# INTELLIGENT NAVIGATION FOR AUTONOMOUS ROBOTS USING DYNAMIC VISION

C.Hock; E.D.Dickmanns
Universität der Bundeswehr München
W.Heisenberg Weg 39, D-8014 Neubiberg, FR Germany

## Abstract

With the autonomous road vehicle VaMoRs behavioral competences have been developed over the last decade for visually guided longitudinal and lateral road following including obstacle avoidance; these methods are numerically very efficient and locally adequate. They do not allow global navigation. With the autonomously guided vehicle ATHENE for transportation tasks on the factory floor, indoor landmark navigation has been demonstrated exploiting the same 4D-approach to dynamic machine vision.

Combining the results of both application areas, a very flexible and powerful intelligent navigation scheme is achieved. The background and the basic features of this new method are discussed.

## Key words:

Navigation, Landmarks, Autonomous Robots, Dynamic Vision, Data Fusion.

## 1. Introduction

Route planning and visual guidance of vehicles has been a subject of research in artificial intelligence for a long time. The remote sensing capability of vision allows an agent to orient itself relative to the environment and to other objects up to relatively large distances.

In well developed road networks the capability to perform complex missions, clearly has three essential components: 1. safe movement along the road disregarding navigational aspects in the large (so-called cruise phases), 2. orientation on the mission scale and taking proper navigational decisions when required, and 3. the capability of implementing navigational maneuvers from the previous to the following cruise section.

The first task has been solved and demonstrated with VaMoRs, a 5-ton van with proper sensing and actuation capabilities, extending recursive estimation techniques to image sequence processing with the 4D-approach [Dickmanns, Zapp 86, 87; Dickmanns, Christians 89, 91; Dickmanns, Graefe 88; Dickmanns, Mysliwetz 92]. Lane following, convoy driving, stop-&-go in a traffic jam, and lane changing, all have been demonstrated in the framework of the EUREKA-project PROMETHEUS with the 'Common European Demonstrator-3' VITA of our industrial partner Daimler-Benz. Obstacles may be detected at ranges up to 100 m and proper reactions are triggered through situation assessment and feed-forward or feedback control actuations.

Task two has been tackled in our group first for guiding vehicles on the factory floor [Hock 91]. If a flexible scheme requiring little hardware installations is being looked for, visual landmark navigation is the way to go; the least expensive approach would be that well discernible feature groupings already present in the environment may serve as landmarks and are sufficient for reliable recognition of the actual vehicle position. This is exactly how humans and animals tend to find their way around, even when due to changing lighting conditions and annual seasons the appearance of landmarks changes systematically. The 4D-approach integrating temporal aspects right from the beginning is well suited for realising this scheme efficiently. It requires memory and knowledge processing onboard the system.

The third task mentioned above also has been tackled successfully: On the Autobahn, navigation is simply done by proper lane changing and lane following; up to now, the trigger impulse had to come from the human operator. However, it is relatively simple to achieve full autonomy once the capability of traffic sign recognition can be incorporated. On normal roads, the capability of recognizing crossroads as landmarks and of turning off is the behavioral competence required; this is being worked at [Müller 92].

The term 'autonomous robot' seems to be surprising at first sight, since 'robot' per se is an autonomous device by definition. Most of the industrial robots, however, still have a link between a human operator and the machine. The robot follows a predefined program with no choice of making own decisions. A large amount of research work in the field of robotics is devoted to reducing the need for information exchange between man and machine. A necessary step towards autonomy is to provide intelligence within the onboard devices. Autonomous operation is then determined by the intelligence of the machine.

The dictionary [Hornby 78] explains the word 'intelligence' as the 'power of perceiving, learning, understanding, knowing, mental ability'. Perception and understanding of the operational environment for mobile robots are the main aspects of research work performed at UniBwM over the last decade.

A large fraction of our knowledge about the real world is concerned with the temporal domain; we learn to understand this during early life more or less subconsciously while the capability of crawling, walking and manipulating other objects under earth gravity is being acquired. The temporal sequence of states of moving objects and their transition characteristics constitute very essential knowledge about the real world providing us with the capability of acting adequately even though it does not seem to be represented explicitly. This has long been overlooked in Artificial Intelligence which concentrated its efforts on explicitly represented abstract knowledge about quasi-static relations between objects in the world.

The natural sciences and engineering technology have developed adequate methods for representing these facts about the physical world. They describe them within the framework of differential equations with time as monotonically increasing independent variable. As I.Kant has elaborated in his 'Critiques ...' more than two centuries ago it has to be kept in mind that space and time are not properties of objects. We cannot help carrying it into the world by our sensing and analysis systems; we ourselves exist in these basic four dimensions. Therefore, it was decided to install these basic four dimensions in the 4D-approach to dynamic machine vision right from the beginning in order to be able to deal with the real world efficiently. This was the main contribution of our approach to machine vision; the rest follows almost automatically.

## 2. System components

The availability of two different testbeds each with a navigation system based on the 4D-approach [Dickmanns, Graefe, 88], allows test runs to be performed under various kinds of circumstances. The first vehicle is clearly specified to indoor applications and called 'ATHENE', whereas the second one, dubbed 'VaMoRs', is designed for outdoor usage. Each system has its specific advantages, but the overall design of the navigation system is closely related, so there are no difficulties to transfer well proven solutions between the two. Since 'VaMoRs' is well documented in [Zapp 88; Dickmanns, Graefe 88] more effort is put on describing details of 'ATHENE' in this paper.

Fig.1: Experimental vehicles: a) VaMoRs b) ATHENE

The main components of the indoor experimental setup can be divided into three categories. First, the robot itself, which is a converted AGV equipped with all necessary actuators, interfaces and onboard power supply. Second, a special multiprocessor vision system for realtime image sequence analysis and interpretation is placed on the vehicle, and third, there are two extra computers for the navigational task and the low level control system. ATHENE can be driven autonomously under computer control and serves as a rolling indoor platform for research work on landmark navigation and computer vision.

In the front part of the vehicle an electromechanical pan platform is mounted carrying a standard monochrome CCD camera. Viewing direction control is done either by the navigation system or the vision system itself, depending on the actual task. The camera pointing capability allows active scene search and horizontal tracking, e.g. for initial self orientation or landmark tracking while driving.

For image sequence processing a custom made system BVV 2 [Graefe 90] with four Intel 80286 processors has been utilized in the experiments. This multiprocessor system of the MIMD type consists (in the case of the indoor application) of 4 commercial, standard Multibus I singleboard computers spanning the performance range from 8086 to 80286. Key feature of this multiprocessor vision-system is the physically distributed, thus truly parallel image access capability of all CPUs directly involved in image operations. This overcomes the common I/O bottleneck of general purpose machines, in which usually only one processor has direct image access. Here, no central frame store exists. Any processor linked to the videobus through a custom made videobus-interface (VBI) can simultaneously access and process a subsegment (window) of the digitized 256 by 244x8 bit per pel grayscale image. The VBI basically is a hardware-attachment to a standard singleboard computer containing a window-selection logic and two fast window-buffers storing 4k pel each. Multiple windows can be independently positioned or changed in size, shape and sampling density under software control. It should be noted that except for the VBI no custom hardware and no dedicated image processing devices are being used in this experimental system. The advantages of applying easily programmable standard microprocessors instead, proved to be significant for the system's applicability and efficiency as a research tool. A further key point is the flexible interprocessor communication scheme based on message passing, form-

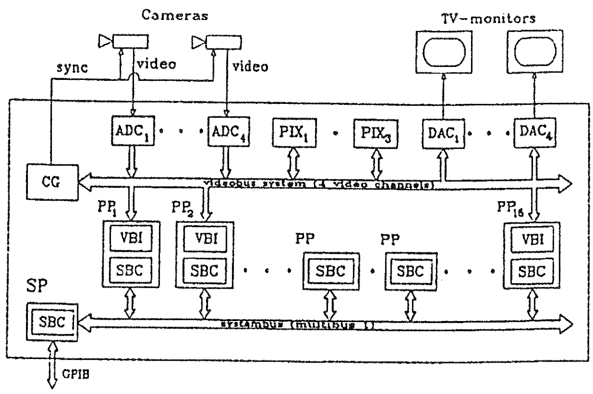



Fig.2: Architecture of BVV 2

ing a loosely coupled system requiring only modest bus bandwith. Using the communication services of the distributed operating system kernel, a desired processing structure can be defined entirely by downloadable application software. Thus, task specific cooperating processor clusters can be formed. Typically, such a CPU group consists of several 'Parallel Image Processors' (PP) at a low hierarchical level that perform local feature extraction operations on their windows. The PPs of a group may be coordinated by a 'General Purpose Processor' (GPP, more recently renamed 4D-object processor 4D-OP) at a higher hierarchical level, which interprets the PPs' feature-data and controls or guides the activities of its PP group (see also fig.2).

Low level control of motors, collection of information from all kinds of sensors and preprocessing of sensor data is performed by a SMP-System (Intel 80186), which holds a number of I/O function boards. The navigation software and the overall management runs on a 80386 PC-AT compatible computer, which also serves for mass storage, real-time data logging and software development. Communication between the three computer systems is done via an IEC bus.

The second testbed serves as 'rolling fieldlab' for computer vision research in outdoor applications with human operators and supervisors on board and is known as 'VaMoRs'. This vehicle has drawn international attention by the demonstration of autonomous road-following at speeds up tp 96 km/h in 1987. This demonstration set a world record for autonomous road vehicles. Beside the physical appearance, the main differences between the two testbeds, looked at from a navigational point of view, are found in the more powerful image processing system and the sophisticated pointing device for the camera. An electromechanical pan-tilt platform carrying two CCD cameras mounted in the center behind the front windshield, hanging from the roof, provides fast 2-axis viewing direction control. Its control is part of the vision-system. Equipped with lenses of different focal length, a scene can be analysed in the wide angle image for global features (such as the road boundaries) and with more detail in the enlarged image (e.g., for focussing on objects or obstacles further away). The camera pointing capability allows active search and tracking, e.g., for initial self orientation, motion blur reduction and continuous road tracking while driving. For obvious reasons it is desirable not to loose the road from the camera's field of view when the vehicle changes its heading or enters a tight curve. Specially for obstacle recognition it is essential to have the camera actively center that part of the scene where potential obstacles are of interest. Instrumental to the success of 'VaMoRs' were two key elements: the 4D-approach, as the core of the guidance system, and the BVV 2 for real-time image sequence processing. In the meantime, VaMoRs has been reequipped with a more powerful transputer network for both image sequence processing and situation assessment as well as vehicle control.

## 3. Perception of the environment

The way of perceiving the environment strongly determines the kind of intelligent behavior exhibited by robots. In this section the potential of optical sensors will be shown.

Sensors generally used for solving the navigational task can be divided into two categories[Cox, Wilfong, 90]: First, there are 'dead reckoning' sensors, which allow the position of the robot to be estimated by integrating sensor information over time. Dead reckoning is usually performed by odometry and inertial guidance sensors. Odometry is the most common form of sensors available on mobile vehicles equipped with wheels. Using dead reckoning, position errors may grow without bounds unless an independent position fix is used to reduce these errors. This is where the second category plays its role. External or environmental sensors are able to provide information on the surrounding environment. Among the many sensors and processing schemes that computer vision has to offer, dynamic vision is the one with the most potential in perceiving the environment.

The various techniques being investigated for object detection and tracking can be roughly categorized into a) edge based using intensity images b) region based using intensity images and c) region based using color images [Kuan et al. 86], [Turk et al. 87], [Wallace et al. 86]. The approach applied here is of the first type as far as the image processing level is concerned. Though this method might be considered the most susceptible to real-world disturbances like shadows or ill-defined, ambiguous edges, it has been shown that in combination with a proper guiding and interpretation mechanism, it is efficient and robust at the same time..

On the feature extraction level, local, oriented edge operators both for detection and tracking are used. Corner finding operators can be realised by searching for adequate constellations of two edge elements. The edge operators are entirely software based, running on a standard microprocessor (8086 or 80286/8 MHz; T222/20MHz transputers more recently). They work directly on raw image (window) data; no prior signal conditioning or smoothing is necessary.

A severe drawback of commonly used edge finding methods (e.g., all 'classical' operators) is that they are purely signal driven and lack scene-descriptive criteria; they treat 'right' and 'wrong' edges, e.g., due to shadows, equally. Poor performance will usually also result under the influence of noise or texture, both inevitable in natural scenes. But even optimized algorithms cannot resolve ambiguities on the low level, even less so, if they work on local support only (as on a window). This shows the need to include more a priori knowledge or to establish some control mechanisms. In our case the guiding mechanism for real-time road boundary and object tracking is based on spatio-temporal scene interpretation utilizing generic 3D geometrical models for the environment and objects, a known ego-motion model and the laws of central (perspective) projection.

Even when considering the relatively simple shape of two converging road boundaries in the image, there are many sources of ambiguity and uncertainty under real world conditions: e.g. there may exist dominant edges across the road due to shadows, there may be multiple nearby parallel edges or intermittent stretches without welldefined boundaries, all additionally blurred due to vehicle motion (fig.3).

Accepting ambiguity on the low level allows the use of simple and fast algorithms there (even more so, if only a fraction of the whole image is processed). Having to resolve ambiguity or uncertainty then on a higher level requires that no essential information is withheld or lost by the low level operations. This, however, will mostly occur if single, optimal results due to local criteria are extracted. So, a well balanced approach is necessary to fine tune the distribution of competence between the signal driven and the model driven processing levels.

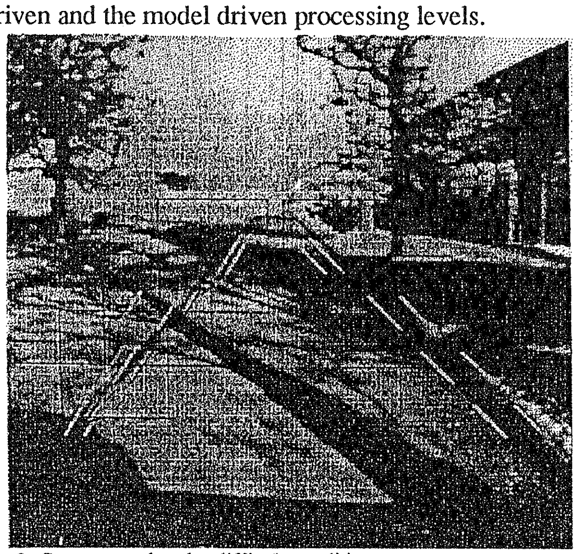


Fig.3: Campus road under difficult conditions

As the proper appearance of the road boundaries in the image can be easily predicted given the observer's relative position and the motion state, in the approach used here local edge extraction is tightly guided and controlled by the interpretation level; i.e. the interpretation level commands the expected edge direction and location plus some optional parameters for adapting the algorithm according to its predictions. In return, it receives a description set of several edge candidates in the area with the orientation sought (fig. 3), plus additional ones from potential edges with similar orientation in a limited sector around the commanded direction. These are checked against the expected edge locations, then the best candidates satisfying the model criteria are selected for updating the state estimates, or they may be rejected at all if falling outside of some allowed threshold around the reference position.

The core algorithm correlates an image area along a search path within the window with an ideal step edge as reference pattern. A very efficient implementation of this technique on a conventional microprocessor has been originally given by [Kuhnert 85]. Very similar directional step edge operators are described in [Canny 86], derived, however, under optimality aspects with respect to shape and operator width; computational simplicity and efficiency has been less emphasized in the latter case.

A version of Kuhnert's algorithm with a significantly improved interface to the interpretation level is being used here. It is better adapted to noisy real-world scenes and applies 'bar masks' with up to 32 discrete orientations, yielding a directional resolution of down to 6 degrees. Up to four different edge element (edgel) candidates are extracted per window, so that for the road boundaries a set of up to 32 edgels per camera may be passed to the interpretation level for selection and further analysis.

On an Intel 80286 microprocessor (8 MHz/no waitstates) it takes less than two video cycles (40 ms) to subsequently analyse two windows (sized 48x48 pixels) at different locations for three different edge orientations and to extract a set of edge candidates for each window. In the transputer system this step is performed on one T222 processor within 8 windows.

## 4. Intelligent navigation using landmarks

With the definition of intelligent behavior of an autonomous system geared to making decisions in response to environmental events it is logical, therefore, that at least crude understanding of the task domain is a basic requirement. In the following section, the evolution from dead reckoning to path following and finally to landmark navigation is presented.

Main sensors for the navigation task performed with 'ATHENE' have been precision shaft encoders on both rear wheels and steering, one rate gyroscope for measuring the turn rate of the robot and one black and white TV-camera including an image sequence processing system. Each of the different sensor types has its specific merits, depending on the robot's state. The signals of the shaft encoders are usefull as long as the robot operates on smooth and well defined surfaces with moderate move-

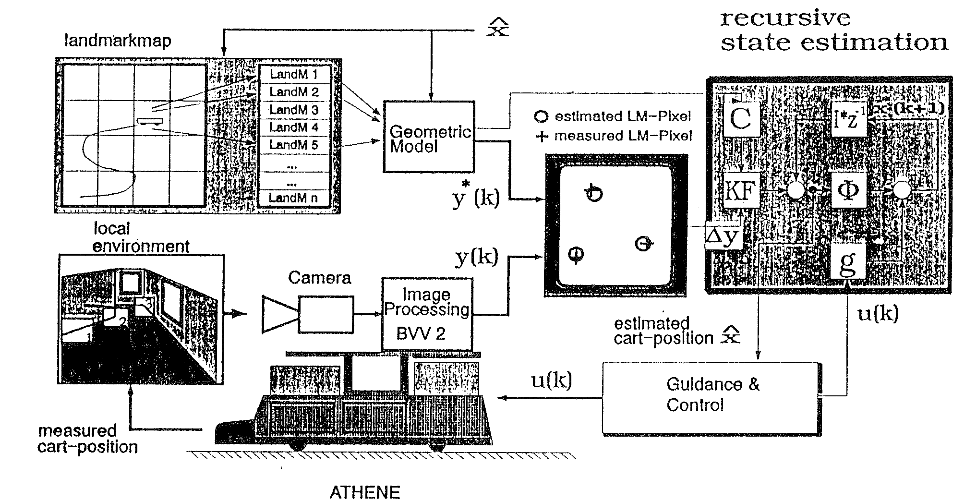



Fig.4: Landmark navigation

ments. Rapid turning of the cart will produce errors partly due to the unpredictable slippage of the wheels. In this case the gyroscope carries the best information about the turning rate, while the camera is too slow to track detected features. The drift of lowcost gyros prevents using the signal for a longer period than about one minute. These measurement errors and perturbations create a discrepancy between the planned and the real location. Therefore depending on the navigational precision required, the position of the robot has to be updated by visual feedback [Hock, 91](see fig .4). For this purpose there are two categories of visual aids to navigation. It will be shown, that the combination of both will yield a powerful and stable method for traveling autonomously from point to point. The first mode is called 'path or lane following' and has been well proven over a long period of time on the testbed 'VaMoRs'. This approach is tailored to well structured environments like hallways or paved roads with or without lane markings. In path following, motion control by visual feedback is limited to one dimension, the lateral deviation from the nominal trajectory. Longitudinal control only affects time but not the spatial trajectory shape.

While driving on a road, the temporal curvature changes in a certain look ahead range in front of the vehicle create a time varying guidance input to the control system. For road image sequence interpretation the assumption is made that any change in slope of the road boundaries in the image originates either from motion of the vehicle relative to the road, from road width changes, or from changes in its horizontal and/or vertical direction. Introducing road curvature as a state variable to be estimated by a Kalman filter was proposed and realized in [Dickmanns, Zapp 86] in combination with a dynamical model of vehicle motion. As high speed roads exhibit linear changes of curvature over runlength, due to ego-motion the relations between the curvature parameters can be formulated as a compact system of difference equations for sampled data systems. Thus, a dynamical model for these road parameters also exists. Besides being essential for high speed lateral and longitudinal vehicle control, together with the road width the curvature parameters yield a very compact spatial shape representation of the road in terms of differential geometry [Dickmanns, Mysliwetz 92]. The location of the road boundaries in the image is mainly determined by the state variables: lateral offset of the vehicle from the lane center, camera heading $\Psi_k$ relative to the road direction and the horizontal and vertical road curvature parameters.

There is a difference between autonomous navigation in hallways and on roads. Hallways are a guidance network with predominantly straight connection lines. The surface to be travelled on can be considered as a flat and smooth plane. Therefore, the lateral deviation from the nominal path can be expressed in terms of lateral distances to adjacent walls. Any other object with known parameters for its geometrical description (environmental model) may serve for this purpose as well. The relative lateral distance to the object $y_v$ and the runlength coordinate then constitute the state variables. Vehicle displacements from the preplanned trajectory may be caused by misalignments and odometric errors.

It is clearly seen that the application of following an indoor corridor is a subset of the more complex road following task. But as soon as the lane markings disappear or a decision has to be made with respect to which object or landmark heading has to be selected, an areal navigation method is required. The vehicle now has to travel across open areas or through extended halls utilizing information derived from nearby landmarks.

The main difference between path and areal navigation is that in the first case only topological information may be needed, whereas for the two-dimensional case the geometrical relations between landmarks must be known.

The solution for advancing from one landmark to the next one will be chosen according to the navigation method seemingly preferred by living beings. In this approach the absolute distance between two landmarks need not be known. Therefore, odometry looses importance in this case. This is similiar to the situation when a person gives advice on how to reach a certain street intersection. Explicit information about the distance to the intersection is not needed if the ability of landmark recognition can be presumed. A strategy of getting close to the trigger event must be known, like following the current street. The missing distance information will be substituted by recognizable patterns, as soon as they get in sight. Since it cannot be guaranteed that reliable optical information will be

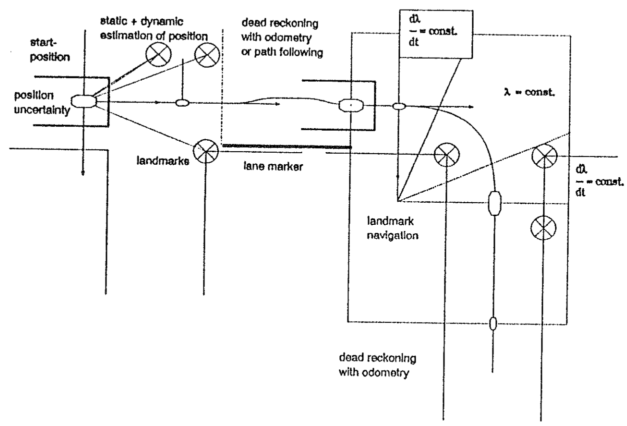


Fig.5: Data fusion of odometric and visual information

available, such as known landmarks or the boundaries of the current path, an intelligent data fusion module is introduced. It is fed by the output of error models of each single sensor and the result of the interpretation process of the scene. In case of ill conditioned optical measurements, the navigation system is driven entirely by dead reckoning algorithms (see fig.5). As soon as new landmarks come in sight and the estimation process has stabilized, smooth transition is performed towards a purely optical navigation method. This switching forth and back is a dynamical process and leads to a stable and robust navigation system.

## 5. Maps

As mentioned in the previous section, at least knowledge about topological relations between landmarks has to be available to the navigation system. This is provided via databases, which will be called 'maps' in the sequel. They also contain additional information about the nearby environment; sometimes it may be advantageous to include specific parameters for maneuver elements necessary for solving the driving task; these specific maps will be briefly called 'task maps' and are handled as separate units. For improved handling in creating and maintaining these databases it has proven advantageous to subdivide the complete set of informations on the environment into specific maps (see fig.6). The concept of these maps will be discussed in the remainder of this section.

A map of landmarks contains a network of one-dimensional trajectories and the characteristics of specific trigger events. The information on the entire workspace is not stored in one single map, but is partitioned into several distinctive ones. Thus, the complete system of landmark maps consists of local maps and their topological connections. Each local map is centered around one landmark or a tight cluster of landmarks and carries the information how to reach the adjacent landmarks. The term 'local' depends on the measurement accuracy available and on the structure of the environment.

The vision system is conditioned for detecting and localizing single features, therefore landmarks are considered as specific groups of features or parts of objects that exist in the natural environment and that have a stable geometrical constellation relative to each other. These landmarks have to be unambiguously detectable by the vision system. Their features are given qualitatively as a priori knowledge to the navigation system. Qualitative aspects and the local 3D coordinates of the landmark position are given in the landmark map and have a reference to both the environmental map and the task map.

The environmental map contains information about the layout of the operational environment. Static obstacles are stored as well as areas that are prohibited to the robot. For indoor application the most convenient way to create such a map is to use digitized blue prints of the building, where the autonomous navigation has to take place.

The task map has a list of the current job orders. One example for the interaction of all three maps will follow: If the job order says 'pick up piece A at spot B and take it to C', the first thing to figure out is where the spots B and C and the current location of the vehicle in the environmental map are. For the sake of simplifying the process, it is assumed that the current position and orientation is known and that the procedure for initializing the system has been done already. Next step is to find the adequate connections within the network of possible trajectories. Afterwards the vehicle has to travel along a known hallway at a predefined lateral distance to the wall, that is safe enough not to bump into objects, which might hang from the wall. This 'wall following' navigation mode is maintained until reaching the landmark 'intersection B'. A short stop which may include a docking manoeuvre provides the opportunity to pick up 'piece A'. The next task is, say, to make a 90 degree turn and follow the path until landmark C is in sight. So the landmarks 'intersection B' and the 'object C' may serve both in global and local positioning. With a couple of specific landmarks the

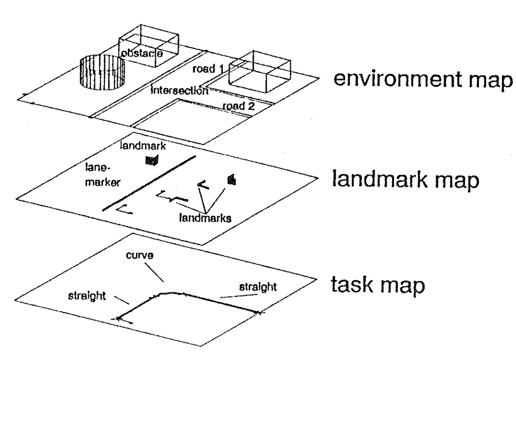


Fig.6: Maps for landmark navigation

navigation system is able to locate unambigously the vehicle on the map.

For recording the landmarks one can think at least of two possible solutions. The first one is to measure manually the 3D-coordinates of the landmarks relative to a local reference and store the values into the landmark map. This procedure implies a lot of work and is prone to errors. The second solution is to guide the vehicle by an operator through the area where afterwards the real mission is to take place. The operator directs the camera towards an appearing landmark and a support software determines how well recognizable by the vision system the landmark picked is. Essentially, the system applies feature extractors and some normalization procedures for recognizing image features relevant to the recognition algorithms (for details see [Kuhnert 90]). Since the human operator has a substantially better understanding of the situation, he has to accept the feature as a landmark before the position can be stored together with other attributes in the landmark map. This map may be considered as the logical link between the task map and the environmental map.

The semi-automatic accumulation of landmarks will produce rich information on the close surrounding and a large amount of data which have to be managed in an efficient way. The process of finding out which landmarks will be well visible and appropriate during a mission may be very time consuming. One solution to this problem may be a computer simulation of the whole setting including a simulated run of the vehicle. Several criteria for optimizing the constellation of landmarks are used to filter out useless objects. Only those landmarks get a stamp, that yield sensitive information about space and, therefore, are usefull for the navigation task. After that, the stamped and well visible landmarks are copied from the original data base; they now get an event stamp from where and how long they have been seen. After that, they are sorted in the sequence as they come in sight during the ride. The relative position of the vehicle on the map is a pointer to those landmarks, which are the best for the navigational task. This simulation of the real mission produces a very compact landmark data base with an event driven search algorithm.

During real-time mission performance no extensive search algorithm is needed, only the event pointer has to be run along with the distance traveled.

All the prerequisites for succesfull landmark navigation have been discussed in former sections. The part, that is still missing, is how to link odometric sensing, perception of the environment and a priori knowledge to an autonomously moving robot. In the next section the well proven 4D-approach will be reviewed. It will be shown that this approch is a unique tool for integrating different sources of knowledge to a reliable representation of a robot's state relative to its environment.

## 6. The integrated spatio-temporal approach

Before going into technical details, some fundamental considerations are made about the internal representation of time and space.

A sensor system in the real world always is at a certain point in space at the one and only 'present' time; this point in time is part of a continuous 'time ray'. A physical object cannot be at two different locations at the same time. In order to move from one location to another, energy is required for ac- and deceleration, and time will go by because the energy available to effect locomotion is bounded. These facts constitute constraints on the motion process which may help considerably when tracking locomotion of objects, especially when ac-/ decelarations are very limited in magnitude as is the case in most of the occurrences in our natural and even technical environment (exceptions being bullets shot by guns for example).

Therefore, if we have a good internal representation of a situation in our environment we are in a much better position to understand the next image of a real-time sequence if an internal representation is available which allows to predict how the process under observation is going to evolve over time taking certain control or perturbation inputs into account. If this prediction model is approximately correct one can concentrate the limited data processing capabilities on the data originating in the local environment of the predicted spot, thereby making the sensing process much more efficient; in addition, also the data processing algorithms may be adjusted to the predicted situation thereby further increasing efficiency. This positive feedback favors the evolution of powerful prediction capabilities since in spite of additional computing resources required for prediction the overall requirements may be decreased for the same performance level; on the other hand, completely new performance levels and new qualities of deeper understanding of environmental processes may be achievable with this approach.

It might be argued, that human culture and its achievements are an outgrowth of nature having discovered this positive feedback during evolution.

In figure 7 qualitative display of internal representation density over the sliding time axis which moves from right to left is given. At the point 'here and now' (shown stationary at the cross-section of the two orthogonal axes) sensors provide data on the actual state of the real world. These data are interpreted taking high-level spatio-temporal world models into account. These dynamical models are derived from those developed for system design and analysis in engineering. In addition, it is taken into account that measurement data usually are superpositions of actual process states (the desired quantities to be recovered) and of measurement noise which is to be deleted. In order to be able to make this distinction, the models representing temporal behavior have to contain both the 'eigen-'characteristics (that means how states change over time when left on their own) and the response characteristics with respect to control- or perturbation inputs.

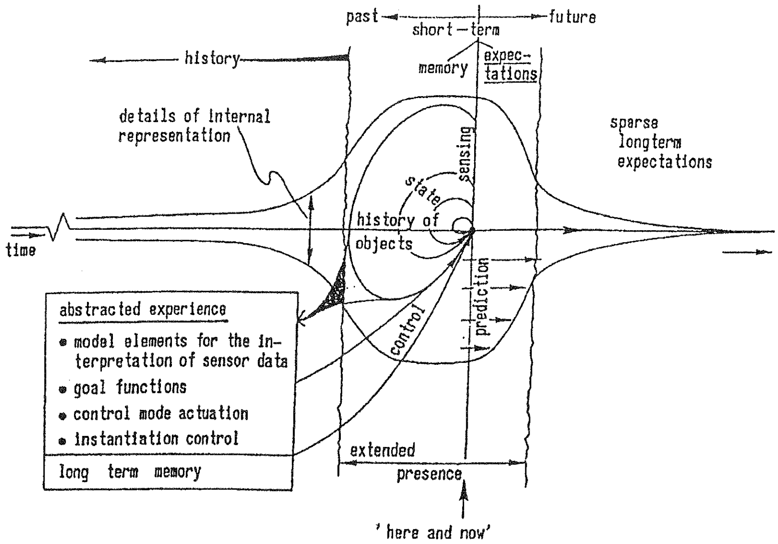



Fig.7: Representation density over time

Once this is represented, predictions of the state evolution over time may be obtained at relatively low cost. Since usually neither the control nor the perturbation inputs of the future are known, prediction usually stops at one cycle (for the normal prediction-error-feedback state estimation process) or after only a few cycles in order not to incur too much uncertainty. For well known feedforward control time history inputs in order to achieve some maneuver element (for example lane change in road vehicle guidance with a sine-like steering angle input over time using proper parameters for period T and amplitude A) reliable predictions over longer temporal ranges (seconds) are possible. Taking standard perturbation statistics into account, even longer ranges over entire maneuver sequences may be meaningful (like prediction of the time needed to go from point A to B). In the average, however, the number of predicted events will vanish on the future time scale to the right.

If good internal models are available for generating rich actual internal representations from the actual data measured, it will be impossible to store all these data as a 'personal history of adventures'; it is not necessary, though. Since the time histories of the state variables may be regenerated from stored initial conditions and control as well as perturbation time history inputs once a proper model for the dynamic behavior is available, only the latter ones need be stored. For these again, instead of pointwise storing each individual time history, parameterized generic models would allow very efficient storage since a dense data input vector may be replaced by a few parameters needed to feed the proper function call. This shows that proper temporal models may be very efficient in reducing memory requirements if things are properly organized. Past process state time histories and events may then be reconstructed actively from combining only a few stored historical data with stored model knowledge. This principle is the basic advantage of the 4D approach combining space and time in an integrated manner.

This type of data compression into valid models is symbolized in figure 7 by the formation of a reduced tail on the past time axis (left). Quasi-static knowledge resulting from this is used lateron for triggering proper control activities depending on the situation encountered. Standard perturbations are counter-acted by feedback control laws which are implemented by a direct loop from the sensory data to the corresponding actuators (see center of fig.7) via internal state variables of recognized objects; this allows stable behavior under perturbed conditions without the explicit knowledge levels having to interact with the high frequency data stream. Only unforseen situations and unpredicted new features discovered lead to an activation of the more knowledge based hypothesis generation part controlling the active set of internal dynamical models (lower left in fig.7).

Seen from this point of view, the entire 'mental' internal world of representations has as its purpose to provide the system with capabilities of data interpretation well suited for control outputs which enable the system to achieve its goals; previous experience may be exploited for this purpose contributing to the rating of a system as being intelligent or not.

## The 4D-approach

For a reliable description of mechanical processes in our everyday environment science has found the framework of threedimensional space and time to be well suited. Objects are defined in this environment as units having special properties or functions. For simplicity, we confine ourselves at present to rigid objects which may be moved as units having constant shape over time (e.g. vehicles, obstacles) or which are static parts of the environment (roads,buildings,installations etc.). Each object has a spatial shape, a position and an angular orientation in a framework relative to the observer, all in 3-D. Objects are classified according to their mobility: 1. Environmental objects are fixed to an environment and determine its visual appearance, like roads, road shoulders, trees and buildings, walls; 2. static objects are presently at rest, however, they may be moved or may even belong to the last class; 3. objects able of autonomous locomotion. The vehicle itself is an object of class 3, for which a model of its locomotion capabilities and of some basic geometrical properties are known. This includes the cause-and-effect relationships with respect to activating the controls and the state transition over time. In addition, the position and orientation of the vision sensor relative to those parts of the body interacting with the environment, i.e. the wheel base, are assumed to be known.

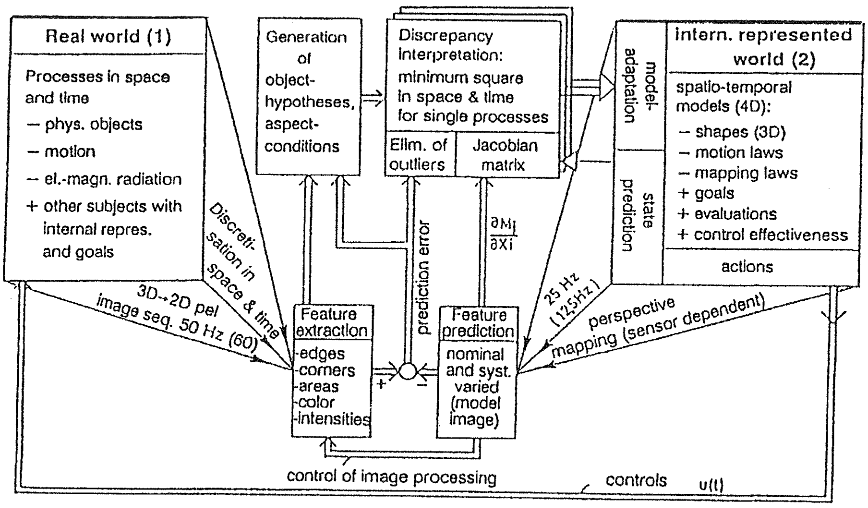



Fig.8: Survey block diagramm of 4D approach

The key tools for integrating space and time in the internal representation are the dynamical models, which are used for capturing the behavior over time of a physical process. As usual in rigid body mechanics, the motion of bodies is separated into center-of-gravity (cg) translation and rotation around the cg. These motion components are described by ordinary differential equations including the effects of control input. For digital control, transition matrices and control effect matrices are derived using well known methods.

Control inputs to the mobile robot carrying the vision system lead to changes in the visual appearance of the world through egomotion. The continous motion of the vehicle and the relative position in the world over time is sensed by conventional black and white video cameras. They record the incoming light intensity from a certain field of view at a fixed sampling rate. By this imaging process the information flow is discretized in several ways.

There is a limited spatial resolution in the image plane and a temporal discretization of 16 2/3 or 20 ms, usually including some averaging over time. This reduces the data flow to a sequence of 2D arrays at fixed time intervals (20 ms). Instead of trying to invert this image sequence for 3-D-scene understanding, a different approach by analysis through synthesis has been selected. From previous human experience, generic models of objects in the 3-D-world are assumed to be known in the interpretation process. This comprises both 3-D shape, recognizable by certain feature aggregations, given the aspect conditions, and motion behavior over time. In an initialisation phase, starting from a collection of features extracted by the low level pel processing (BVV 2, lower center left in fig.8), object hypotheses including the aspect conditions and the motion behavior (transition matrices) in space have to be generated (upper center left). The motion capabilities of the robot, which are constraints characterizing the object, are represented by difference equations, describing the state evolution. With the help of these so-called dynamical models, it is possible to predict the object states to that point in time when the next measurement is going to be taken. By applying forward perspective projection to features measured, using the same mapping conditions as the video camera, a model image can be generated, which should duplicate the measured image if the situation has been interpreted properly. The situation is thus 'imagined' (right and lower center right in fig.8). The big advantage of this approach is that due to the internal 4-D model not only the actual situation at the present time but also the sensitivity matrix of the feature positions with respect to state changes can be determined and exploited over time, the socalled Jacobian matrix. This rich information is then used for adjusting the state estimates recursively in a least squares manner based on the differences between the predicted and the measured feature positions. By this approach, the nonunique inversion of the perspective projection is bypassed based on the continuity conditions captured in the spatio-temporal world model (4-D model). For details see [Dickmanns, Graefe 88] and the references given there. This approach has several very important practical advantages:

- no previous images need be stored and retrieved for computing optical flow or velocity components in the image as an intermediate step;
- the transition from signals (pel data in the image) to symbols (spatio-temporal motion state of objects) is done in a very direct way, well based on higher level knowledge, the 4-D world model integrating spatial and temporal aspects;
- intelligent nonuniform image analysis becomes possible, allowing to concentrate computer resources to areas of interest known to carry meaningful information;
- viewing direction control can be done directly in an object-oriented manner
- the image processing computer architecture can be structured modularly according to the internal representation of spatial objects.

## Dynamical model

As mentioned above, it is intented to recover the actual positions relative to landmarks by measuring their feature position in a temporal image sequence. The prime interest within a known planar surrounding is the position $(x_B, y_B)$ and the angular orientation $(\Psi)$ of the vehicle. Control inputs $(U_\lambda, U_V)$ result either in acceleration in longitudinal direction $(\dot{V}_B)$ or in turning the front wheel (ω). These aspects yield the dynamical model as described in [Hock 90a].

After linearization and with the help of standard

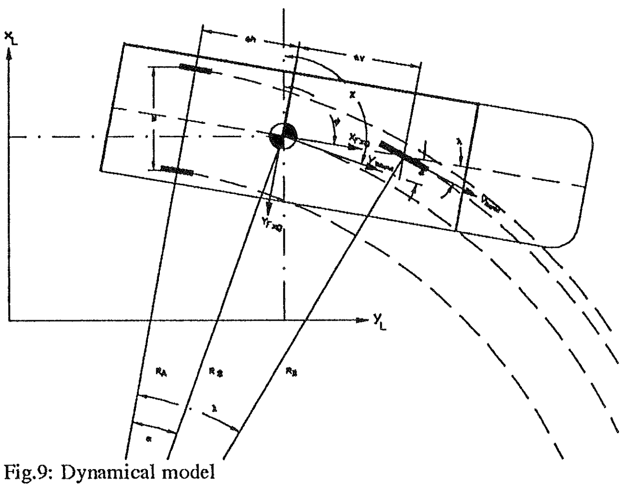


Fig.9: Dynamical model

methods of modern control theory the discrete state transition form is derived (see fig.9).

## Geometric model

The geometric properties of the scene are exploited in combination with the laws of perspective projection in order to describe the position of relevant features in the image plane as a function of relative spatial state. The landmarks are modeled as 3D objects with known coordinates of their centroid and the spatial feature distribution relative to this. The perspective projection equations give

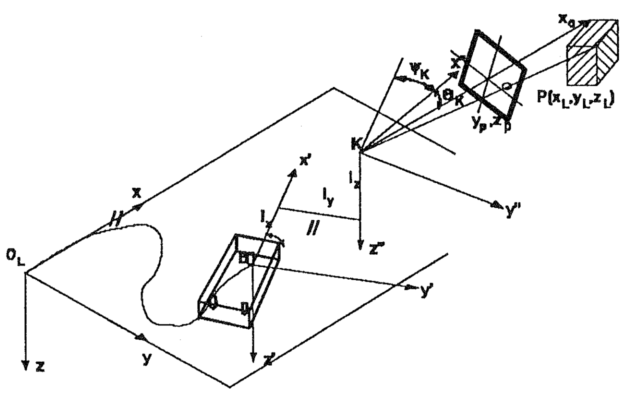


Fig.10: Geometrical model

the horizontal coordinate $y_{pi}$ and the vertical coordinate $z_{pi}$ of the landmark $L_i$ as measured in the image plane (see fig.10).

## Recursive state estimation

The dynamical models link time to spatial motion, in general. 3D shape models exhibit the spatial distribution of visual features which allow to recognize and track objects. In order to exploit both dynamical and shape models at the same time, the prediction error feedback scheme for recursive state estimation developed by Kalman and successors in the 60-ies has been extended to image sequence processing by our group [Wuensche 86; 88]. There are many publications on this approach so that only a short summary will be given here ( see e.g. the survey articles [Dickmanns, Graefe 88; Dickmanns, Myslivetz 92]). The Kalman filter approach introduces knowledge about the dynamical behavior of a process, about the measurement relations and about noise statistics of both process and measurements in order to obtain best estimates of the process states in a least squares error sense recursively as new measurement data arrive. It even allows to substitute this knowledge for missing measurements of state components; these are reconstructed in a way to best fit the overall model. In the 4D-approach to dynamic vision, the Extended Kalman Filter (EKF) for nonlinear systems (see [Maybeck 79]) has been further extended to perspective mapping as the measurement process; the reconstruction capability is thereby exploited for bypassing the strongly nonlinear perspective inversion, utilizing all continuity conditions for spatio-temporally represented objects in 3D space (shape, carrying well visible features) and time (motion constraints, given by the dynamical model, the differential equations of motion).

State estimation, as used here, plays a dual role in the visual interpretation process:

First, it yields a direct transformation from feature locations in image sequences into physical quantities in space (such as $x, y, \Psi$ and their time derivatives), which are related to control actuations.

Second, when using this approach also the control inputs (u) of the vehicle can explicitly be taken into account. Via known dynamics (state transition matrix $\Phi$) the system's state $x^*$ at the next sampling time can be predicted, thus also the expected appearance of landmarks can be computed as a vector $y^*$. This information is used directly to guide the feature extraction process where to look for edges or lines of tracked landmarks.

Only those features matching best the predicted location will be selected and used to actually drive the interpretation process. The selection step is augmented by the information contained in the estimation error covariance matrix $P$. Mapping the predicted uncertainty of the state estimates into measurement space yields the innovation variance, which defines the allowed neighborhood of the predicted values $y^*$ in which the new incoming measurements should lie. Based on this information and by processing only single measurements sequentially, outliers can be rejected. This selection capability reduces measurement noise and is crucial for the robustness of the approach under real-world conditions.

It should be noted, that the measurement equations have to be evaluated only in the forward direction, from state space into the image plane. The non-unique inverse perance, the UD-factorized version of the square-root-filter is used [Bierman 75]. Details may be found in [Wuensche 88; Mysliwetz 90; Bierman 77; Maybeck 79]. By exploiting the sparseness of the transition matrix in the dynamical models a speedup can be achieved. Special care has to be taken in the initialization phase when good object hypotheses are in demand. From feature aggregations which may have been collected in a systematic search covering extended regions of the image, the existence of objects has to be hypothesized.

The general form of a dynamical model for a system of order n (number of state components necessary to uniquely specify the system state $x(t)$ with r control inputs $u(t)$) is, in vector notation

$$\dot{x} = f(x, u, t) + v(t), \tag{1}$$

where $v(t)$ is process noise with covariance matrix $Q$.

The measured variables $y$, an m-vector, are related to the state vector $x$ through the nonlinear relation (including perspective mapping with parameters $p$)

$$y = h(x, p) + w, \tag{2}$$

where $w(t)$ is a measurement noise term with covariance matrix $R$.

Let $\Phi(x_0, u_0, t_i)$ be the linearized transition matrix from $x_0, u_0$ at $t_i$ to $x^*$ at $t_{i+1}$ for the deterministic part of (1) and $C$ the Jacobian matrix of the deterministic part of (2)

$$C(x^*) = \frac{\delta h(x)}{\delta x}\bigg|_{x = x^*}, \tag{3}$$

$P$ is the covariance matrix of the state variables $(n \times n)$ and $KF$ is the Kalman filter gain matrix $(n \times m)$, then the EKF procedure may be summarized as follows.

1. During initialization (t = 0) a hypothesis of the complete state vector and the covariance matrices is given.

$$Q, R, P^*(0), x^*(0) \tag{4}$$

$$KF(0) = P^*(0) \cdot C^T \cdot \{C \cdot P^*(0) \cdot C^T + R\}^{-1}$$

2. The position of each feature in the actual image can be predicted by forward application of the laws of perspective projection exploiting a model of the camera used for measurement.

$$y^*(t_i) = h(x^*(t_i), p) \tag{5}$$

3. The difference between the actually measured value and the predicted value multiplied by the Kalman filter gain matrix $KF$ is used to update the state prediction vector $x^*$ to form the new estimate vector $\hat{x}$. In case of no measurement data the second additive term is 0.

$$\hat{x}(t_i) = x^*(t_i) + KF(t_i) \cdot \{y(t_i) - y^*(t_i)\} \tag{6}$$

4. The control input will be determined.

$$u(t_i) = -k^T \cdot \hat{x}(t_i) \tag{7}$$

5. A state prediction $x^*(t_{i+1})$ for the next measurement can be made, where $\Phi$ is the state transition matrix over one sampling period and $G$ is the control effectiveness matrix for the components of the control vector $u$.

$$x^*(t_{i+1}) = \Phi \cdot \hat{x}(t_i) + G \cdot u(t_i) \tag{8}$$

6. The covariance matrix of the actual prediction error is calculated.

$$\tilde{P}(t_i) = P^*(t_i) - KF(t_i) \cdot C \cdot P^*(t_i) \tag{9}$$

7. Estimation of error covariances $P^*$

$$P^*(t_{i+1}) = \Phi \cdot \tilde{P}(t_i) \cdot \Phi^T + Q \tag{10}$$

8. With the error covariances $P^*$ and the Jacobian matrix $C$ the Kalman filter gain matrix $KF$ is computed.

$$KF(t_{i+1}) = P^*(t_{i+1}) \cdot C^T \cdot \{C \cdot P^*(t_{i+1}) \cdot C^T + R\}^{-1} \tag{11}$$

9. Set i+1 to i, wait for the new measurement and go to step 2.

$$i = i + 1 \tag{12}$$

The current best estimates are the roofed ones ( ^ ) which originate from the expected ones ( * )(eqs. (7,8)) by adding the measurement innovations with the gain matrix $KF$. This matrix is influenced by the covariance matrices $Q$ and $R$ which give room for filter tuning in order to adjust convergence behavior (see [Maybeck 79, Wuensche 88, Mysliwetz 90] for details).

With the information, that is on the one hand gained from perception of the environment and on the other hand brought into the system by a priori knowledge, it is made possible with the help of the 4D-approach to determine the robot's state. The next substantial step towards a successful autonomous navigation is a sophisticated guidance and control system. In the following section our solution to this problem will be discussed.

## 7. Vehicle guidance and control

In vehicle guidance and control, the top down component of mission realization according to some plan developed on the basis of a more or less accurate world model has to meet with the bottom up component of actual vehicle performance and environment encountered. Without knowledge about the road network, reaching the desired goal by controlling the vehicle by chance is unlikely; however, without watching the sensory input and comparing it to some apriori knowledge about safe trajectory steering in a local environment, reaching the desired goal is equally unlikely. The latter case is even dangerous, while the former one may be an enjoyable ride when local control is adequate, but with respect to the intended mission it will be in vain.

Human drivers may be very good at the control level even in completely unknown environments exploiting general knowledge about roads and driveways, traffic rules and traffic participants. With respect to guidance, this is considered a minor problem assuming some basic navigational skills and some local support by knowledgeable people or correct maps. A similar approach to the overall problem of performing a mission has been taken for the autonomous computer-guided vehicles. Safe behavioral competences in driving have been developed first; the necessary capabilities for visual landmark recognition and mission performance are added now.

### Intelligent motion control

The 4D vision process yields the full spatio-temporal state of objects including the spatial velocity components between objects, if properly set up [Dickmanns, Christians 89]. For example, in the road vehicle navigation problem both the road curvature parameters in the look-ahead range and the state of the own vehicle relative to the road may be estimated. With this knowledge a state feedback control law can be applied in order to obtain a lane following competence of the autonomous vehicle[Dickmanns, Zapp 86,87; Zapp 88].

In order to make the different options (and maybe developmental steps) in the evolution of intelligent visual road vehicle guidance more clearly visible, several stages of control realizations will be discussed.

### Output feedback

The simplest case is a single control lateral guidance by proportional output-feedback to the steering wheel. The measured output variable is for example the position of the dark to bright transition indicating the road boundary in one or several lines of a TV-image of a camera looking approximately tangential to the road. If the feedback coefficients are properly chosen (and probably adjusted to the vehicle speed) already good lateral guidance in simple situations can be achieved [Zimdahl et al. 86].

Figure 11a gives a block-diagram description of this bottom line visual control mode.There is no internal representation of spatio-temporal objects; control is actuated in such a manner as to keep the measured value close to a predetermined desired one: in our case the position of the measured image feature close to a position fixed by the designer.

### Implicit notion of state

In the next step towards intelligence an implicitly available model of the process under control allows much more flexible control computation. The measured data are checked against predicted values derived from internal spatio-temporal models.This allows

1. the elimination of outliers and
2. intelligent data smoothing exploiting known noise statistics.

At the core of these recursive estimation methods are dynamical models of the process under control capturing their typical behavior over time. The essential internal variables are directly geared to the physical process in the real world, i.e. its state. The measured output variables are thus transformed by the estimation process into state variables of objects. From this notion, exploiting general knowledge from systems dynamics, (optimal) feedback laws can easily be derived. Fig.11b shows the corresponding block diagram.

The control computation in this mode is still rather fast, although much more involved than in case 12a. Because of the internal representation of the full physical state of the vehicle relative to the road, longitudinal and lateral control can be handled easily; even the cross-influence from road curvature on acceptable longitudinal speed is readily taken care of [Dickmanns, Zapp 86,87]. With today's microprocessors update rates of 25 Hz are easily obtained using modest parallelisation.

This is a conventional control application not requiring any special intelligence(except for the recognition of the object road). The performance achieved by this reflex-like mode of operation is surprisingly high. Since the underlying model captures all the essential aspects of the real world process rather well, a large variety of lane following situations can be handled with just one (possibly adaptive) feedback law. In this way, a behavioral competence is realized through this special data feedback structure.

In refined versions of this scheme, the differentiation of the internal representation into a situation involving several independently represented objects is of advantage for more transparency and for obtaining an easier to handle interface to the human user. For example, in [Dickmanns 88] the road in the look-ahead range and the own vehicle have been completely separated yielding a nice modular structure; however, all of this is completely implicit to the program. Up to this point it is just a convenience for the user of the program. In the next step, this is going to be exploited for improving behavioral competences.

### Implicit notion of situations

Once the notion of spatio-temporal objects and their state is available, classes of relative states among objects can be recognized requiring similar behavioral actions.For example, if the road is free of obstacles, the lane following mode with automatic speed adjustment to curvature may be run. If, however, an obstacle is encountered, either the vehicle has to stop in front of it or it may pass the obstacle if there is enough free space to one side. Figure 11c shows the program structure for realizing this more flexible type of behavior: In parallel to the state estimation it is tested,whether there is a candidate for an obstacle on the road. If this is true, a sequence of actions may be triggered: The longitudinal control mode running is interrupted (the lateral one remaining unchanged for the time being) and the vehicle is put into a deceleration mode either by another feedback law (e.g. preset decelaration rate) or by a prestored feedforward control or by a superposition of both.

In a more refined version of this scheme, the understanding of the situation may be differentiated to a more detailed level. If the road is wide enough and if the size and the position of the obstacle relative to the road can be estimated, the autonomous vehicle may be able to decide by itself whether the obstacle can be passed without leaving the road and touching the obstacle. This check and the corresponding special control activation can be performed in several different ways:

a) The simplest one is to have some heuristic test procedure included in the code which works directly on image data; if certain patterns are matched the system could trigger some special control mode (parameterized feedforward) which guides the vehicle past the obstacle. This partially intelligent reaction is not very satisfactory in general; it may, however, be sufficient for certain applications.

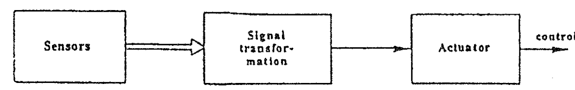
b) A more refined procedural approach determines the best estimate of the relative state of all objects involved. This combined state (the obstacle situation) is then analysed using a preprogrammed classification scheme; as a result, feedforward or feedback or mixed control modes may be triggered for passing the obstacle. Special viewing direction control schemes may be invoked for careful feedback guidance of the vehicle past the obstacle.

c) The last scheme will be treated separately in the next section since it involves explicit knowledge representation.

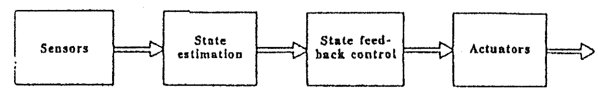
The first two behavioral schemes can be subsumed under the blockdiagram of figure 11d. Depending on the number of behavioral rules implemented, relatively complex behaviors may be realized by this approach without resorting to explicit knowledge representation. It seems that in biological systems (animals) a similar scheme is widely used. Very well adapted motion behavior can be observed in rather nonintelligent species.

Our autonomous vehicle 'VaMoRs', has demonstrated all its achievements using this rule based, switched direct feedback control strategy [Dickmanns, Christians 89; Zapp 88]. Convoy driving on a freeway and 'stop-and-go' in heavy traffic is the latest achievement using this scheme [Dickmanns, Mysliwetz 90]. Switching between feedback schemes with proper smooth transitions is the key to well adapted motion behavior. There is no direct interdependence between the number of objects n, the number of available feedback control laws $m$ and the number of feedforward control programs $r$.

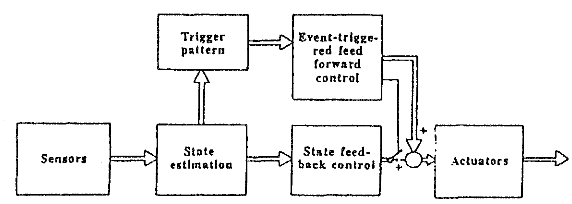
The approach developed is, from a functional point of view, similar to Brook's subsumption architecture [Brooks 87]; however, all the subsumptions are realised in software based on a full spatio-temporal internal representation of relevant objects.This makes the system more flexible, allows easy changes of concepts and an evolu-



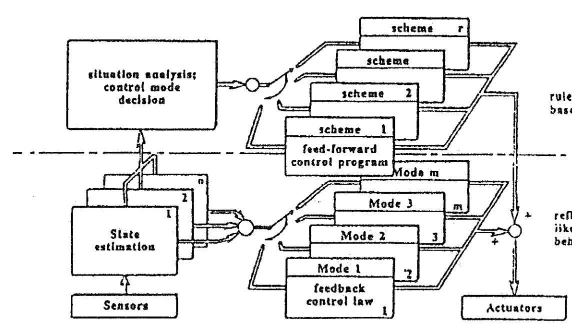

a) Output signal transformation directly into control actuation



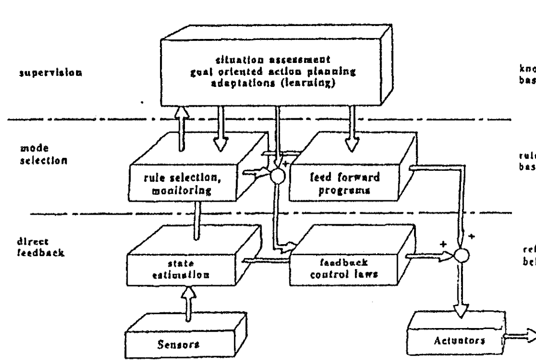

b) Model based spatio-temporal state estimation (for one object) and reflex-like state feedback control (implicit notion of object state)




c) Event detection and triggering of a prerestored parameterized control sequence for more flexible reactions (implicit notion of situation)




d) Selectable fast, reflex like feedback control determination with triggered feed forward components; situation dependent control mode




e) Hierarchiecal scheme for adaptable fast control determination

Fig.11: Steps in the evolution of intelligent control

tionary path to higher developed decision and control levels including knowledge based reasoning.

Explicit higher order world models

If the number of behavioral modes for different situations involving many different objects becomes larger and larger it may become advantageous to structure the behavioral competences according to application areas and classes of situations. The notion of goals to be achieved becomes of importance during this process. This developmental step may be the point where intelligence proper comes into play for autonomous systems since it is here that reasoning enters the field.

There is no more a direct link between a given situation and the control mode selection for the lower level in fig.11d. In the knowledge base there is now a set of goals for the system determining the decision depending on the situation. Some cost function to each goal yields a decision criterion. Which control mode or which parameter set is going to be applied depends on the actual minimal value of several cost functions evaluated before decision taking; those yielding the least cost usually will be the ones selected. In order to avoid frequent switching in ambivalent situations, thresholding or temporal constraints may be introduced more or less heuristically.

In fig.11e, parallel objects, modes and schemes of fig.11d are shown for simplicity byrectangular boxes. They encapsulate the basic cognition and behavior capabilitiesof the system on which the highest knowledge based level can build and which it exploits for realising its plans.

By structuring the system in this way there is no need for steady, especially fast reactions on the highest level since well trained feedforward or feedback control applications on the lower levels are supposed to take care of the continuous fast reaction components. The highest level just has to do the monitoring and triggering.

Mission performance using landmarks

For the low speed AGV ATHENE the three-level scheme of figure 11e has been slightly modified and may be shown as a cascaded triple feedback loop like displayed in figure 12. In the outermost navigation loop the approximate direction of the movement is calculated from different sources of a priori knowledge, but mainly utilizing the job order (task map) and the environmental map information (see fig.4 and 5). The job order tells the vehicle to travel from a certain spot to a different location, meanwhile performing some given tasks. The environment map (e.g. of the building) provides the heading direction, that is the most convenient course towards the desired destination. With the help of the implemented simulation of the whole setting it is possible to determine trajectories free of collisions with known obstacles. Furthermore, only those features for navigation will be marked in the landmark map, which will be visible during the real mission. Depending on the operational mode, the reference trajectory parameters are obtained either relative to an object or as a predefined sub-task. Because of the positional uncertainty the vehicle may have at the starting location, the parameters for distance and direction will be corrected, as soon as the real mission starts and the first landmarks are in sight. All the long term planning is executed in a so called mission planner module, which delivers a sequential list of single mission orders.

At the next level, a pilot-module will take these orders and produce appropriate parameters for the path controller. Vehicle path tracking is done by calculating a desired heading angle, based on the mission order and the positional error. The pilot is responsible for navigation in the local environment and performs its task together with the state estimation module within a cycle time of 100 ms.

The control of the steering angle and the velocity of the cart is performed by monitoring the signals of the gyroscope and the odometry. These specific control laws are implemented on a seperate control computer; therefore, the cycletime is less than 20 ms.

## 8. Experimental results

The approach described above has matured during half a decade of experimentation with two experimental vehicles at the university:

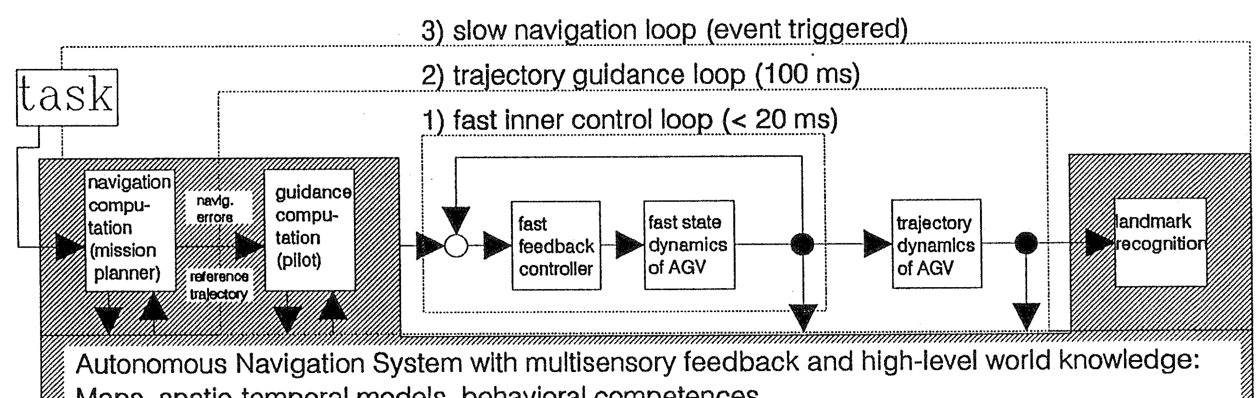



Fig.12: Realization of visual landmark navigation

1. VaMoRs, the experimental vehicle of UniBwM for autonomous mobility and machine vision, a 5-ton van. Always inexpensive PC-type computers have been used for the higher levels: initially, one PC based on the Intel 80286 microprocessor in addition to the BVV 2 with 8086 single board computers sufficed for guiding VaMoRs at its maximum speed of 96 km/h on an empty Autobahn in 1987 exploiting the 4D approach. Only through the powerful and intelligent interpretation constraints introduced by the integrated spatio-temporal models has it been possible to achieve these results with that low computing power on board. Since 1987 Intel 80386 single board computers have been installed on an intermediate hierarchical level in the BVV 2 [Mysliwetz, Dickmanns 87] resulting in much more robust road recognition under strongly perturbed environmental conditions through shadows of trees.

In 1991 all application software developed up to that point in different computer languages was translated into C and ported onto transputers. In a transition phase, both BVV 3 and transputers are used jointly; with the next generation of transputer processors the BVV will disappear.

Since 1984 active viewing direction control has been applied in the framework of our vision systems [Mysliwetz 84]. In 1986 it has been implemented for better recognition of curved roads [Mysliwetz, Dickmanns 86]. The microprocessor for viewing direction control is since integrated in the BVV 2. Especially with the introduction of a bifocal camera pair for better resolution further away this automatic viewing direction control became essential.

2. The vision guided testbed ATHENE was built up in the year 1990 and is equipped with an almost identical sensor system as 'VaMoRs' except for the second TV-camera, but emphasis has been put on autonomous landmark navigation. The operational environment has been provided with landmarks in the form of well discernable, static objects. Either the global position or the location of each target relative to the prescribed local trajectory has been known. The task of the real-time image processing system was to recognize the object and to deliver the corresponding measurement data to the navigation software. The event driven data fusion filter and a Kalman filter are used to combine different qualities of sensor data and to gain the best estimate of the robot's state.

In case of ill conditioned optical information, the vehicle guidance system is able to travel a reasonable distance between target sightings. This is a kind of 'instrument flight', realized with the memorized knowledge about the environment and the egomotion of the vehicle.

The allowable distance travelled between optical updates is a function of how much drift from the nominal path is still safe for not colliding with an obstacle and for finding the next known landmark.

Implementation for the AGV ATHENE started with the dead reckoning navigation approach. Reproduceable experiments showed, that it is possible on a smooth surface to travel over a distance of 20 meters with a lateral error of less than 1 cm. Another test course, shaped like an oval, showed that after a 16 meter ride and a 360 degree turn the heading error was less than 1 degree.

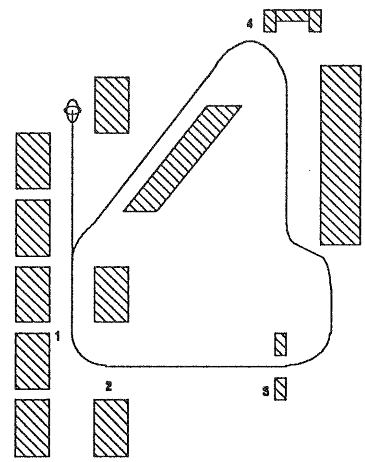


Fig.13: Demonstration experiment for landmark navigation

These results have been obtained after putting some effort into the servo control mechanism. Stable and effective control laws have been derived to allow accurate and safe operation of the vehicle. The control system is split up into different levels in order to have short reaction times for the vehicle to follow the trajectory commanded. But on rough ground, dead reckoning by itself does not yield any acceptable performance.

After implementation of the landmark navigation mode, ATHENE moved autonomously around the laboratory area. The course consisted of four hallways with a total length of about 100 meters and a width of 1.80 meter. Four 90 degree turns connect the hallways. The speed during an autonomous drive has been between 0.2 m/sec in narrow corners and 0.5 m/sec in straight hallways. Final experiments in late 1991 in a factory environment demonstrated the high precision navigation capability with visual feedback from landmarks. The task to be performed by the robot was the following (see fig.13): Starting position was at a roughly known location. The diameters of the error ellipses were between 10 and 25cm. After initialization with an artificial landmark (1) a straight line of workbenches on the right hand side had to be followed until reaching landmark (2); it consisted of a left turn corner. Next landmark (3) was an extremly narrow doorway ( 4 cm free space at each side of the vehicle). A predifined path in a dead reckoning manner leads to the fourth landmark (4), which consisted of a closed door. A left turn brought the vehicle back to landmark (1), where it stopped. Then, a backward docking maneuver was performed to the starting position. The error ellipse now was less than 5 cm in diameter. The same course has been performed a second time after simulating a loading procedure.

## 9. Conclusions

The overall system architecture for flexible automation of vehicle guidance based on dynamic vision has been validated on two different testbeds. The basic sensory inputs to the system are video signals from CCD-TV cameras in combination with odometric and inertial sensor data from the vehicle body. The 4D-approach to dynamic vision yields a natural way to integrate different sources of sensor data. The visual navigation methods, path following and landmark navigation, utilize the expectation-based approach perfectly by exploiting prediction error feedback with full spatio-temporal models for servo-maintaining an internal representation close to the real world objects. Both methods may be used in a complementary or an exclusive mode. In experimental results it has been shown that the combination of both yields very robust and precise navigation capabilities for an autonomous vehicle through structured environments.

## 10. References


[Bierman 75]: 'Measurement Updating Using the U-D Factorization.' Proc. IEEE Control and Decision Conf., Houston, TX, pp 337-346, 1975.

[Bierman 77]:Factorization Methods for Discrete Sequential Estimation, Academic Press, New York, 1977.

[Brooks 87]: 'A Hardware Retargetable Distributed Layered Architecture for Mobile Robot Control.' Proc. IEEE Robotics and Automation, Raleigh NC, April 1987, pp 106-110

[Canny 86]: 'A computational Approach to edge detection.' IEEE Trans. Pattern Recog. Machine Intell., Vol. PAMI-8, No. 6, Nov.1986, pp 679-698.

[Cox, Wilfong 90]: 'Autonomus robot vehicles', Springer-Verlag, New York Berlin Heidelberg, 1990.

[Dickmanns 88]: 'Dynamic computer vision for mobile robot control.' In: Jarvis, R.A.(ed.): Proceedings of the International Symosium and Exposition on Robots, Sydney, Nov. 1988, pp 314-327

[Dickmanns, Christians 89]: 'Relative 3D-state estimation for autonomous visual guidance of road vehicles.' In: Kanada, T. e.a. (ed.): 'Intelligent Autonomous Systems 2', Amsterdam, Dec. 1989, Vol.2 pp. 683-693; also appeared in: Robotics and Autonomous Systems 7 (1991), Elsevier Science Publ., pp 113-123

[Dickmanns, Graefe 88]:a) Dynamic Monocular Machine Vision, Machine Vision and Application 1, International Journal pp. 223-240, Springer International, 1988.
b) Applications of Dynamic Monocular Machine Vision, Machine Vision and Application 1, International Journal pp. 241-261, Springer International, 1988.

[Dickmanns, Mysliwetz 92]: 'Recursive 3D Road and Relative Ego-State Recognition.' IEEE-Trans. PAMI, Special Issue on 'Interpretation of 3D Scenes', 1992 .

[Dickmanns, Zapp 86]: 'A Curvature- based Scheme for Improving Road Vehicle Guidance by Computer Vision.' In: 'Mobile Robots', SPIE Proc. Vol. 727, Cambridge, Mass., Oct. 1986, pp 161-168

[Dickmanns, Zapp 87]: 'Autonomous High Speed Road Vehicle Guidance by Computer Vision.' 10th IFAC Wordl Congress, Munich, July 1987, Preprint Vol 4, pp 232-237

[Graefe 90]:The BVV-Family of Robot Vision Systems, IEEE Workshop on Intelligent Motion Control, Istambul, Türkei, August 1990.

[Hock 90a]:Dynamisches Modell für Dreiradfahrzeuge zur Landmarkennavigation, Universität der Bundeswehr München, UniBwM/LRT/WE13/IB90-4, März 1990

[Hock 90b]:ATHENE, ein Projekt zur Landmarkennavigation, 6. Fachgespräch Autonome Mobile Systeme, Karlsruhe, November 1990

[Hock 91]:Landmark Navigation with ATHENE, 5. International Conference on Advanced Robotics, Pisa, Italy, June 1991

[Kalman 60]:'A new Approach to Linear Filtering and Prediction Problems'. Trans ASME, Series D, Journal of Basic Engeneering, 1960, pp 35-40

[Kuan, Phipps, Hsueh 86]: 'A real time road following vision system for autonomous vehicles'. Proc. SPIE Mobile Robots Conf., Vol. 727, Cambridge, MA, Oct. 1986, pp 152-160.

[Kuhnert 85]: 'Komponenten fuer die modellgestuetzte Interpretation dynamischer Szenen' (in German). Final Report of BMFT-Project 08IT 15113, Aerospace Dept., Universitaet der Bundeswehr Muenchen, 1985.

[Kuhnert 90] : Dynamic Vision Guides the Autonomous Vehicle ATHENE, Japan-USA Symposium on Flexible Automation. Kyoto, Japan, July 1990.

[Maybeck 79]:Stochastic Models, Estimation and Control, Vol. 1 Academic Press, 1979

[Müller 92]: 'Feedforward Control for Curve Steering for an Autonomous Road Vehicle.' IEEE Conf. on Robotics and Autmation, Nice, 1992.

[Mysliwetz, Dickmanns 87]:'Distributed Scene Analysis for Autonomous Road Vehicle Guidance'. Proc. SPIE Conf. on Mobile Robots, Vol. 852, Cambridge, USA, 1987, pp 72-79

[Mysliwetz 90]:'Parallelrechner-basierte Bildfolgeninterpretation zur autonomen Fahrzeugsteuerung'. Dissertation, Universität der Bundeswehr Muenchen, LRT, 1990.

[Turk, Morgenthaler, Gremban, Marra 87]:'Video road-following for the autonomous land vehicle'. Proc. IEEE Int. Conf. Robotics and Automation, Raleigh, NC, Apr. 1987, pp 273-280.

Wallace et al. 86]:'Progress in Robot Road-Following'. Proc. Int. Conf. Robotics and Automation, San Francisco, CA, Apr. 1986, pp 1615-1621.

[Wuensche 86]: 'Detection and Control of Mobile Robot Motion by Real-Time Computer Vision'. In Marquino N. (ed), Advances in Intelligent Robotics Systems. Proc. of the SPIE, Vol. 727, Cambridge. Mass., 1986, pp 100-109.

[Wünsche 88]:Bewegungssteuerung durch Rechnersehen. Fachberichte Messen, Steuern Regeln Bd. 20, Springer-Verlag, Berlin, 1988

[Zapp 88]:Automatische Straßenfahrzeugführung durch Rechnersehen, Dissertation an der Fakultät für Luft- und Raumfahrttechnik der Universität der Bundeswehr München, 1988.

[Zimdahl, Rackow, Wilm 86]: 'OPTOPILOT - ein Forschungansatz zur Spurerkennung und Spurführung bei Straßenfahrzeugen. VDI Berichte Nr.162, 1986, pp 49-60