of breadth of topics covered and focus on both high- and low-level autonomy and their integration with an active vision and a ground control dialogue system.

A generic UAV setup consists of an air vehicle with payload (quite often a still or video camera), a tactical control station (usually stationary) with one or more humans in the loop, and a data-link between the station and air vehicle used for down-loading images and data and for uploading navigation and camera control commands. A mission plan is quite often represented as a database of waypoint coordinates with associated loitering and data collection commands. The mission plan is either up-linked via the radio link during the mission or already provided when the UAV begins its mission. Data collection activities generally result in a sequence of still images or analog video down-loaded via the radio link or collected upon the UAV's return. The data is generally interpreted by a group of experts manually either during flight or after the UAV's return. Much of the recent research has focused on low-level autonomy and robust flight control issues such as taking off, landing and getting the UAV to fly robustly from one waypoint to another.

The WITAS UAV project focuses not only on low-level autonomy, but also on intermediate and high-level autonomy coupled with an active vision system consisting of digital video and IR cameras as the main sensory components. The intention is that mission goals are provided in a declarative form and the deliberative/reactive system generates the database of waypoints automatically which include loitering and sensory payload commands. The plans are executed and monitored in real-time, sometimes resulting in modification to all or part of the original plan. The on-board active vision system interprets the scene or focus of attention below in cooperation with the reactive and deliberative parts of the overall architecture to interpret the ongoing events below. We assume the use of an on-board geographic information system containing a rich amount of information about the road systems and geographic terrain.

With current technology, it is unrealistic to assume that aviation authorities will permit autonomous aerial vehicles to fly unattended over populated areas without some form of control from the ground station which may include line-of-sight constraints. Consequently, the ground operator is and will remain a vital part of an integrated UAV system in the near future. In order to guarantee clear and concise communication between the UAV and ground operator, multimodal interfaces which enhance such communication play a fundamental role in the overall design of such systems. Part of the research in this project involves just such a system where the ground operator can communicate with the UAV at various levels of abstraction using speech, pointing devices and video viewing. In addition, the actual communication devices may range from standard laptops to smaller PDA like devices. This opens up an interesting set of issues related to the bandwidth of the interfaces and their dynamic management.

In summary, although the full spectrum of issues ranging from low-level control and signal processing to intermediate and high-level autonomy are an essential part of the project, major focus is being placed on the development of deliberative/reactive system software architectures, integration and development of active vision systems and dialogue systems, and knowledge representation issues such as planning, execution monitoring, chronicle recognition, and temporal and spatial reasoning.

2.2 A Typical Scenario

A typical mission goal for our UAV might involve finding, identifying, tracking and trying to discern various patterns associated with a vehicle. Such a mission can be described in terms of achieving the following tasks:

- Locate a vehicle of a particular signature in a designated region. The signature might be given in terms of color and geometry in addition to other distinguishing characteristics.
- Assuming it is correctly identified as the target vehicle, begin tracking the vehicle in a designated region using the on-board GIS to help deal with vehicle occlusion such as going into a tunnel or under a bridge.
- Communicate with ground control for help and advice.
- Attempt to identify certain patterns of behavior such as overtaking, erratic driving, or traversing intersections.
- Return to home base after a designated amount of time.

For this scenario, it may be assumed that the UAV receives as input a vehicle signature, the time (metric) and location coordinates where it was last observed, the designated area of interest, the patterns of interest, and additional time constraints as to the duration of the mission.

This particular scenario is extremely complex and involves robust navigation, high-level decision making, generation of plans, temporal reasoning, dealing with uncertainty both with sensory and qualitative data, chronicle recognition, use of geographic information, anchoring, registration, and signal to symbol transformation of sensory data. Each of these functionalities is a research issue in itself and the integration of these functionalities is probably the most difficult research issue involved.

2.3 Research Methodology

Due to the complexity of both the basic research issues involved in addition to the software and hardware integration issues, the research methodology used has been one of iteration on a base prototype architecture developed early in the project. The iterations are driven by scenarios set up in the operational environment and new functionalities necessary to complete mission goals associated with the scenarios. A simulation architecture has been developed to support initial experimentation and debugging of different parts of the architecture. Video sequences gathered using one of our UAV platforms is currently used for off-board image processing experimentation. Experimental flights combined with simulation experiments will be used throughout the remainder of the project. It is not practical to fly the UAV on a daily or even weekly basis, therefore simulation techniques have an important and major role in the project.

3 UAV Platforms

We are currently collaborating with Scandicraft Systems [5], a university spin-off company that develops autonomous mini-helicopters. The current version in the new series, the Apid Mk III, flew for the first time in October 1999.

The Apid measures 3.63 m from main rotor to tail rotor and is 0.7 m wide. The main and tail rotors measure 2.98 m and 0.62 m, respectively. A 2-cycle, single cylinder modified go-cart motor is used providing 15 Hp at 9500 r/min. Fuel usage averages 5 l/h in hovering mode and 2.5 l/h in horizontal and vertical flying modes. The body is manufactured using carbon fiber/kevlar sandwich material. The payload is 20 kg including fuel. The flight control system software is built around the real-time kernel RTkernel and contains an inertial navigation system with gyro, accelerometers and a tilt sensor, which provide the control system with the platform's attitude and velocity.

Other on-board sensors include a radar altimeter, an IR altimeter, barometer, compass and motor rpm sensor. The platform also contains a differential GPS for positioning. A 1 Watt radio link is used with a 439 MHz frequency band for 2-way communication with the ground station. Information from all sensors can be received from the platform and control commands can be sent to the platform.

The camera system currently being used in experimental flights can contain either a digital video or IR camera. The cameras are contained in a housing with gyro-stabilized pan-tilt gimbals developed by PolyTech/Flir Systems. Panning, tilt and camera zoom can be controlled from the ground via a separate radio link, or on-board using a specially designed interface. A new housing is currently being developed by PolyTech/Flir Systems which will contain bore-calibrated digital video and IR cameras. Figure 1 shows a picture of the APID Mk III with the current PolyTech/Flir camera housing.

We are currently investigating and considering purchase of one or more Yamaha RMAX Aero Robots, a VTOL system developed by Yamaha Motor Company Ltd., Japan [10]. The dimensions of the RMAX are similar to those of the APID, but the RMAX is in production and allows for a larger practical payload of roughly 30 kg which will be required for our new camera housing and on-board system. The production version of the RMAX is currently intended for use as a remotely piloted vehicle and is not autonomous, although an autonomous version is in the works and a prototype has recently been demonstrated.

Figure 1. Scandicraft Systems APID Mk III.

3.1 Use of the Platforms

Experimentation with the Scandicraft Platform is intended to proceed in four stages:

1. In stage one, the platform is being used to collect a library of video sequences over the proposed experimental flying venues. The videos contain footage of vehicle scenarios simulated by project participants driving vehicles in certain well known traffic patterns such as overtaking and U-Turns. The video sequences are then used as raw data for off-board image processing experiments where the quality of the data is close to what would be encountered using the on-board system.
2. In stage two (parallel with stage 1), a mathematical model of the helicopter platform is being derived from the System Build description of the platform used by Scandicraft. This model is being used as the basis for experimentation with, and development of, robust fuzzy controllers for the platform that can be partially validated analytically and through simulation experiments.
3. In stage three, an initial version of the on-board system is being built, but will be initially used from the ground to control the Scandicraft platform. The input to the ground system consists of helicopter state and sensor information in addition to analogue video received via a radio link. The output from the system and to the helicopter platform consists of flight control commands in an autopilot like language and camera control commands which are also sent to the platform via a radio link.
4. In the final stage, the system developed in stage three will be integrated with and placed on-board the platform where both semi- and fully autonomous experimentation will ensue. Due to payload considerations, a different camera system than that intended for the Yamaha platform will be used.

We are currently at the end of stage 2 and the beginning of stage 3 in this part of the project.

Experimentation with the Yamaha platform is intended to proceed in a similar manner, but in this case, an in-house helicopter model developed by our research group in addition to a flight controller and flight command language based on the model is intended to be integrated with the non-autonomous RMAX platform resulting in an autonomous version of the RMAX. The special camera system being developed which is too heavy for the Scandicraft platform is intended to be used with the Yamaha platform. The camera system with specialized housing will be ready by the end of this year.

4 Flight Venues

There is quite a large gap between desirable venues for executing flight experiments from the perspective of the project's long term goals and choice of operational environment, and what is realistically feasible from the perspective of the aviation authorities who are justifiably conservative in approving flight experimentation with experimental UAVs. There is currently a great deal of international interest and movement on this front, where the goal is to define international guidelines approved by aviation authorities for flying unmanned aerial vehicles in controlled airspace and over inhabited areas. It is doubtful that agreement on international or even European guidelines will occur before the end of this project. In the meantime, we are abiding by temporary guidelines set up by the Swedish aviation authorities (SLV) in respect to the Scandicraft APID Mk III platform. The steps toward certification to fly in inhabited areas containing roads, buildings, etc., are as follows:

- Step 1 – Each specific flight experiment and venue has to be approved by SLV under severe restrictions. One can not fly in areas that are inhabited by third parties. A safety zone must be delineated around the flight area and actively monitored for third parties. Guarantees that the flight can be terminated before any third party reaches the flying zone upon entering the safety area must be made in addition to guaranteeing that the platform can not physically leave the safety zone. Several alternatives are use of fly wires or closing fuel valves automatically.
- Step 2 – Flight experiments can be made at arbitrary times, but only in a specific venue and with the same restrictions as above.
- Step 3 – Permanent certification is granted for flying in arbitrary venues under the same restrictions as above.
- Step 4 – Permanent certification is granted to fly in inhabited areas where there are third parties.

Scandicraft is currently allowed to fly under the conditions associated with step 2. We believe they should be able to fly under the conditions of step 3 before the end of 2000. Similar rules will apply for use of the Yamaha RMAX. Currently, there are two venues in Sweden where actual flight experiments can take place, Kvarn, a military area about 40 minutes from LIU, and Revinge, a Rescue Services Training School, in southern Sweden, about 4 hours from LIU. Although not completely ideal for the most sophisticated type of flight experiments we would like to do, each has its advantages and disadvantages and together provide venues which are deemed sufficient to complete legitimate flight experiments in the project.

Kvarn is a military training area near Linköping which is heavily wooded with a number of large clearings. It has a network of dirt roads and several building structures. From the legal flying area, one has views of smaller paved roads and additional buildings.

Revinge is an area near Lund in southern Sweden that contains a Rescue Services Training School. It contains a small town, roughly 1 km^2, with asphalt and dirt roads and a number of different types of building structures ranging from single to three-storied buildings. The town is fenced in and provides an ideal location to do actual flight experimentation without endangering third parties. One disadvantage is the distance from our university (4 hours) and the fact that the terrain is relatively flat. Figure 2 shows an aerial overview[6] of the Revinge area. Figure 3 shows a map over the actual flight area.

Figure 2. Fly-by: Revinge, Sweden

Kvarn will be used for a number of simpler navigation experiments such as following a vehicle on a road or identifying and classifying different types of vehicles. Revinge will be used for a number of more sophisticated tracking and hiding experiments and identifying more complex vehicle interactions.

Unfortunately, both venues rule out high-speed chases over multilaned highways with dense traffic patterns, or long distance scenarios. We will supplement the actual flight experiments with a great deal of simulated experiments. This situation emphasizes the need for creative uses of simulation in our project which will be considered in section 10.

5 Intelligent Vehicle Control Architecture

The Intelligent Vehicle Control Architecture (IVCA) used in the project can be characterized as a multi-layered hybrid deliberative/reactive software architecture with functionality and structure similar in spirit to, but not the same as, the three-layered architectures proposed by Firby [12] and Gat [13]. Conceptually, the architecture can be viewed as consisting of three separate layers, each containing a collection of asynchronous computational processes:

- Deliberative Layer – This layer contains a loose collection of high level services such as planners, trajectory planners, predictors, and

---

[6] This photo is supplied by the National Land Survey of Sweden.

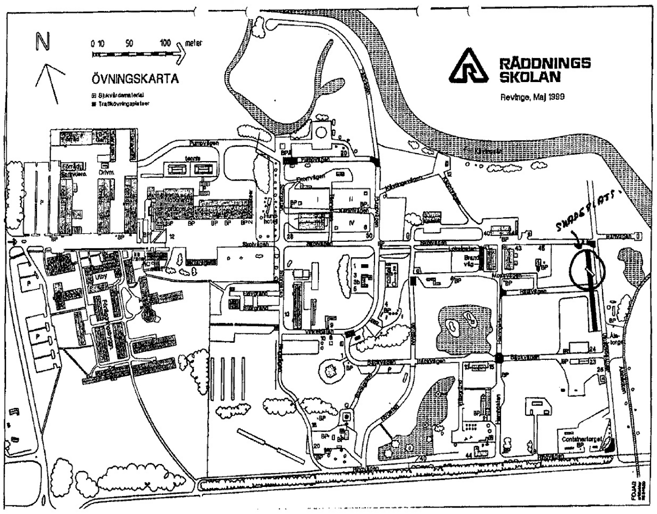

Figure 3. Map: Revinge, Sweden

chronicle recognition packages. These services are called by the reactive layer when its own packages can not achieve mission goals independently.

- Reactive Layer – The reactive layer contains a library of reactive programs specified in the language CONTAPS (concurrent task achieving procedures) developed in the project. A CONTAP can be viewed as an augmented automaton or a collection of triggered rules which have local state and the ability to open channels of communication to other layers or parts of the architecture, including other CONTAPS.
- Process Layer – The process layer is responsible for the concurrent computation of feedback control loops tightly coupling sensing with actuation. It is at this layer that the flight navigation and camera control processing reside.

The architecture contains two main information repositories:

- The Knowledge Structure Repository (KSR) – The KSR contains different types of information associated with high-level deliberative services such as the planner and chronicle recognition packages in addition to the Dynamic Object Repository (DOR) which has the flavor of an object-oriented active database. The DOR is a central and important part of the architecture in that it provides for seamless integration between the signal processing associated with the active vision system and the qualitative processing associated with the reactive and deliberative layers.
- The Geographic Data Repository (GDR) – The GDR is a layered knowledge structure containing spatial data about the operational environment(OE) and access to non-spatial data associated with the spatial structures. The lowest layer contains a repository of digital images over the OE. The next layer contains elevation data correlated with the digital images. The layer above that contains information about the road networks. The layer above that contains landmark and other data.

The different processes associated with the layers in the architecture can access information stored in these structures by opening communication channels to them and querying for information. A number of query languages are being developed for this purpose. Many of the queries implicitly invoke additional processing on data to compute answers. For example, querying the GDR often involves the use of computational geometry, such as in line-of-sight requests.

In the case of the DOR, streams of data may be sent through specially constructed channels with attached filters to access time dependent state variables of dynamic objects such as vehicles identified as being in the camera's focus of attention. These data streams can then be used as input to computational units which process lower level signal information. The computational units provide a means of transforming lower level continuous data into qualitative values associated with state variables in other objects in the DOR, or may simply feed the data flow into CONTAPS in the reactive layer of the architecture, triggering changes in a CONTAPS computational behavior.

Communication between CONTAPS in the reactive layer and helicopter control in the process layer is generally achieved by opening up a communication channel to the navigation service and sending high-level commands in a Flight and Sensory Command Language (FSCL) especially developed for navigating the UAV, requesting services from the vision system and controlling the cameras. The FSCL is similar to an auto-pilot language, but in this case we integrate both navigation and camera platform control into the same language.

For an earlier version of the system architecture description, see Doherty [11].

6 Helicopter Modeling and Control

At the outset of the project, it was assumed that issues regarding low-level autonomy and navigation of the UAV could be delegated to research partners such as Scandicraft or another university group and one could interface intermediate and high-level autonomous tasks to the physical platform in a relatively straightforward manner. This proved not to be the case due to the tight integration required between low- and high-level autonomy and active vision and navigation. We decided to build in-house competence in the area of helicopter control to gain experience and enhance collaboration with Scandicraft and other partners associated with work on the physical platform.

Research and development in this area has progressed along two different tracks:

- Track 1 – We developed our own helicopter model via literature studies and developed a flight controller and flight command language based on this model. Both the model and flight controller are used in a number of simulation experiments. This work provides the basis for simulating that part of the process layer associated with helicopter navigation. We expect to use the results of this track and integrate them with the Yamaha RMAX platform.
- Track 2 – We derived a mathematical model of the Scandicraft platform based on a SystemBuild model provided by Scandicraft. This model is being used for experimentation with more versatile fuzzy-based flight controllers and some experimentation may be done at a future date with these flight controllers in the Scandicraft platform.

In the remainder of this section, we will focus on results and experience from track one.

Skarman [15] is a derivation of an aircraft model. It provides a fairly thorough introduction to rigid body mechanics, and was made as a preparation for helicopter modeling. Skarman [16] is based primarily on literature studies, and is a derivation of a helicopter model. Two basic lessons were learned from this study:

- It is possible to control the magnitude and direction of the main rotor force of a helicopter through the rotor mechanism. All moments required to handle the gyroscopic effects on the rotors are generated by aerodynamic forces.
- In non-transient situations, the helicopter forward and sideways accelerations are determined by gravity in addition to the helicopter pitch and roll angles.

Hence one has access to helicopter acceleration, through which the helicopter can be put to work. This acceleration control is not exact, but outer loops can take care of the errors.

From weapons technology, and in fact from the ancient Greeks, one can borrow and modify algorithms, which use such an acceleration control to bring the vehicle to a given point. These algorithms are optimal in the sense that they minimize the work executed by the so called induced drag. This is of less interest in the helicopter case, but nevertheless, the trajectories generated match the flight requirements for mission tasks quite nicely.

Based on the optimization principle, one can design variations of these algorithms, with additional properties. One can, for instance, make the helicopter arrive at and pass a given point with a given heading. This accounts for sideways helicopter control. In the forward direction, we have designed a non-linear braking algorithm, which uses the helicopter's retardation and acceleration capabilities optimally.

These two control principles are the basic ones used and serve as a basis for the helicopter to execute plans consisting of waypoints, to fly in an anticipatory manner when approaching moving objects, and to catch up and follow moving objects with their own velocities.

Access to these capabilities is acquired through the high level control language called FSCL, which provides the interface to the helicopter control system. The same language is also used for the control of the helicopter camera's gimbals. Here are three sample statements in the FSCL language:

- **Object is 39**
- **Look at object**
- **Fly to object with cruising velocity (28) passing with velocity of object**

Assuming the complete command is a conjunction of the above, the camera will pan and tilt toward object 39, the helicopter will fly to it and then follow it.

7 The Vision System

The deliberative and reactive layers of the architecture communicate directly with the core vision system via the FSCL command language which requests services from it. Intermediate and ongoing results of the image processing and interpretation facilities are stored in the dynamic object repository which serves as an interface between the deliberative and reactive layers of the architecture and the vision system. The feedback loop created between the deliberative/reactive components, the dynamic object repository and the vision system, facilitates the active focusing of the vision system on interesting objects relevant to the current task and modification of the current focus of attention, in addition to the dynamic anchoring of *perceived* objects in the qualitative representation of the current focus of attention.

The core vision module consists of a preprocessor which grabs image frames from a video camera at varying rates of capture interspersed with single images. The image processing task at hand dictates the burst rate of capture and varies continuously. The focus of attention or several foci of attention in the video images are also controlled by the preprocessor. The variable frame rate image sequences are then input into the filtering and segmentation modules. Here they are processed to generate static and dynamic features. Some examples of static features on computed images are color or regions, orientation and complexity or curvature, as defined by divergence/rotation descriptors. Dynamic features are computed from burst time sequence volumes consisting of luminance images. Motion flow is computed from the burst sequence regions.

This low-level feature computation is employed to produce high level descriptors which can then be used for interpretation or matching. Some examples are combinations of line/edge statements for different orientations in vector or tensor form, and combinations of curvature and color in an image. For example, an FSCL request may ask for the vision system to find a vehicle with a particular signature in a certain region. The signature is defined in terms of color and geometry. The vision system might return a collection of potential candidates which match (with some uncertainty), the original signature. The CONTAP requesting the service from the vision system will determine if any of the candidates are suitable by analyzing the identified object's states in the dynamic object repository, checking for similarity. If a positive determination is made, then the CONTAP might request additional services such as tracking or identification of patterns involving several vehicles.

Generally, the vision system tries to determine the position, velocity, color and type of vehicle, or vehicles, in the foci of attention. This involves accurately determining the position of the UAV and camera angles, mapping positions in image coordinates to geographical coordinates, anchoring identified objects into qualitative descriptions of road segments, estimating absolute and relative motions of objects, and indexing or matching the view from the camera with the information in the geographical data repository so as to derive additional information about a situation, or generate additional constraints to assist the operations carried out in the vision system.

The description above is radically simplified. For a more detailed account of these issues and some of the proposed and implemented solutions, see Granlund [14].

8 Knowledge Representation of Actions and Events

In order to understand the observed ground scenarios, to predict their extension into the near future, and for planning the actions of the UAV itself, the system needs a declarative representation of actions and events. This part of the WITAS system will build on our earlier works: 'Features and Fluents' for the analysis of range of applicability for proposed logics, Temporal and Action Logic (TAL) which is capable of representing many of the currently discussed aspects of actions and change, and Cognitive Robotics Logic (CRL) for characterizing imprecise observations, control, and goal-directed behavior.

Besides its use for the fully autonomous operation of the system, this level of knowledge representation is also essential for the dialogue with the operator, and in particular the verbal dialogue. The design of the dialogue system will be discussed in the next section.

In the course of our work on the KR aspect of the WITAS system, one limitation of contemporary research on logics of actions and change has become very apparent: it has only addressed very general issues, such as 'concurrency' and 'causality'. We *also* need concrete solutions to a large number of more specific problems, such as how to represent the different types of vehicles that drive on roads, how to characterize different traffic maneuvers (turn right, turn left, yield, change lane left, etc), or how to characterize the different road structures where these maneuvers take place.

It may be argued that such choices of knowledge representation are specific to every project and every application, and that therefore they do not have any general scientific value. We question that position. If KR research would produce a library of representation 'modules' for various classes of real-life phenomena, such as the ones just mentioned, then later projects would be able to build on them and to use them for their needs. We believe that research in this area ought to be organized in such a way that later work can build directly on the results of earlier work. We will try to initiate such a cumulative chain in the framework of the WITAS project. We have begun this process by cataloguing different traffic scenarios using video sequences which are accessible from our project web site.

9 The Dialogue System

Work with the dialogue system and multi-modal interface for ground operator communication with the UAV and vice-versa is being developed in cooperation with a group under the guidance of Professor Stanley Peters and Dr. Oliver von Klopp Lemon at the Center for the Study of Language and Information (CSLI) at Stanford University [4].

The WITAS multi-modal conversational interface consists of a “community” of software agents (under the Open Agent Architecture) each responsible for some sub-task in the dialogue interaction, along with agents responsible for management functions such as barge-in. Dialogues about multiple topics can be interleaved, in contrast to familiar “form filling” dialogue systems where an inflexible ordering of inputs and outputs is required. Different styles of dialogue are supported in various time and resource bounded contexts.

In more detail the system consists of the Nuance speech recognizer, Gemini (a Natural Language parser and generator), a graphical user interface (GUI), and dialogue agents which update context, resolve references, classify users' communicative actions in terms of dialogue moves, and communicate the interpreted utterance to the UAV. A “meta-agent” allows either users or the UAV to barge-in. On the system side of the dialogue additional agents determine a) how to respond to user dialogue actions, and b) how to inform the user of perceived changes in the world, employing a combination of GUI events and speech synthesis (using Festival).

The interaction between the ground operator and UAV provides a fascinating and wide spectrum of research topics covering traditional dialogue, speech recognition and multi-modal interface research issues, in addition to new topics such as adjustable autonomy and its influence on the mode of dialogue or interface, and the role of the ground operator as an advisor or extra resource for the UAV.

10 Simulation Architecture and Environments

Simulation plays an important role in the WITAS project. Unlike ground robotics research, it is difficult to deploy UAV's on a daily basis. Consequently, a great deal of testing various functionalities is done through simulation. At an early stage in the project, we developed a model-based distributed simulation environment to support the design and evaluation of our software architectures and helicopter controllers. In general, the simulation environment is suitable for testing many of the intermediate- and high-level services which are part of the deliberative/reactive architecture, the interface language FSCL to the helicopter controller and vision system, and to some extent, various capabilities of the vision system itself. Most importantly, the simulation infrastructure is useful for testing integration of many of the software modules with each other.

In the following sections, we will describe the simulation architecture and simulation environments currently used in the project.

10.1 Simulation Architecture

Real-Time CORBA is used as the software communication infrastructure for both the simulation architecture and the IVCA to ensure plug-and-play capability and to provide a flexible research environment. The simulation architecture consists of an object state simulator (OSS) responsible for simulation of the physics of the operational environment and the dynamics of ground vehicles and the UAV in the simulated world. A helicopter control module (HCM) is coupled to the OSS. It contains the model and outer control loops described in section 6. It receives FSCL commands as input from the IVCA module, which contains software implementing the deliberative and reactive layers of the architecture, and outputs helicopter state information to the OSS. The HCM can also output camera control commands to the camera control module (CCM) which in turn provides its device state information as input to the OSS.

The OSS computes a state vector of all dynamic objects. This information is requested by the visualizer which renders changes in the simulated world. The output of the visualizer provides a sequence of image frames from the point of view of the camera. This sequence of frames is sent as input to the vision module (VM). The vision module contains the core vision capability which includes preprocessing, signal processing, and segmentation operations. The VM receives both the image frame sequence from the visualizer and FSCL commands from either the HCM or IVCA module. The VM outputs information about objects in the focus of attention of the camera which is sent to the IVCA and stored in the DOR for use in the IVCA module.

With this type of software architecture, it is possible to couple different versions of helicopter control or even deliberative/reactive architectural components to the core simulator. The setup also permits us to incrementally factor out various parts of the architecture moving from software implementation and emulation to hardware-in-the-loop simulation. A limited experiment in this spirit was performed replacing some of the computation intensive signal processing implemented in the VM with an alternative implementation on SHARC processors which have parallel signal processing capability.

10.2 Simulation Environments

In the following, we will consider several types of simulation environments and approaches to generating them. In the project, we have worked with three:

- Pure Virtual Environments – These simulation worlds are generated using tools such as MultiGen or 3D-Max and are completely virtual.
- Pseudo-Virtual Environments – These simulation worlds combine both real digital photo textures and emulated virtual entities such as vehicles which are super-imposed on the digital photos.
- Enhanced Pseudo-Virtual Environments – These simulation worlds are similar to the previous case, but their terrain models are generated from actual laser sensor data gathered from a helicopter.

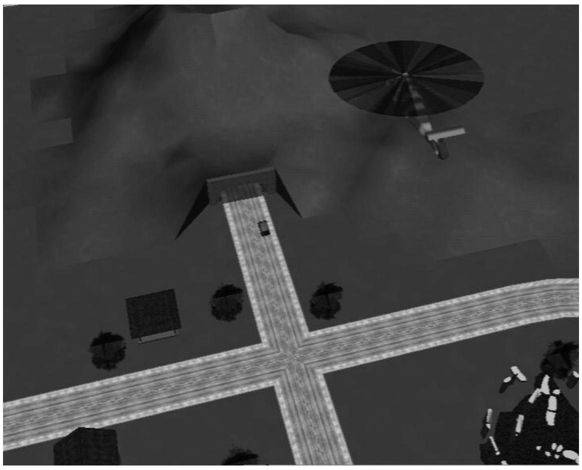

Figure 4. Virtual Simulation; Traffic/Tunnel Scenario.

Figure 5. Pseudo-Virtual Simulation over Stockholm

As mentioned previously, much of the project is scenario driven. A set of scenarios are devised to test various functionalities of the architecture and simulation experiments are performed with iterations on the architecture and incremental modification of the scenarios. For example, the first simulated operational environment consisted of a simple road network with a number of intersections and a bridge and tunnel (see figure 4). The static virtual world was constructed using MultiGen and the simulation was rendered on an SGI Onyx machine. This virtual environment was set up to experiment with the deliberative and reactive components in the architecture, plan generation and prediction, chronicle recognition, and vision system recovery from tracked vehicles occluded by physical obstacles such as the tunnel.

Purely simulated worlds are helpful, but not much of a challenge for the image processing capabilities of the vision system. In the next stage, we did some experimentation with the use of mosaics of digital images as textures in the virtual environment and super-imposed virtual vehicles on the road system in the digital photos. As the helicopter flew over the environment and pointed its emulated camera, the sequence of rendered image frames as viewed from the camera was fed into the vision system and processed. This modification imposed more realistic challenges and additional complexity for the image processing capabilities of the vision system (see figure 5).

Recently, we have entered a new phase in our simulation experiments. We are interested in a very close match between the simulated environment we experiment in and the environment we will be flying over in actual flight experimentation. We are developing a new simulated operational environment for Revinge, one of our test areas, in the following manner:

- Generate elevation data for the Revinge area using a laser sensor. Data accuracy is roughly 1dm - 1cm in the x,y,z direction depending on the type of post processing used.
- Postprocess the data and generate an appropriate elevation model in a suitable format such as intensities in a regular grid or TIN.
- Take digital photos at the same time. In processed form, the data will consist of orthographic photos with 3-4 cm ground resolution.
- Use the photos and elevation data to generate a three-dimensional model of the Revinge area. Building textures and additional features will be generated using ground photos and architectural blueprints of buildings.

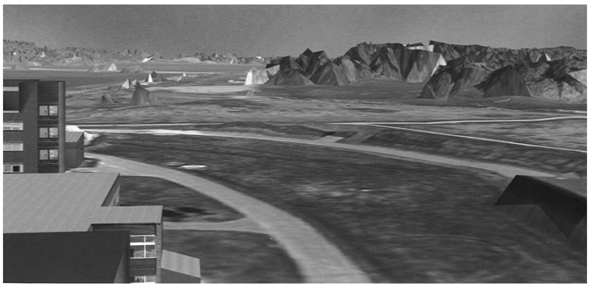

Figure 6. Enhanced Pseudo-Virtual Simulation over FOA and Linköping University

Suitably processed elevation and photo data will be used to generate several of the layers in the geographic data repository used as part of the on-board GIS system for the UAV.

Although we are currently doing this with a manned helicopter and post processing the data off-line for purposes of generating a realistic simulation environment and data for the on-board GIS, one research topic of great potential is to generate models of the environment online and in real-time as the helicopter flies over areas where there is a lack of data. We are currently collaborating with the Swedish Defense Research Institute (FOA) in Linköping, Sweden, in this and the sensor platform area. For an interesting overview of techniques related to the approach above, see Söderman and Ahlberg [17]. Figure 6, taken from [17], shows a semi-automatically synthesized terrain model and virtual reality of an area around our university generated using these techniques.

11 Related Work

The following (non-exhaustive) list of university research projects using UAVs are representative examples of the different focuses on research in the area:

- University of California, Berkeley – BEAR, the Berkeley Aerobot Project [2] is concerned with the development of intelligent control architectures for unmanned air vehicles. They are using multiple flying helicopters as the experimental testbed and are interested in a number of issues such as multi-agent, multi-modal control and visual servoing.

- Georgia Institute of Technology, Atlanta – The Unmanned Aerial Vehicle Research Facility (UAVRF) [8] focuses on the development of a Generic VTOL UAV testbed that may be used to flight test other research projects such as advanced controllers, fault-tolerance algorithms and autonomous operation algorithms.
- Carnegie Mellon University – The Autonomous Helicopter Project [3] is concerned with development of a vision-guided robot helicopter which can autonomously carry out a well-structured set of mission goals in any weather conditions and using only on-board intelligence and computing power. This project included the Haughton Crater Mission.
- Stanford University – The Hummingbird Project at the Aerospace Robotics Laboratory [6] has the goal of demonstrating the practicality of using inexpensive robot helicopters to perform tasks without the need for highly trained human operators. This group has built its own helicopter, called the Hummingbird, and has competed successfully in the International Aerial Robotics Competition organized by the AUVS (see below).
- NASA – Although the Deep Space I Project [7] does not work directly with UAVs, but spacecraft, this project is worth mentioning due to the similarity with architectures used in the WITAS and other projects, and to the many issues both projects have in common such as planning and scheduling in realtime and on-line diagnostics.

For a rich source of related links, see the Association of Unmanned Vehicle Systems (AUVS) site [1].

12 Additional Information

For additional information, access the WITAS UAV Project's web site [9]. In addition to publication lists and textual descriptions of the project, a growing body of video sequences pertaining to the project are continually placed on-line. These sequences may be of interest to groups doing research in related areas and could be used as raw data or benchmarks for experimentation.

The reference list in this paper is far from exhaustive and there is a great deal of related work in many of the research areas touched upon in this project overview. We refer the reader to our web page and other technical publications for pointers to related work.

ACKNOWLEDGEMENTS

This project is funded by a generous grant from the Knut and Alice Wallenberg Foundation, Sweden.

Thanks to Oliver von Klopp Lemon for contributions to the dialogue systems section.

Many current and previous project members have contributed to results in this project. Rather than list all their names, we refer you to our project web page in section 12.

REFERENCES

[1] AUVS– The Association of Unmanned Vehicle Systems . `http://users.erols.com/auvsicc/index.html`.

[2] BEAR – The Berkeley Aerobot Project. `http://robotics.eecs.berkeley.edu/bear/`.

[3] CMU – The Autonomous Helicopter Project . `http://www.cs.cmu.edu/afs/cs/project/chopper/www/heli_project.html`.

[4] Computational Semantics Laboratory, Center for the Study of Language and Information (CSLI) . `http://www-csli.stanford.edu/semlab/witas.html`.

[5] Scandicraft Systems, Sweden. `http://www.scandicraft.se/`.

[6] The Hummingbird Project – Stanford University . `http://sun-valley.stanford.edu/users/heli/`.

[7] The NASA Deep Space I Project . `http://ic.arc.nasa.gov/ic/autonomy.html`.

[8] UAVRF – The Unmanned Aerial Vehicle Research Facility. `http://controls.ae.gatech.edu/labs/uavrf/`.

[9] WITAS UAV Project Web Site. `http://www.ida.liu.se/ext/witas/`.

[10] Yamaha Motor Co., Ltd., Japan. `http://www.yamaha-motor.co.jp/sky-e/index.html`.

[11] P. Doherty, 'The WITAS integrated software system architecture', *Linköping Electronic Articles in Computer and Information Science*, **4**(17), (1999). `http://www.ep.liu.se/ea/cis/1999/017/`.

[12] R. J. Firby, P.N. Propopowicz, and M.J. Swain, 'The animate agent architecture', in *Artificial Intelligence and Mobile Robots: Case Studies of Successful Robot Systems*, eds., D. Kortenkamp, R.P. Bonasso, and R. Murphy, AAA Press/The MIT Press, (1998).

[13] E. Gat, 'Three-layered architectures', in *Artificial Intelligence and Mobile Robots: Case Studies of Successful Robot Systems*, eds., D. Kortenkamp, R.P. Bonasso, and R. Murphy, AAA Press/The MIT Press, (1998).

[14] G. Granlund, K. Nordberg, J. Wiklund, P. Doherty, E. Skarman, and E. Sandewall, 'WITAS: An intelligent autonomous aircraft using an active vision system', in *Proceedings of the 2nd Int'l Symposium on Uninhabited Aerial Vehicles*, (June 2000).

[15] E. Skarman, 'An aircraft model', *Linköping Electronic Articles in Computer and Information Science*, **4**(013), (1999). `http://www.ep.liu.se/ea/cis/1999/013/`.

[16] E. Skarman, 'A helicopter model', *Linköping Electronic Articles in Computer and Information Science*, **4**(014), (1999). `http://www.ep.liu.se/ea/cis/1999/014/`.

[17] U. Söderman and S. Ahlberg. Konstgjorda verkligheter framställs på löpande band, FOA Tidningen, number 6, December, 1999. Published by the Swedish Defence Reserach Institute.