

WITAS Project at Computer Vision Laboratory

A status report (Jan 98)

Thord Andersson, Gösta H. Granlund, Gunnar Farneback, Klas Nordberg, Johan Wiklund
Computer Vision Laboratory
Department of Electrical Engineering
Linköping University, S-581 83 Linköping, Sweden
Phone: +46 13 282460, Fax: +46 13 138526, email: thord@isy.liu.se

Abstract

WITAS will be engaged in goal-directed basic research in the area of intelligent autonomous vehicles and other autonomous systems. In this paper an overview of the project is given together with a presentation of our research interests in the project. The current status of our part in the project is also given.

Key words

WITAS, autonomous systems, active vision, UAV.

1 Introduction

The Wallenberg laboratory for research on Information Technology and Autonomous Systems (WITAS) started working on January 1, 1997 following a decision by the Knut and Alice Wallenberg Foundation to fund the research project.

The laboratory is a consortium of four research groups at Linköping University; three at the Department of Computer and Information Science and one, the Computer Vision Laboratory (CVL), at the Department of Electrical Engineering.

WITAS will be engaged in goal-directed basic research in the area of intelligent autonomous vehicles and other autonomous systems. In its present outline, the major goal is to demonstrate, before the end of the year 2003, an airborne system which is able to make rational decisions about the continued operation of the aircraft, based on various sources of knowledge including pre-stored geographical knowledge, knowledge obtained from vision sensors, and knowledge communicated to it by radio. The system shall be airborne, but is not necessarily in direct control of the aeroplane.

For more information, please visit the homepage of the project:

<http://www.ida.liu.se/ext/witas/eng.html>

2 WITAS projects at CVL

For this project we foresee a number of research problems of particular interest. Although they are traditional parts of the vision and robotics fields, they are accentuated by the use in autonomous systems, requiring particularly robust functionalities. As the use of autonomous systems will increase in other areas, this research work is essential for future development [4].

The particular requirements will give us the motivation and the opportunity to develop and to test a new sensing, processing and representation architecture. The traditional structure of a video camera followed by a processor does not provide the performance required for most demanding situations in robotics. We propose the development of a new vision architecture which will very rapidly switch its focus of attention between different parts and aspects of the image information. It will contain three major units:

- *Actively Controlled Sensing*
- *Fast VLSI Processing Architecture*
- *Multi-Resolution Semantic Scene and Object Representation*

Many of the components of this structure have been tested as stand-alone procedures earlier, but it has not been feasible to integrate all of them into one system. The real gain will however appear as all of them are combined into one system, as any one of them does not separately achieve full performance on its own. Below follows a short description of each item. See also the W3-site <http://www.isy.liu.se/cvl/Projects/WITAS.html>

Actively Controlled Sensing

A traditional approach in robotics has been to use video camera sensing. This is generally suboptimal for most situations, and a more flexible and controllable arrangement is desirable. There are a number of requirements on the sensing process:

- *Wide angle of view*

- *High spatial resolution*
- *High time resolution*
- *High light sensitivity*

In order for the system to obtain sufficient overview for navigation and localization of objects, it is necessary that the sensing is made over a sufficiently wide view angle. To allow robust identification of objects and details, it is necessary that the sensor provides sufficient spatial resolution. Interpretation of dynamics in scenes, such as motion of objects, requires a sufficient time resolution or frame rate.

Fortunately, all these requirements are generally not present simultaneously, a fact which relinquishes not only the demands on the sensing system but furthermore on the subsequent processing system. This illustrates the usefulness of a sensing camera with a number of controllable properties:

- *Controllable integration time to adjust exposure*
- *Controllable frame rate to adjust time resolution*
- *High resolution sensor element*
- *Region of interest (ROI) control of readout from sensor*
- *Variable zoom optical system, or multiple focal length systems*
- *Fast control of camera visual field orientation*

Not all variables are independent, such as the integration time and the frame rate.

A typical sensing procedure may be that the camera moves around mechanically, while images of wide angle resolution are recorded. There is generally no particular reason to have a high time resolution for this case. As the system detects something of interest it will orient the camera to that direction and zoom in on that part. If it is needed to measure motion, the sensor will increase the frame rate accordingly.

What is important for this process to be useful is that it goes very fast. The visual field of the sensor will "jump around" in the scene, switching between different regions, objects and aspects, very much like we do with our head and eye movements. This allows us to obtain a sufficiently high resolution in all important aspects, without flooding the subsequent processing system with redundant information.

The goal of this subproject is to develop such a controllable camera sensor. Sensor elements and other components having the desired properties are available today, but what is required are the control mechanisms which allow close integration with the processing structure.

Fast VLSI Processing Architecture

The processing to be carried out in the preceding sections is quite formidable. For that reason we plan to implement certain parts of the system using special purpose VLSI circuits to be designed. Together with conventional commercial VLSI circuits they will give an architecture for fast and flexible processing.

The first unit to be implemented as a special purpose VLSI circuit is a fast convolver, which will employ methods for fast separable filters, developed at the laboratory [1]. Together with an architecture which allows fast control of operations and data flows [6], this will give a very flexible focus of attention architecture, where the processing resources can be switched between different parts of the incoming data in a way which is well adapted to the patch-like nature of the input image data.

Multi-Resolution Semantic Scene and Object Representation

Together with the actively controlled sensor goes a very different processing and representation structure, compared to conventional methodology. The data which arrives from the sensor(s) can be viewed as patches of different sizes, rather than frame data in a regular stream and a constant array arrangement. These patches will cover various parts of the scene at various resolutions. Some such patches may in fact be image sequence volumes, giving a sequence of images at a suitable time sampling from a particular part, to allow estimation of the motion of objects. The information from all such various types of patches has to be assembled in some suitable form.

The conventional array form of image information is not very useful in this situation, as the patches will be of different sizes and resolutions. It is also necessary to have the information in some interpreted form to fulfill its purpose to evoke actions. This implies that the information should be in terms of content or *semantic* information, rather than in terms of pixel values. An interpretation of the information is made, and the information is represented as *linked objects*. The more extensive discussion of the methods for representation of objects as linked structures will be made elsewhere [3].

3 Overview of WITAS system architecture

In order to get more than separate, "free floating", subsystems from each of the participating groups, the interfaces between the subsystems are targets for considerable attention. Almost all interfaces and subsystems are already "in the loop", making it possible to detect design-flaws and unrealistic assumptions in an early stage. Another advantage with this approach is that the groups have gained insight in one another's

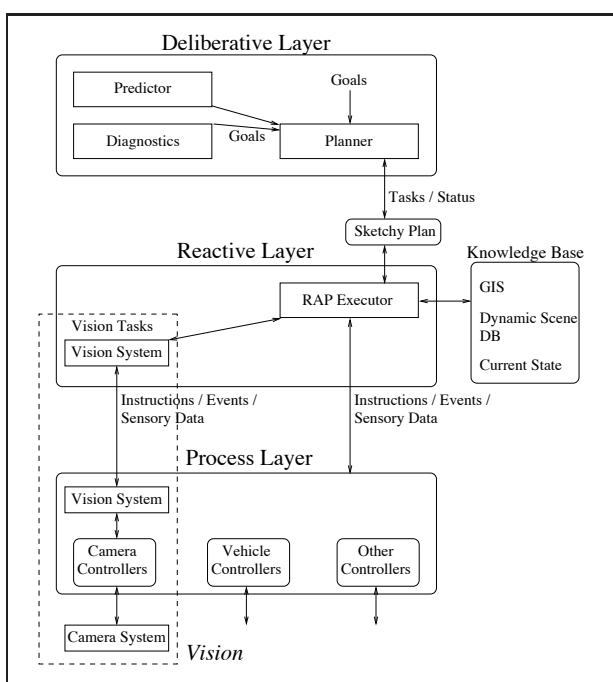


Image 1: WITAS UAV Agent Architecture

fields. The overall structure of the integrated systems is presented in figure 1.

The autonomous decision software architecture, which encloses the other subsystems, consists of three layers; the process, reactive, and deliberative layers (see Fig. 1). This part is developed jointly with the Department of Computer and Information Science, but will for completeness briefly be mentioned here. See also

<http://www.ida.liu.se/ext/witas/eng.html>

The *process layer*, which is the lowest layer, executes a tight loop of a sensing-and-acting type of activity and is characterized by sense/act control modules called *behaviors*. Each behavior is designed to deal with a relatively narrow aspect of the total flight control system. For example, a behavior may represent a particular feedback/feedforward control algorithm such as altitude/heading control. Some behaviors may be designed in such a way so that they encapsulate more complex control algorithms that implement particular flight maneuvers such as left- and right-turns, increasing and decreasing of altitude, etc. Additional behaviors may deal with information gathering and analysis tasks not directly related to the flight control systems which might include identification of map segments, classification of earth-bound vehicles, predicting the behavior of such vehicles, gathering information about weather conditions and their consequent classification. The main part of the vision system belongs to the process layer.

The intermediate, *reactive layer* combines behaviors into *goal-achieving sequential activities* by designating an order on these behaviors based on the continuous sensing of pre-conditions in the environment, where pre-conditions allow or disallow particular behaviors. Essentially, a goal-achieving sequential

activity is intended to represent the variety of ways a particular mission can be executed. Examples of such missions are “follow a car”, “identify traffic patterns”, etc. A program package “RAP”, developed by Firby [2], will probably be used to implement some of the functionalities in this layer. The higher levels of the vision system belongs to the reactive layer.

The highest, *deliberative layer* will perform a number of *deliberative activities*. For example, it may provide guidance to the reactive layer when the latter is unable to realize its goal-achieving activity, i.e., can not continue its execution since certain pre-conditions are not fulfilled as expected, or certain behaviors have failed. This layer can also be augmented with functions that monitor the execution of the activities from the sequential level, monitor the environment, and predict how the execution of an activity will proceed at a future point in time, etc. The deliberative layer will also include limited look-ahead capabilities which will assist in avoiding entry into undesirable states.

4 Overview of Vision system architecture

The vision system provides to the decision system a set of skills or vision-primitives. Each skill provides certain clearly defined information, such as the velocity of a specified object.

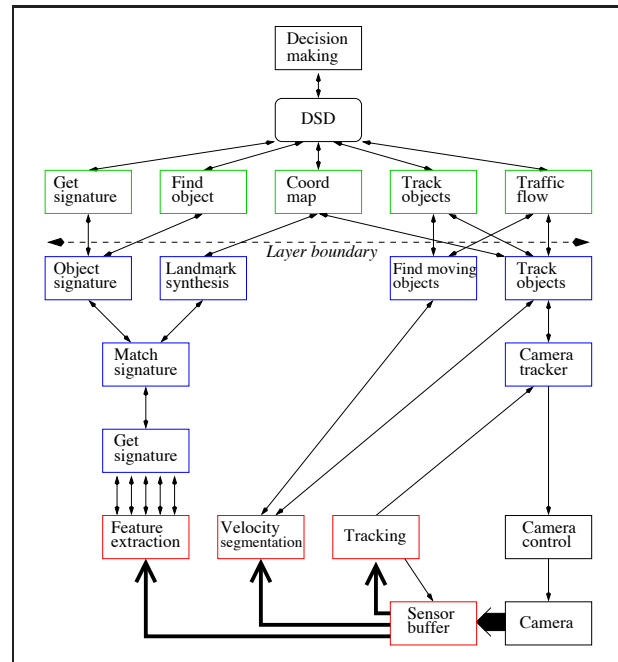


Image 2: Vision system block structure

A skill can be implemented in different ways, each with different characteristics. One version is perhaps very fast but not that accurate, another the opposite. This gives the decision system the opportunity to choose the skill that fulfills the current demand on speed and quality. In figure 2 a coarse overview is presented of the current vision architecture. The

skills, and the rest of vision system that belongs to the reactive layer, are in the top of the figure. The information the skills generate is made available to the decision system through the *Dynamic Scene Database* (DSD). The DSD contains both static and dynamic information about the environment. The static information is, for example, data about roads in the neighborhood. The dynamic information is data describing a non-static object, for example a car. The dynamic information is in addition linked to the appropriate static information, e.g. the car is linked to the road segment it moves on.

Current state of implementation

The structure and the algorithms are and probably will be changing considerable in phase with the development of other systems. The following functions have been implemented so far.

From the camera in figure 2, a number of ROIs are readout to a sensor buffer. At least one of the ROIs is defined as the, known as, *stabilizing ROI*. This ROI (or these ROIs) covers an area on the ground with stationary structures of high contrast. By tracking these stabilizing ROIs on the sensor we can calculate a coarse first approximation of how the image has been translated due to the movements of the UAV since the last frame. With this information we can make a coarse but fast motion stabilization of the ROIs in the sensor buffer by assigning each of them a offset. This “sensor tracking” is the first and the fastest loop in the vision system. The tracking could be implemented in many ways but we currently use a simple mask correlation.

The next step is the *Camera tracker* loop. The movement of the object (or group of objects) we are currently tracking, this could be a car or a road segment, is calculated and the result controls the camera. This calculation can be made either by a fast and simple mask correlation as above, or as the result of the robust and accurate spatiotemporal filtering techniques we have developed at the laboratory [5]. The spatiotemporal filtering is also used to implement the “Find moving objects” and “Velocity segmentation” boxes.

A very important skill is the “Coord map”, see figure 2. All operations below it produces results and relates to *image coordinates* while the decision system always “talks” in *world coordinates*. The task of the “Coord map” skill is to make the transformation between the two coordinate systems. In order to do that, the skill tracks a number of stationary ground structures, called *landmarks*, with *known* world coordinates. From the tracker loop the skill gets the corresponding image coordinates of the landmarks and it can thereby calculate the map. The skill gets its information about suitable landmarks from the GIS-database, which is part of the knowledge database of the system. In geographical regions where there are to few landmarks the skill has to estimate the trans-

formation map based on directional data from the camera (tilt angle etc.) and positional data from the navigation system.

A video containing results from the above mentioned motion stabilization and spatiotemporal filtering is available.

5 Future work within CVL

We will continue to work towards the research goals specified in section 2. In the short term, the following items are important:

- Specification and implementation of skills.
- Specification of implementation of operations in hardware. This is done in collaboration with one of the other participating WITAS groups.
- Specification of camera characteristics

References

- [1] M. T. Andersson and H. Knutsson. Controllable 3-D Filters for Low Level Computer Vision. In *Proceedings of the 8th Scandinavian Conference on Image Analysis*, Tromsø, May 1993. SCIA.
- [2] R. James Firby. *Adaptive Execution in Complex Dynamic Worlds*. PhD thesis, Yale University, USA, 1989. Yale University Technical Report, YALEU/CSD/RR #672.
- [3] G. H. Granlund. Response Generation and Learning Crucial Issues in Machine Vision. In A. Pinz and W. Pözlleitner, editors, *Machine Perception Applications. Proc. of the IAPR TC-8 Workshop in Machine Perception Applications, Technical University, Graz, Austria, 2–3 September, 1996*, volume 93, pages 155–184, Oldenbourg, Wien, Austria, September 1996. IAPR, Österreichische Computer Gesellschaft. Invited paper.
- [4] G. H. Granlund, H. Knutsson, C-J. Westelius, and J. Wiklund. Issues in robot vision. *Image and Vision Computing*, 12(3):131–148, April 1994. Invited paper.
- [5] C-F. Westin. *A Tensor Framework for Multidimensional Signal Processing*. PhD thesis, Linköping University, Sweden, S-581 83 Linköping, Sweden, 1994. Dissertation No 348, ISBN 91-7871-421-4.
- [6] J. Wiklund and H. Knutsson. A Generalized Convolver. In *Proceedings of the 9th Scandinavian Conference on Image Analysis*, Uppsala, Sweden, June 1995. SCIA.