Computer Vision and Mobile Robotics Workshop (CVMR'98), Santorini, Greece, Sep. 1998.

Reactive Robot Navigation Based on a Combination of Central and Peripheral Vision†

Antonis A. Argyros
Institute of Computer Science
FORTH
P.O Box 1385,
711 10, Heraklion, Crete
Greece

and

Fredrik Bergholm
Computer Vision and Active
Perception Laboratory
NADA/KTH
Stockholm
Sweden

Abstract

In this paper, we present a new method for vision-based, reactive robot navigation that enables a robot to move in the middle of the free space by exploiting both central and peripheral vision. The system employs a forward-looking camera for central vision and two side-looking cameras for sensing the periphery of the robot's visual field. The developed method combines the information acquired by this trinocular vision system and produces low-level motor commands that keep the robot in the middle of the free space. The approach follows the purposive vision paradigm in the sense that vision is not studied in isolation but in the context of the behaviors that the system is engaged as well as the environment and the motor capabilities of the robot. It is demonstrated that by taking into account these issues, vision processing can be drastically simplified, still giving rise to quite rich behaviors. The advantages of the method is that it does not make strict assumptions about the environment, it requires very low level information to be extracted from the images, it produces a robust robot behavior and it is computationally very efficient. Results obtained by both simulations and from a prototype on-line implementation demonstrate the effectiveness of the method.

1. Introduction

The term navigation refers to the capability of a system to move autonomously in its environment, by using its own sensors. This definition is general enough to contain the navigation capabilities of a plethora of biological and mechanical systems. The more specific term *visual navigation* is used for the process of motion control based on the analysis of data gathered by visual sensors. The field of visual navigation is of particular importance mainly because of the rich perceptual input provided by vision. Moreover, navigation that is based on other types of sensors, in contrast to vision, often requires modification of the environment (e.g. insertion of emitters) which imposes constraints on the exploitation of such methods in unknown environments.

The problem of visual navigation has been traditionally treated without taking very much into account the environment of the robot, its body and the characteristics of the behavior that the robot is about to exhibit. Typically, monocular or stereoscopic visual systems are assumed and the effort is then focused in constructing a general representation of the environment that may thereafter serve the solution of any vision-related problem. During the last decade, a new vision paradigm has attracted the interest of the computational vision research community. In this paradigm, called *active and purposive vision* [1], vision is more readily understood in the context of the visual behaviors in which the system is engaged. Consequently, it tries to explore those aspects of the world that are important to the system at a given point in time, instead of aiming at a general representation of the environment which, besides being extremely difficult to extract, it is probably also not needed. The interest in purposive vision is largely motivated by the fact that all biological vision systems are highly active and purposive. *"Vision and intelligence cannot be disembodied: they have no meaning if they have no goals to achieve and a body to interact with the environment"* (from [2]). The purposiveness of visual processes enables the formulation and the solution of simpler problems that have a relative small number of possible solutions and can be treated in a qualitative manner [3]. The goal of algorithms that answer the above questions is to solve many, specific problems under general (loose) assumptions rather than trying to solve a general problem, which can only be done under very restrictive assumptions.

---

† This research has been carried out during the first author's appointment to CVAP/NADA/KTH and was funded under the VIRGO research network (EC Contract No ERBFMRX-CT96-0049) of the TMR Programme.

In this paper, we describe a new method for visual robot navigation based on the principles of purposive vision. By employing a forward-looking camera for central vision

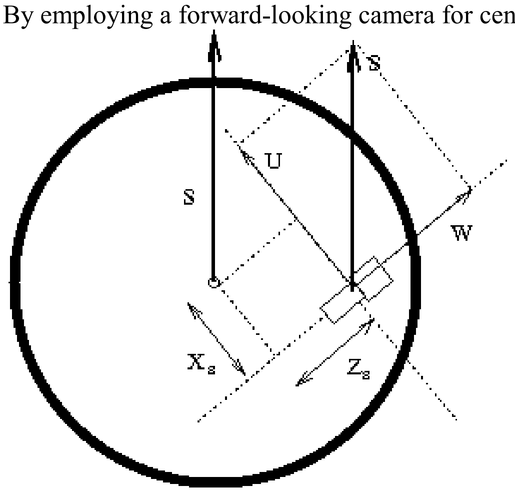

Figure 1: Top-down view of the robot geometry. The robot can translate in the forward direction with velocity S and rotate around an axis that is vertical to the figure plane and passes through the center of the robot's body.

Figure 2: Example of geometric distortions produced by a wide field of view camera

and two side-looking cameras for sensing the periphery of the visual field, reactive robot navigation has been achieved. The developed method combines the information acquired by this trinocular vision system and produces low-level motor commands that keep the robot in the middle of the free space. The aim of this research is not what vision can offer towards building a general-purpose world representation, but how the visual system of a robot can be designed to assist robots with specific bodies and motor capabilities in exhibiting particular behaviors. It is demonstrated that by considering the behavior and the motor capabilities of the robot when designing the visual system of the robot, leads to theoretically simpler and computationally more efficient solutions.

The rest of the paper is organized as follows. Section 2 describes the requirements that the behavior poses to the design of the robots visual system. Section 3 presents issues related to the motion information that can be computed by each camera as well as how this information is processed and used to drive the robot. Section 4 presents results from simulations of the method as well as implementation issues and results obtained by an on-line implementation on the robotic platform. The paper is finalized in section 5 where the main conclusions of this work are summarized and future research plans are described.

2. The behavior, the environment and the body

The study of the behaviors that should be exhibited by an observer, its environment as well as the study of the observer himself provides valuable hints on how the sensors should be placed for a behavior to be implemented. In this work we assume a robot that can translate in the forward direction and rotate (pan) around its vertical axis (Fig. 1). We aim at developing a vision based reactive navigation capability that enables a robot to navigate in indoor environments (long corridors, narrow passages), avoiding collisions with walls and obstacles. The term reactive is used to express lack of a particular destination that could be set by using maps of the environment, landmark recognition etc. The goal of the robot is to wander in free space, keeping away from walls and obstacles. Free space is defined based on the motor capabilities of the robot: since the robot moves on a plane, all 3D structures that do not belong to this plane can be potentially harmful if the robot crashes on them and are therefore considered as obstacles. Since the robot is about to "live" in indoor environments, it is expected to be able to handle situations where long corridors and narrow passages are encountered. It can be shown that difficulties arise when only central vision is used (i.e. a camera or a fixating stereo configuration at the direction of translation). Consider for example a camera with a typical field of view of 30 degrees. Then in a 2 meters wide corridor the robot can only see obstacles that are approximately 4 meters ahead and it is therefore quite difficult to maneuver accurately. On the other hand, the use of cameras with wide field of view [4] give rise to depth dependent geometric distortions that are difficult to correct (see Fig. 2 for an example).

In order to implement this behavior, it appears quite natural to exploit the information provided by peripheral vision, i.e. visual information at large angles with respect to the direction of forward translational motion (see for example the configuration in Fig. 3). By using such a camera configuration, the robot is able to perceive walls and

Figure 3: The KTH head with two extra cameras mounted on it for implementing peripheral vision.

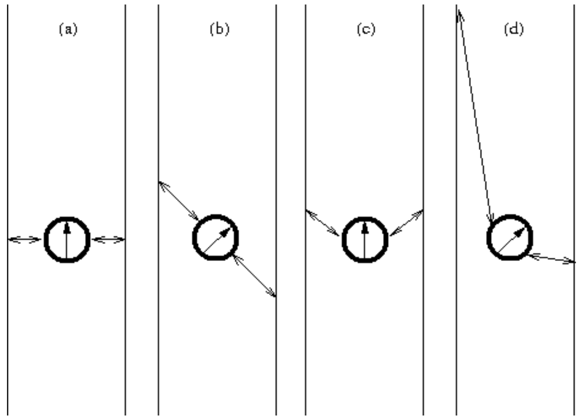

Figure 4: (a), (b) If a robot with lateral peripheral cameras is in the middle of the free space, it perceives equal distances from left and right obstacles independent of pose. (c) and (d) If a robot with slanted peripheral cameras is in the middle of the free space, it perceives equal distances from left and right obstacles only if its pose is correct (i.e. parallel to the obstacles).

obstacles that are immediately close to it. Moreover, the target behavior may be implemented by indirectly *comparing* structure information acquired by the left and right cameras instead of computing *precise* structure information. This approach is motivated by experiments [5] that study the behavior of honeybees. In these experiments, bees were trained to navigate along corridors towards a source of food. The bees were observed to navigate in the middle of the corridor. The eyes of the bees are pointing laterally (at about 180 degrees). The behavior is based [5] on velocity information computed at the left and right eyes of the bee. In simple terms, if the bee is in the center of the corridor, it perceives the world as “leaving" its optical field with the same velocity in both eyes, while if the bee is closer to one of the sides of the corridor, it perceives it as moving faster. Santos-Victor et al. [6] proposed the divergent stereo approach in order to exploit this finding in robots. They exploit visual information that is captured by two cameras with optical axes of opposite orientation that are mounted perpendicularly to the direction of forward translation.

Our research has been carried out in the context of a general framework for vision-based robot navigation [7] and differs to the approach in [6] in several ways. First, peripheral cameras are not be placed in opposite directions because decisions on forward motion should not be influenced by “past" structure information. Moreover, it turns out that control is facilitated when the cameras are slanted. See for example Figs. 4(a) and 4(b) where the cameras are placed laterally on the robot body. The robot perceives equal distances at its left and right side, independently of pose. However, the situation is different in cases 4(c) and 4(d) where the cameras are slanted towards the direction of translation. If the robot is in the middle of the free space, it perceives equal distances from the walls only if its pose is parallel to the walls. Therefore, in this case, balancing distances fixes also the pose of the robot. Second, we study the effects of the observer’s rotational motion in the flow computed by the two peripheral cameras. We also show how central vision (i.e. visual information acquired in the direction of the translation) can be used along with peripheral vision in order to simplify the problems to be solved.

3. Method description

We set a reference coordinate system on the robot’s body so that the Y axis coincides with the robot’s rotational axis and the Z axis is parallel to the robot’s translational motion. As it has already been discussed, we assume that the robot is capable of translating with velocity S and rotate (pan) with velocity β. The nodal point of the right peripheral camera is in coordinates (X_S, Y_S, Z_S) with respect to the reference coordinate system. Consider now a scene coordinate system adjusted to the nodal point of the right peripheral camera. It turns out that the equations relating the 2D velocity (u, v) of an image point p(x, y) to the 3D translational velocity (U, V, W) and rotational velocity (α, β, γ) of the projected 3D point P(X, Y, Z) are [8]:

$$u = \frac{-Uf + xW}{Z} + a\frac{xy}{f} - \frac{xY_S}{Z} - \beta\left(\frac{x^2}{f} + f\right) - \frac{xX_S}{Z} - \frac{Z_S f}{Z} + \gamma\left(y - \frac{Y_S f}{Z}\right)$$

$$v = \frac{-Vf + yW}{Z} + a\left(\frac{y^2}{f} + f\right) - \frac{yY_S}{Z} - \frac{Z_S f}{Z} - \beta\frac{xy}{f} - \frac{yX_S}{Z} - \gamma\left(x - \frac{X_S f}{Z}\right) \quad (1)$$

where f is the focal length of the camera. The translational velocity S of the robot produces a translational velocity (U, 0, W) at the camera coordinate system, while the rotational velocity of the robot produces a rotational velocity β at the camera coordinate system. Note also that $Y_S = 0$ for the specific camera configuration. Therefore, Eqs. (1) become

$$u = \frac{-Uf + xW + X_S\beta}{Z} - \beta\left(\frac{x^2}{f} + f\right),\ v = \frac{yW + X_S\beta}{Z} - \beta\frac{xy}{f} \tag{2}$$

The projection of the optical flow (u, v) along the intensity gradient direction (i.e. the perpendicular to the edge at that point) is also known as normal flow. The normal flow is less informative than optical flow but can be computed robustly and efficiently from image sequences by just using differentiation techniques. Moreover, in contrast to the computation of optical flow, no environmental assumptions such as smoothness are required for normal flow computation. For the above reasons, the proposed method for reactive robot navigation relies on the computation of the normal flow field. Let (n_x, n_y) be the unit vector in the gradient direction. The magnitude u^M of the normal flow vector is given by:

$$u^M = n_x u + n_y v \tag{3}$$

By substituting Eq. (2) in Eq. (3) we obtain for each of the left (L) right (R) and central (C) cameras:

$$\begin{aligned}
u_L^M &= \left(\frac{-U + \beta Z_S}{Z_L} - \beta\right) n_x f + \left(\frac{W + \beta X_S}{Z_L} - \beta x\right)\frac{(xn_x + yn_y)}{f} \\
u_R^M &= \left(\frac{U + \beta Z_S}{Z_R} - \beta\right) n_x f + \left(\frac{W - \beta X_S}{Z_R} - \beta x\right)\frac{(xn_x + yn_y)}{f} \\
u_C^M &= \left(\frac{xn_x + yn_y}{Z_C}\right) S - \left(\frac{xn_x + yn_y}{f} + fn_x\right)\beta
\end{aligned} \tag{4}$$

where Z_L, Z_R and Z_C represent the depth of the 3D points perceived by the three cameras, respectively. By selecting normal flow vectors for which it holds that $xn_x + yn_y = 0$ [1], we obtain:

$$f_{LD} = \frac{-U + \beta Z_S}{Z_L} - \beta,\ f_{RD} = \frac{U + \beta Z_S}{Z_R} - \beta \tag{5}$$

$$f_{CD} = -\beta \tag{6}$$

and by selecting normal flow vectors for which it holds that $(n_x, n_y)=(0, 1)$ (the vertical normal flows), we obtain:

$$f_{Lh} = \frac{W + \beta X_S}{Z_L} - \beta x,\ f_{RH} = \frac{W - \beta X_S}{Z_R} - \beta x \tag{7}$$

Equations (5) and (7) employ functions of depth that can be computed by the peripheral cameras. They can get a simpler form by noting that the function acquired by the central camera (Eq. (6)), gives the rotation β. Thus, Eqs. (5) and (7) can be rewritten as:

$$f_{LD}^d = \frac{-U + \beta Z_S}{Z_L},\ f_{RD}^d = \frac{U + \beta Z_S}{Z_R}, \tag{8}$$

$$f_{LH}^d = \frac{W + \beta x_S}{Z_L}, f_{RH}^d = \frac{W - \beta X_S}{Z_R} \tag{9}$$

Thus, central vision can be used to derotate the flow fields produced at the peripheral cameras. Having exploited this capability, it turns out that:

---

[1] The selected normal flow vectors are those that are tangent to circles centered at the image center.

$$\left(f_{RD}^{d}+f_{LD}^{d}\right)\tan(\phi)+\left(f_{RH}^{d}-f_{LH}^{d}\right)=\left(U+\frac{X_S}{Z_S}+W\right)\left(\frac{1}{Z_L}-\frac{1}{Z_R}\right), \tag{10}$$

where ϕ is the angle by which the peripheral cameras are slanted towards the direction of translation. Equation (10) can be rewritten in a simpler form as follows:

$$A=C\left(\frac{1}{Z_L}-\frac{1}{Z_R}\right) \tag{11}$$

In Eq. (11), A is a quantity that can be directly computed from functions of normal flow that have been extracted from the central and peripheral cameras.. C is an unknown constant that depends on the characteristics of the body of the observer as well as its constant translational velocity. Function A is equal to zero when the left and right cameras are in equal distances from world points and takes positive or negative values depending on whether the right camera is farther or closer from obstacles compared to the left camera. Therefore, the computable quantity A can be used to control the rotational velocity by keeping the quantity A as close to zero as possible, achieving this way the desired behavior.

4. Implementation issues - Experimental results

The experimental evaluation of the proposed method has been based on both simulation results as well as on results obtained by an on-line implementation of the method on a real robotic platform.

Simulations have been based on the KHEPERA simulator [9], which has been modified to simulate the central and peripheral cameras of the robot. The aim of the simulation experiments was to test the control law used to drive the robot. Thus the function A of Eq. (11) has been simulated and the simulated robot was set to navigate in various environments. Several experiments were conducted. Figure (5) shows a sample run. The trace of the robot during the simulation is also presented. It can be seen that the robot moves in a smooth path among the various obstacles of the environment.

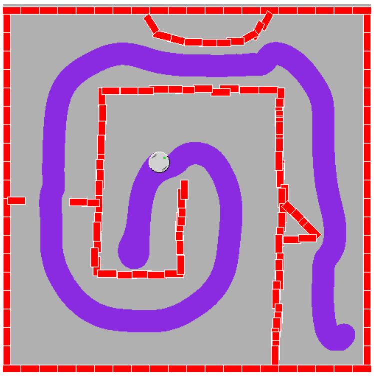

Figure 5: A run of the simulated robot. The robot performs a smooth path in the corridors of the simulated world.

One of the most interesting results of simulations was the difference in the behavior of the simulated robot depending on whether the peripheral cameras were laterally placed or slanted on the robot's body. It turns out that slanted cameras outperform the laterally placed cameras.

The simulation experiments are, of course, not capable of testing the performance of the method when real vision processes are employed. For this reason, an on-line implementation of the method has been realized. The platform used was a LABMATE ROBUTER on which "Charlie", the KTH vision head has been mounted. Two extra cameras were mounted on "Charlie" implementing the peripheral vision, while one of the two "Charlie" cameras was implementing central vision (Fig. 3). A SUN Ultra Sparc was responsible for peripheral vision processing and a PENTIUM processor running LINUX was responsible for central vision processing. The distributed processing as well as the interprocess communication was based on the TCX communications library [10]. Various navigation scenarios

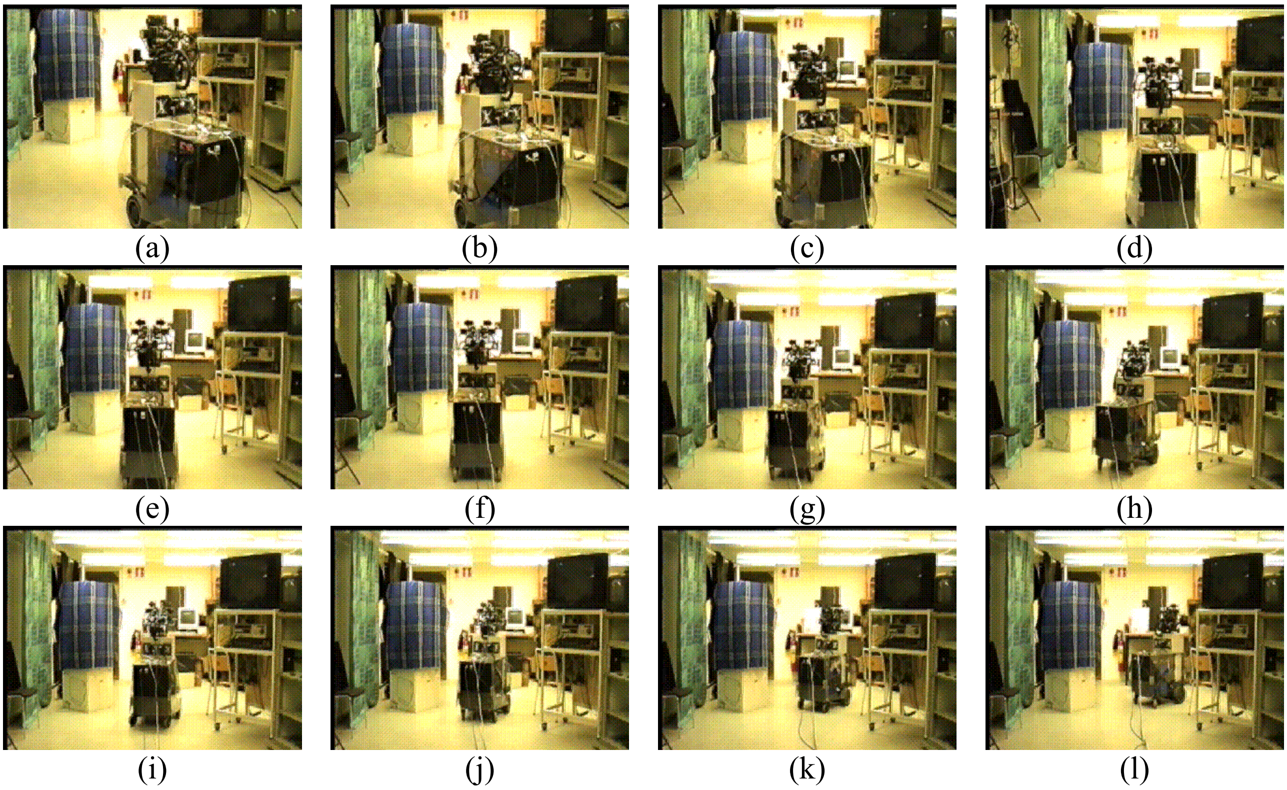

Figure 6: Snapshots of a navigation session.

have been tested in which the robot successfully managed to perform maneuvers in narrow passages. Due to space limitations, we present snapshots from one such experiment in Fig. 6.

5. Conclusions

A method has been proposed that enables a robot to navigate in free space based on a combination of central and peripheral vision. The method does not make strict assumptions about the environment, it requires very low level information to be extracted from the images, it produces a robust robot behavior and it is computationally very efficient. Results obtained by both simulations and from a prototype on-line implementation demonstrate the effectiveness of the method. Peripheral vision seems to be very useful for achieving certain behaviors and its combination with central vision seems natural and appears to be powerful. Future research work will investigate ideas on further exploiting combinations of central and peripheral vision.

References

[1] Y. Aloimonos, I. Weiss and A. Bandopadhay, "Active Vision'', *IJCV*, 1(4): 333-356,1988.
[2] R.A. Brooks, "Intelligence Without Reason'', *AILAB Memo 1293*, MIT AI Lab, April, 1991.
[3] Y. Aloimonos, "Purposive and Qualitative Active Vision'', *Proceedings of DARPA Image Understanding Workshop*, 1990, 816-828.
[4] T. Camus, D. Coombs, M. Herman and TH Hong ,"Real-Time Single-Workstation Obstacle Avoidance Using Only Wide-Field of View Divergence", In Proceedings of ICPR'96.
[5] MV Srinivasan, M Lehrer, WH Kirchner and SW Zhang, "Range Perception through Apparent Image Speed in Freely Flying Honeybees", *Visual Neuroscience*, 6:519-535, 1991.
[6] J Santos-Victor, G. Sandini, F. Curotto and S. Garibaldi, "Divergent Stereo in Autonomous Navigation: From Bees to Robots", IJCV 14 (2): 159-177, March 1995.
[7] J. Dias, F. Bergholm, and P. Fornland, "VIRGO Vision-based Robot Navigation Research Network, Overall Considerations and a Framework for Vision-based Navigation'', Tech. Rep. ISRN KTH/NA/P--96/27--SE, Sept. 1996.
[8] H.C. Longuet-Higgins and K. Prazdny, "The Interpretation of a Moving Retinal Image'', *Proceedings of the Royal Society, London B*, 385-397, 1980.
[9] Olivier Michel, Khepera Simulator 2.0, *User Manual, University of Nice, Sophia Antipolis.*
[10] C. Fedor, "TCX: An Interprocess Communication System for Building Robotic Architectures'', Users Manual, 1994.