Situation Identification by Unmanned Aerial Vehicle

H.S. Nguyen, A.Skowron, M. Szczuka
Institute of Mathematics, Warsaw University
Banacha 2, 02–097, Warsaw, Poland
emails: {son,skowron,szczuka}@mimuw.edu.pl

Abstract

An approach to a multi-facet task of situation identification by Unmanned Aerial Vehicle (UAV) is presented. The concept of multi-layered identification system based on soft computing approach to reasoning with incomplete, imprecise or vague information is discussed.

1 Introduction

The task of controlling Unmanned Aerial Vehicle (UAV) in the traffic control applications arises a plethora of different problems (refer to [14]). Among the others the task of identifying the current road situation and deciding whether it should be considered as normal or potentially dangerous. The main source of information is the video system mounted on board of the UAV. It provides us with images of the situation underneath. Those images are gathered by digital video cameras working in visual and infrared band. With use of advanced techniques coming from the area of Image Processing and Computer Vision it is possible to identify and describe symbolically the objects existing within such as cars, road borders, cross-roads etc. Once the basic features are extracted from the image the essential process of identifying situation starts. The measurements taken from the processed image are matched against the set of decision rules. The rules that apply are contributing to the taking of final decision. In some cases it is necessary to go back to the image since more information should be drawn. The set of rules that match the image being processed at the time can identify a situation instantly or give us the choice of interpretations of what we see. In the latter case we may apply a higher level reasoning scheme in order to finally reach the decision. The entire cognition scheme that we construct is based on the concept of learning the basic notions and reasoning method from the set of pre-classified data (image sequences) that were given to us prior to putting the system into live.

Learning techniques for pattern extraction from visual input has been used in [3]. In the paper we propose to use multilayered learning of soft concepts as a basic tool in the identification of situation on the road. The basic approaches we use are rough sets [5], rough mereology [7] and granular computing paradigm [15], [16] built over the first two approaches [7].

Different tasks in the process of object identification require different learning techniques. In many applications, it is necessary to create some complex features from simple ones. This observation forces a hierarchical system of identification with multi–layered structure. In case of autonomous systems, some parameters of this layered structure are determined from experiment data in some learning processes called *layered learning*(see [12]).

Soft computing is one of many modern methods resolving some problems related to complex objects (eg. classification, identification and description) in real life systems. We emphasize two major directions: *computing with words* and *granular computing* aiming to build foundations for approximate layered reasoning (see [15], [16]).

In this paper we present an approach for layered learning based on soft computing techniques. We illustrate this idea by the problem of road situation identification. We also describe a method for automatic reasoning and decision making under uncertainty of information about objects (situation, measurement, etc.) using our layered structure.

2 The Problem of Complex Object Identification

In order to realize what are the problems behind the task of complex object classification/identification let us bring a simple example. Lets consider an image sequence showing two cars on the road turn, one taking over the other with high speed (see Figure 1). Now the question is how to mimick our perception of this situation with automatic system. What attributes in the image sequence should be taken into account and what is their range?

We may utilize background knowledge we have gained from human experts as well as general principles such as traffic regulations. However, the experts usually formulate their opinions in vague terms such us: ”IF there is a turn nearby AND the car runs at high speed THEN this situation is rather dangerous”.

The problem of finding out which attributes are being taken into account within our background knowledge is the first step. After that we have to identify the ranges for linguistic variables such as: ”turn nearby”, ”high speed”. The only reasonable way to do that is learning by example. The choice of proper learning scheme for basic attribute semantics is another problem that have to be overcame.

Yet another complication is the traditional trade-off between exactness and effectiveness. In case of real-time systems like UAV we need sometimes to omit some attributes since it takes too much time and effort to measure them. The proper choice of attributes for a particular situation is crucial, and we cannot expect to find a universal one. We should therefore possess a mechanism for dynamic update of feature vector as situation below evolves.

From the very beginning we allow our model to use notions formulated vaguely, imprecisely or fuzzy. That triggers the need for constructing the inference mechanism that is capable of using such a constructs as well as to maintain and propagate the levels of uncertainty.

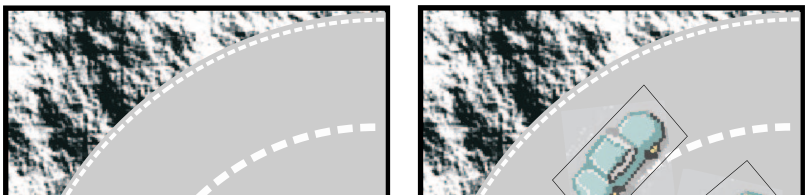

Figure 1: The example of image sequence.

The mechanism of propagating uncertainty should work both top-down and bottom-up. It is important to be able to get a conclusion with a good level of certainty provided we have measurements certain enough. But, it is equally important to have possibility of solving this equality other way - having the requirements for final answer determine allowable uncertainty level in the lower layers of reasoning scheme. And once again learning by example seem to be the best way to do that. Of course, the character of data (spatio-temporal) must be utilized.

3 Construction of Identification System

Construction of Identification Structure includes:

- Learning of basic concepts (on different layers) from sensor measurements and expert knowledge (see [10], [11]).

- Synthesis of interfaces between different learning layers (in particular, for uncertainty coefficients propagation, decomposition of specifications and uncertainty coefficients as in [7]).

The domain knowledge can be presented by levels of concepts. In case of UAV, we propose three-layer knowledge structure.

The first layer is built from the information gained using image processing techniques. We can get information about color blobs in the image, contours, edges and distances between the objects identified. It is possible with advanced computer vision techniques and additional information about placement and movements of UAV (see [8], [4]) to get the readings that are highly resistant to scaling, rotation and unwanted effects caused by movement of both target and the UAV. Information at this stage is exact, but may be incomplete. Some of this information, however expressed in exact units may be by definition imprecise e.g. car speed estimated from the image analysis.

The next layer incorporates terms that are expressed vaguely, inexactly. At this level we use linguistic variables describing granules of information like "Object A moves fast". We also describe interactions between objects identified. Those interactions such as "Object A is too close to Object B" are vaguely expressed as well.

Third layer is devoted to inference of final response of the system. In the basic concept, the inference is based on the set of decision rules extracted from the knowledge we have with regard to the set of training examples. In

Figure 2 we present an example of knowledge structure for "danger overtaking maneuver".

Very crucial for operation of the entire system is the role of interlayer interfaces. They are responsible for unification of "language" between layers.

The original features may be measured with use of different techniques and units. Therefore it is necessary to unify them. At the first glance this task may look simple. What can be difficult in changing pixel to centimeters or degrees? But, we have to realize that in most cases subsequent images are affected by various distortions caused by e.g. changing angle of view as UAV hovers over the scene. Compensation and augmentation of various elements in order to pass a unified set of measurements to the next level requires many times the interface to perform task such as pattern recognition, intelligent (adjustable) filtering and so on.

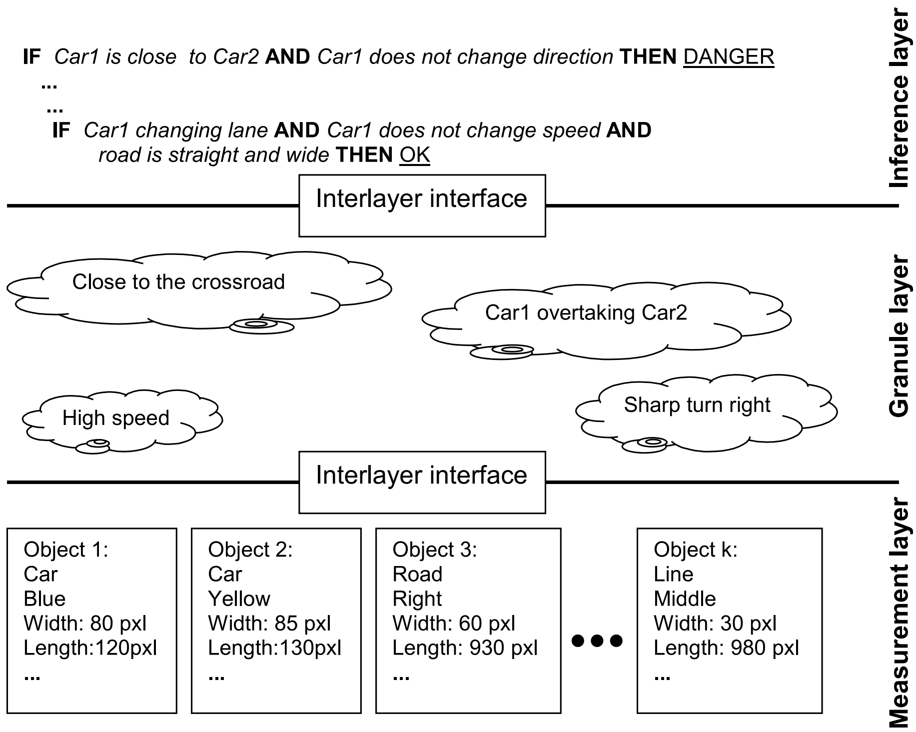

Figure 2: The layered structure.

The interface between granule and inference layers is responsible for adjusting the output from the process of matching measurements against existing granule concepts. In order to be able to apply the decision rules, we have to express the setting of granules in a particular situation using the terms used by rules. This process includes, among others, the elements of spatio-temporal reasoning, since it is the interface that sends the information about changes in mutual placement of granules in time and about changes in placement of object within granules. This is especially crucial if we intend to make our system adaptive. The ability to avoid another error and to disseminate cases that were improperly treated in the past strongly relies on interface capabilities.

The learning tasks are defined in the same way as in Machine Learning. For example, the identification problem for UAV can be formulated as follows: "*given a collection of situations on roads:* $TRAINING\_SET = \{(S_1, d_1), ..., (S_n, d_n)\}$ *labeled by decision values (verified by experts) induce the concepts corresponding to the decision classes*".

The main problem of layered learning is how to design the learning schema consisting of basic learning algorithms that allow to identify properly new situations on the road.

One can see that the natural approach is to decompose the general, complex task into simpler ones. In case of knowledge structure presented in Figure 2 we can propose the following learning schema (Figure 3).

Unfortunately, this simple schema does not work well, because of the occurrence of phenomena characteristic for layered learning, such as:

1. **Constrained learning**: It is necessary to determine the training data for particular learning algorithm in the design step. Usually, training data is presented in form of decision table. For example, the decision table for "Granule Learner 1" (see Figure 3) consists of:

    - conditional attributes: *"car speed", "atmosphere condition", ...*. In general, these attributes defined by properties of objects from Measurement Layer.
    - decision attribute: in the simplest case, this attribute has two values: YES – for positive examples and NO – for negative examples.
    - examples (objects, cases) are taken from situations of TRAINING_SET restricted to the conditional attributes.

    the problem comes from the fact that for every situation we have only the global information whether this situation is dangerous or not and

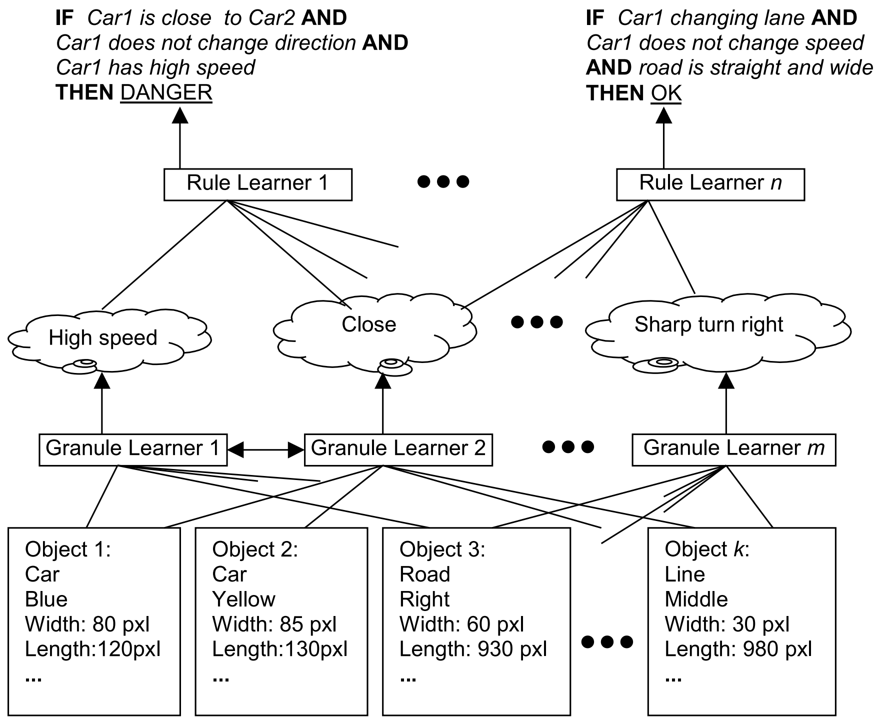

Figure 3: Learning in layered structure.

which rule can be used to identify this fact, but we do not have information about particular basic concepts that contribute to that decision. For these situations, we only have some *constrains* between basic concepts. For example, for some situations we can have a constraint for "Granule Learner 1" and "Granule Learner 2" of the form: "for this situation, one of those two learners must give a negative answer".

2. **Tolerance relation learning**: The learning algorithms (Learners, see Fig. 3) used are not perfect. The possibility of imprecise solution and learning error is their native feature. This raises the question of possible error accumulation in consecutive layers. The problem is to set constrains for error rates of both individual and ensembles of algorithms in order to achieve acceptable quality of the overall layered learning process. We can apply the general approach for this problem called Rough Mereology (proposed in [7]). This idea is based on determining some standard objects (or *standards* for short) for every concept and

corresponding tolerance relation in such a way, that if a new situation is close enough to some basic concepts, this situation will be close enough to higher level concept as well.

Three components must be in place to make tolerance learning work:

- Parameterized set of tolerance relations. The similarities between objects (measurements, situations) can be determined if only we know the right configuration of parameters. Each similarity description consists of two parts - the set of parameters identifying similarity measure (not necessary being a distance in strict sense) and the closeness parameter ε which tells us how to determine neighborhood for a given object.
- The decision algorithm that is "similarity tolerant". This "tolerance" means that for objects which are close with respect to a given similarity measure (lay in the neighborhood) the algorithms' response does not change. More precisely, an algorithm A such that:

  $$\forall_x \; d(x,o) < \varepsilon \Rightarrow A(x) = A(o) = decision(o) \tag{1}$$

  where d is the similarity measure, x is a considered object, o is a pre-classified reference object, $decision(.)$ is the mapping of reference objects onto decision values. Such algorithms should be also parameterized in order to provide the control over their behaviour for different tolerance relations.
- The control algorithm that manages selection and tuning of parameters for both tolerance relations and decision algorithms to be used. The problem is to make such a control in a way that allow proper reaction to the overall performance of entire system. This problem can be solved by application of the learning concept that is (in principle) similar to the idea of backpropagation learning in multilayered, feedforward neural networks. Similar idea scan be found in [6]

4 Construction of Reasoning Engine

Reasoning mechanisms should take into account uncertainties resulting from uncertainties of sensor measurements, missing sensor values, bounds on resources like computation time, robustness with respect to parameter deviation and necessity of adaptation to the changing environment.

In the simplest version the reasoning engine spins around the set of decision rules extracted from background knowledge and observation of learning examples. The output of matching feature measurements against predetermined information granules is fed to that layer thru interlayer interface. The rules that match in a satisfactory degree are contributing to final decision. There are two sub-steps in this process. First is the determination of the degree of matching for a particular rule. This is done with use of techniques known from rough set theory and fuzzy theory. Second is the determination of final decision on the basis of matching and not matching rules. This is almost a classical example of conflict solving. To perform such a task many techniques have been developed within rough set theory itself, as well as in related fields of soft computing.

Since the entire systems operates in very unstable environment it is very likely that the decision is not reached instantly. The inference mechanism requires more information. The request for more information addresses underlying components of the system. To make the decision process faster we should demand the information that not only allows to find matching rules, but also allows elimination of possibly largest set of rules that can be identified as certainly not relevant in a given situation. This is the kind of reasoning scheme resembling medical doctors approach. The physician first tries to eliminate possibilities and then, basing on what remains possible, orders additional examinations that allow to make final diagnosis. The very same mechanism in case of situation identification allows to decrease size of inference system and in consequence, improve effectiveness.

Construction of Reasoning Engine includes:

- negotiation and dialog methods used for (relevant) concept perception on different layer;
- spatio-temporal reasoning schemes based on concept perception on different layers with application of soft computing methods (using e.g., rough or/and fuzzy approaches) to reason about the perception results on different layers.

Negotiation and dialog methods are core component of interlayer interfaces, especially while they operate top-down i.e. process the requests send from the inference layer to these below. If in the inference layer no rule match enough in the current situation, the necessity of drawing additional information arises. The catch is to obtain result with minimum possible effort. We can identify within inference layer the set of information granules that, if better described, will be enough to take decision. But, on the other hand, getting this particular, additional information may be complicated from the granule layer point of view. Implicitly, drawing additional information requires also the negotiation between granule and measurement layers.

Within both negotiation processes (inference – granules and granules – measurements) we have to establish the utility function. Such a utility function, if maximized, allow to find the most reasonable set of granules or measurements for further processing. Unfortunately, we may not expect to find an explicit form of this function. Rather we should try to learn its estimation from the set of examples using adaptive approximation methods such as neural networks.

Yet another element that may require negotiation is the matching of granules against measurement in case we have distributed or multi-source data. That comes from the fact that different sources of information (e.g different sensors) may provide us with data relevant to a single granule. It easy to see that, for example, primitive referred by the granule "A is close to B" may be concluded independently from both visual and infrared scanning. The key is to choose the measurement which in further steps of identification procedure will be more effective.

5 Conclusions and Further Research

We presented a proposition of autonomous system, that can learn how to identify complex objects from examples. Our idea was illustrated by example of dangerous situation identification system which can be integrated for UAV project [14]). We listed some new characteristic aspects for layered learning methods. In next papers we intend to describe a system that learns to make complex decisions. By complex decision we mean, e.g., the family of action plans [1]. Such a system will not only tell us what's going on below (danger/no danger), but also recommend what further action should be performed by UAV. This problem is extremely interesting for autonomous systems [14], [2].

Acknowledgments

This work has been supported by the Wallenberg Foundation - WITAS Project, by the ESPRIT–CRIT 2 project #20288, and by grant 8T11C02519 from the State Committee for Scientific Research (KBN) of the Republic of Poland.

References

[1] Bäckström Ch.: Five Years of Tractable Planning, New Directions in AI Planning, EWSP'95, Assisi, Italy, 1995, IOS Press, Amstaerdam.

[2] Doherty P., Gustafsson J., Karlsson L., and Kvarnström J.: TAL: Temporal Action Logics Language. Specification and Tutorial, Linkping Electronic Articles in Computer and Information Science, Vol. 3 , 1998, nr 015, http://www.ep.liu.se/ea/cis/1998/01.

[3] Fernyhough J., Generation of Qualitative Spatio-Temporal Representations From Visual Input, University of Leeds, 1997.

[4] Pal S.K., Ghosh A., Kundu M.K., Soft Computing for Image Processing, Physica–Verlag, Heidelberg, 2000

[5] Pawlak Z.: Rough Sets. Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Dordrecht, 1991.

[6] Han L., Peters J.F., Ramanna S., Zhai R., Classifying Faults in High Voltage Power Systems: A Rough-Fuzzy Neural Computational Approach, LNAI 1711, Springer Verlag, 1999, pp. 47-54

[7] Polkowski L., Skowron A.: Towards Adaptive Calculus of Granules, In: [16], vol.1, pp. 201–227.

[8] Russ J. C., The Image Processing Handbook, CRC Press LLC & Springer Verlag, Heidelberg, 1999

[9] Skowron A., Stepaniuk J.: Tolerance Approximation Spaces, Fundamenta Informaticae, Vol. 27, 1996, 245–253.

[10] Skowron A., Stepaniuk J.: Towards Discovery of Information Granules, Lecture Notes in Artificial Intelligence 1704, Springer-Verlag, 1999, pp. 542–547.

[11] Skowron A., Stepaniuk J.: Information Granules in Distributed Environment, Lecture Notes in Artificial Intelligence 1711, Springer-Verlag, 1999, pp. 357–365.

[12] Stone P. : Layered Learning in Multiagent Systems: A Winning Approach to Robotic Soccer, MIT Press, 2000.

[13] WWW SPACENET page: `http://agora.scs.leeds.ac.uk/spacenet/`.

[14] The WITAS project homepage: `http://www.ida.liu.se/ext/witas/`.

[15] Zadeh L.A.: Fuzzy Logic = Computing with Words, IEEE Trans. on Fuzzy Systems Vol. 4, 1996, pp. 103–111.

[16] Zadeh L.A., Kacprzyk J. (Eds.): Computing with Words in Information/Intelligent Systems vol.1–2, Physica–Verlag, Heidelberg, 1999.