A neuroinspired cognitive behavioral control architecture for visually driven mobile robotics *

Cornelia Beck1∗∗∗, Umberto Olcese2∗, Alberto Montagner2, Stefan Ringbauer1, Heiko Neumann1, Antonio Frisoli2, Rita Almeida3, Massimo Bergamasco2, and Gustavo Deco3

1Institute for Neural Information Processing
Ulm University
Oberer Exelsberg
D-89069 Ulm, Germany
heiko.neumann@uni-ulm.de

2PERCRO Laboratory
Scuola Superiore Sant’Anna
piazza Martiri della Libertà 33
56127 Pisa, Italy
a.frisoli@sssup.it

3Department of Technology
Universitat Pompeu Fabra
Passeig de Circumval·lació, 8
E-08003 Barcelona, Spain
ritap.almeida@upf.edu

Abstract—Several studies have shown that the optic flow serves as a tool for navigation for animals. Flying insects use it to follow paths and avoid obstacles, while in primates it represents an additional input that can improve navigational performance. A neuroinspired architecture for optic flow calculation and decision making, based on the cortical organization of the human brain, has been developed and successfully used as a novel control system for a mobile robot. Navigation in a corridor and obstacle avoidance are achieved relying only on the optic flow. Moreover, the presence of the optic flow improves the performance of the robot in target reaching and obstacle avoidance tasks in a virtual environment, affecting its trajectories as shown by experiments with human subjects.

Index Terms—Mobile robot, optic flow, behavior-based robotics, neuroinspired architecture, autonomous navigation.

I. INTRODUCTION

Optic flow represents a very important information for aiding the navigation of both vertebrates and invertebrates [1], [2]. Different kinds of information can be extracted from optic flow, e.g., speed, travelled distance, time to contact, heading, and distance relative to obstacles appearing in the periphery of the visual field. Several attempts have been made to apply an optic flow based navigation strategy to mobile robots, using either engineering and computer science techniques [3]–[6] or trying to replicate the mechanisms that have been identified in neural systems of insects [7], [8]. First prototypes that mimic biological behavior of insects are now also tested in the automible industry, e.g., to automatically prevent accidents by using bee-based avoidance of collisions (see http://www.nissan-global.com/EN/NEWS/2008/STORY/080926-01-e.html).

Humans and many animals can estimate the optic flow in their surround in detail and very quickly. Small animals like insects can easily navigate in their natural environment, avoiding static and moving obstacles that are in their way. Also humans can handle very difficult tasks. Consider for example a scenario on the highway where the driver is moving himself, but can still rate the speed of other cars to navigate safely even within heavy traffic and at high speed. The objective of this work is to develop a system for a mobile robot (see Fig. 1) that follows the neural mechanisms found in biological systems. In particular, a control architecture (Fig. 1) is proposed based on a module emulating V1/V2 and MT/MST cortical areas for the computation of the optic flow and a decision module, corresponding to the functionality of the prefrontal cortex (PFC). It consists of 3 competing populations of neurons that simulate the decision for the navigational strategy of bees. Both the optic flow estimation and the control mechanisms are neuroinspired. Using this architecture we want to explain and mimic the navigation behavior observed for animals and humans.

We show that this architecture is effective for performing successful navigation in a way which is comparable to other optic flow based systems. The developed algorithm is evaluated using a standard control technique and a neural controller. The results are compared with each other and with the trajectories achieved by manually driving the robot navigation. Furthermore, preliminary experiments of the same architecture running in a virtual environment are presented to show that the presence of an optic flow component modulates the navigation behavior as demonstrated in psychophysical experiments involving human subjects.

II. COMPUTATION OF VISUAL INFORMATION

The visual input that represents the basis for the robot steering is computed using a neural model. The main part of this model is the computation of the optic flow of the input sequence based on the biologically inspired approach of Bayerl & Neumann [9]. Two model areas derived from the motion processing in the human brain are simulated to achieve a robust flow estimation, namely areas V1 and MT that are part of the visual cortex. In both areas neurons tuned to specific velocities can be found. We simulate at every image position neurons responding to a velocity with a maximum of 30 pixels shift in both horizontal and vertical
Fig. 1. The robotic platform and the biological control loop with, in red, the pathway we focused on in this work. The two cameras mounted on the robot head provide the input images. In areas V1/V2 and MT/MST the optic flow computation takes place (Sect. II). The PFC (prefrontal cortex) represents the decision module whose output will be send to the PMC (primary motor cortex, Sect. III). Then the corresponding steering command will be effected by the robotic platform. The small image shows the 5 DOF head with which the robot was equipped including the two cameras.

direction ($\Delta x$ and $\Delta y$). The two areas are connected in a recurrent loop to stabilize and improve the optic flow estimates. Within the two model areas, the same principles of a three level processing cascade are employed. First, an integration step is done (Eq. 1), followed by a feedback modulation (Eq. 2) and, finally, the activity within the area is normalized using a mechanism of lateral shunting inhibition (Eq. 3, [10]).

$$\dot{a}^{(1)} = -a^{(1)} + net_{IN} \ast G(\text{space}) \ast G(\text{velocity})$$ (1)

$$\dot{a}^{(2)} = -a^{(2)} + (a^{(1)})^2 \cdot (1 + C \cdot net_{FB})$$ (2)

$$\dot{a}^{(3)} = -A \cdot a^{(3)} + a^{(2)} - \left( B + a^{(3)} \right) \sum_{\Delta x, \Delta y} a^{(2)}$$ (3)

In these equations, $a^{(i)}$ represents the neural activity of a neuron at a spatial position tuned to a specific velocity. $net_{IN}$ is the feedforward driving input, $G$ represents a Gaussian filter. Feedback from a higher area is depicted as $net_{FB}$, $A$, $B$, and $C$ are constants. These processing steps are derived from data of neurophysiological experiments and have previously been successfully applied to different tasks in visual processing (e.g., long-range grouping and texture boundary formation, [11]).

In detail, the iterative computation of the optic flow of a pair of two images is done in the following way: Firstly, initial optic flow estimates (called "hypotheses") are generated using a correlation based approach. These hypotheses represent the input $net_{IN}$ to V1. Then, the processing cascade is applied in V1 (Eq. 1-3). The first step integrates the activity of neurons (the hypotheses) in a spatial neighbourhood tuned to the same or similar velocities, then feedback from MT is used to enhance the activity at positions and for velocities that have been highly activated in MT during the last time step. Due to larger receptive fields (RFs) in area MT the feedback does not only affect the corresponding afferent neuron, but also a certain surround. This enables the spatial propagation of activity over time, which is necessary to solve the aperture problem, i.e., to achieve correct flow estimates along 1D structures. The normalization step implicitly enhances the activity of unambiguous features at a position as all neurons at one position have to share the overall energy (cmp. Eq. 3). After the processing cascade in V1 is completed, the V1 output is integrated in the larger RFs of area MT and then again normalized ($net_{FB} = 0$ for MT).

The integration in MT contributes significantly to the stabilization and robustness of the estimates. Also, the recurrent processing loop helps to stabilize coherent optic flow from iteration to iteration. At the same time, the optic flow estimation can recover quickly from sudden flow changes as the feedback from V1 to MT is only used as a modulatory input and not generating flow components if they are not currently detected in the feedforward input. This is also important to prevent the system from hallucinations that could be generated if feedback caused activation in lower areas without corresponding initial flow activities.

The model described so far works fine on both artificial and real input test sequences like, e.g., the Yosemite sequence. However, when the computation is used on a robotic platform, particular problems have to be taken into account. As the navigation in real-time requires very fast and stable optic flow computation, the model is calculated in an algorithmic way [12] that is accelerated due to a sparse representation of neural activity and a very efficient initial motion detection stage based on an extended version of the Census Transform [13]. Furthermore, the use of cameras on a robotic platform always includes shaking and jittering. This leads to strong rotational movements that are disturbing the relative movement of the surround that we wish to detect. To stabilize the optic flow we added a temporal integration of the detected optic flow. This reduces wrong estimates that just appear between two succeeding images and smoothes the overall estimation. In addition to the optic flow computation itself, we also use visual form information to improve the navigation with the model. An edge detection algorithm is applied to detect salient vertical structures in the image. As these might belong to an obstacle, the optic flow at these positions is multiplicatively enhanced to get higher weight in the navigation decision (Sect. III-B). Also, we compute the motion gradient in
A nonholonomic mobile robot (Fig. 1) has been developed to perform the experiments. This platform is equipped with a 5 DOF anthropomorphic head, thanks to which it was possible to position the two cameras in a slightly divergent configuration. In this configuration a field of view of about 75° is obtained. The robot can be moved by setting two commands. One command corresponds to the mean angular velocity of the two wheels, thus setting the forward speed, the other to a difference between the two angular velocities, corresponding to the rotational speed of the robot. A Polhemus Liberty tracking system (Polhemus, Colchester, VT) was employed for detecting the trajectory of the robot in the different tasks it performed. This device allowed the detection of the absolute coordinates of the robot in the environment. The following software structure was developed as interface between the robotic platform and the visual modules: A main framework was designed using the XVR development environment (VRMedia Srl, Pisa, Italy - www.vrmedia.it), responsible for the communication with the robot. Two modes are possible: The framework can either be applied on the robot using the images of the cameras as input and sending the motor command to the robot platform or a virtual environment is generated (also in XVR) where the robot movements are simulated. This represents the possibility to test the behavior in a surround that is not including noise and does not require real-time computation.

B. Feature extraction and discrete state machine controller

Many basic optic flow based control strategies try to balance left and right flow values, in order to keep the robot in the middle of a corridor-like environment [3]–[6]. The approach that was employed in this work was to use the average speed \( v_L \) and \( v_R \) in two regions in the left and right regions of the field of view as inputs to the control system as shown in Fig. 2. As for the navigation in a corridor only the horizontal components are important to guide the steering, we calculated the mean speed using only the velocity component in x-direction \( \Delta x \). To compute these values, first the mean speed \( \Delta v_{L,R} \) at each position \( i \) was calculated (Eq. 4) followed by the computation of the mean speed \( v_{L,R} \) in the whole left/right region (Eq. 5).

\[
\Delta v_i^{MT} = \frac{\sum_{\text{left, right}} a_i^{MT} \cdot \Delta x / \sum_{\text{left, right}} a_i^{MT}}{\Delta x / \sum_{\text{left, right}} a_i^{MT}} \tag{4}
\]

\[
v_{L,R} = \frac{1}{\| \{ L, R \} \|} \sum_{i \in \{ L, R \}} v_i^{MT} \tag{5}
\]

If the left flow is greater than the right one, the robot is closer to the left wall and will have to move to the right in order to keep in the center and vice versa. From this basic idea all the following control algorithms were developed. A third region, the location of the FOE that is here located in the middle of the field of view, was employed to calculate the mean rotational component, to be subtracted from all other flow values. This simple strategy can be applied here as the FOE can be calculated in advance because the cameras are fixed and not moving independently from the robotic platform within the experiments. During forward movement, the expansional flow field generated contains zero movement in this region, so the optic flow estimated there can only be due to an additional rotational movement. The strategy works well in the case of slow rotations, for strong rotations, nevertheless the compensation is not precise enough to compensate the effect completely.

As a first means to control the platform a discrete state machine controller was developed to enable an initial test of the possibility to move the robot in a corridor-like environment using only optic flow. Two values are computed from the left and right optic flows: \( d \) is the difference between the absolute left and right flow values; \( t \) is the sum of the absolute left and right flow values, divided by a normalization factor. The \( d \) value is a function of the distance between the robot and the center of the corridor. Positive values correspond to the robot
Our goal was twofold: to verify that the neuroinspired optic flow algorithm could be successfully used for driving the robot will not rotate for some time steps to enable forward movement leading to an expansional flow field, in order to let the flow stabilize to reliable values.

A control strategy employing continuous rotations leads to a more oscillating behavior as the rotational part in the flow field is overlapping the expansion flow field what disturbs the navigation strategy.

C. Development of a bio-inspired decision module

A basic bio-inspired decision module was previously developed [15], [16] for studying the process of decision making in primates. This module consists of two neural pools inhibiting each other (Fig. 3A). Inputs to the system are the mean left and right speed \( v_L \) and \( v_R \) (Eq. 5). The output command was the difference between the activities of the two pools. Compensation of self-rotation and additional variables for allowing flows to stabilize after every steering command are still employed. The equations which regulate the dynamic behavior of the two neural pools are as follows:

\[
\begin{align*}
\dot{\omega}_1 &= -\omega_1 + \frac{\alpha}{1 + e^{-b(w_L + w_R - v_L - v_R)}} \\
\dot{\omega}_2 &= -\omega_2 + \frac{\alpha}{1 + e^{-b(w_L + w_R - v_L - v_R)}}
\end{align*}
\]  

where \( \omega_1 \) and \( \omega_2 \) are the activities of the two neural populations, \( w_e \) is a self-excitatory weight, \( w_i \) is the weight corresponding to the reciprocal inhibition, \( \alpha, b \) and \( c \) are system parameters.

We have used this simple network for testing the possibility of controlling the navigation of the robot employing a bio-inspired decision module. The potential field of such a system is designed in such a way that, when the two inputs are balanced, there are two symmetrical stable states, corresponding to left and right rotation. Due to random oscillations in the activity of the network, the state of the system can change between the two stable states and, more often, stabilize in an additional equilibrium point between the two states, leading therefore to no rotation. When one input is stronger than the other, the system goes to the corresponding stable state and thus the output command will be a rotation.

The two-pool control system represents a first step toward a fully biological navigating device. The mechanism of setting the output to zero for some steps after each rotation command is still used and represents an artificial component. As this inhibition could also be performed by a third neural pool so that flows could stabilize autonomously, we added a third inhibitory neural pool (Fig. 3B) to obtain a fully functional bio-inspired control system. In order to obtain appropriate parameters for the network, we employed a genetic algorithm designed to minimize the distance between the output commands of the new network and those of the traditional control system [17]. The genetic algorithm explores the space of the network parameters (connection weights and cell parameters), by randomly choosing at step \( i \) a set of "chromosomes" (or individuals) and subsequently promoting their evolution through crossover and mutations. The fitness of each individual is set by calculating the difference between the output control sequence calculated with the standard algorithm and that calculated with the chromosome parameters, in the least square sense. The individuals with the best fitness are not modified for the next generation, the worst ones are substituted by completely new ones and all other undergo crossover.

D. Results

The performance of the different control systems was tested for both corridor centering and obstacle avoidance. Our goal was twofold: to verify that the neuroinspired optic flow algorithm could be successfully used for driving...
the mobile robot and to show that a neuroinspired control strategy has performances comparable to those provided by standard systems. Two different environments were developed for that purpose. The first environment was a curved corridor (Fig. 4), where the system’s ability to perform corridor centering was tested. The second environment was a straight corridor with two obstacles (Fig. 5), used for testing the control systems in an obstacle avoidance task. The obstacles were quite large (approx. the size of the robot), so that the optic flow could be sufficiently affected by their presence. In both environments walls showed a high level of textures in order to enhance optic flow detection.

Ten trials were performed for each control system in both environments. The robot was able to navigate in both environments without hitting neither the side walls nor the obstacles. In order to compare the performances of the different control systems, the variability in the robot trajectory was evaluated. To obtain a reference trajectory, ten trials were performed by manually driving the robot in the environments. It must be pointed out that the trajectories obtained in this way can be evaluated only in terms of path variability. Human subjects manually steering the robot were only instructed to keep it equidistant from side walls. The results obtained with the automatic control all showed the expected behavior, with a slight bias to remain closer to the inner side of the curved corridor. This tendency can be explained by the rather small field of view of the cameras that restrains the part of the inner wall that can be seen. For both experiments, the curve providing the best fit for the trajectories obtained by manually driving the robot were found and then used for the other control strategies. A 5\textsuperscript{th} order polynomial curve was employed for the corridor centering task, while a 3\textsuperscript{rd} order Fourier interpolant curve was found to be more appropriate for the obstacle avoidance task. The $R^2$ value and the root mean square error (RMSE) were then employed to compare the different performances (Table I).

The optic flow calculated with the neuroinspired algorithm resulted to be effective for driving the robot in a corridor-like environment and for avoiding large obstacles. The performance of the control strategies is very similar to that obtained by manually driving the robot. All control strategies tested allowed the system to perform very similar mean trajectories.

The path variability is always slightly lower for the manually driven robot, while the autonomous control strategies show globally similar performances. In the corridor centering task, the standard and two-pool systems are almost indistinguishable, the three-pool algorithm shows some more oscillations. In the obstacle avoidance task, the standard system shows a better $R^2$ value, however, all strategies have approximately the same RMSE value.

We can conclude that a fully neuroinspired control system for optic flow based navigation can be successfully employed for autonomous navigation of a mobile robot.

![Fig. 5. Trajectories performed in the obstacle avoidance task employing the different control strategies (blue curves) and fitted average path (red curve). Corridor walls and obstacles are plotted in black. Clockwise from top left: manual driving, standard control, 2-pool neural control, 3-pool neural control.](image)

TABLE I

<table>
<thead>
<tr>
<th>TABLE I</th>
<th>TRAJECTORY FITTING PARAMETERS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Corridor centering</td>
</tr>
<tr>
<td></td>
<td>Manual</td>
</tr>
<tr>
<td>$R^2$</td>
<td>0.9906</td>
</tr>
</tbody>
</table>

IV. TARGET REACHING

In this experiment the robot had to reach a target that was located in its field of view. The aim of this experiment was to verify whether we can achieve a result that is comparable to human behavior as investigated by Warren et al. [2]. In their experiments they measured that the trajectories that humans use to reach a target are not only influenced by the direction of the target, but also by the optic flow estimated during navigation towards the target: Optic flow information helps to navigate in a straighter path. They modelled this behavior with a second-order dynamics that describes the acceleration of the angle $\Phi$ between the direction in which the subject is moving and the target from a “bird’s eye view”.

In a first step we simulated a robot moving in a virtual environment (created with the XVR environment) towards a target. A slightly adapted version of the dynamics of the model of [2] and [14] was used to compute the steering
Fig. 6. Target reaching: The different trajectories of the robot are shown navigating without and with optic flow information. (A,B) The target is reached in a more direct way if the optic flow influences the trajectory. (C) If an obstacle is blocking the way towards the target, the robot starts earlier to avoid the obstacle and gets back faster to the straight heading direction towards the target.

The optic flow was again computed with the model of human optic flow estimation as described in Sect. II. Exemplary trajectories of such a navigational simulation are depicted in Fig. 6A/B to show the ways that are chosen with and without the information of optic flow. The results replicate the data measured by Warren et al. If no optic flow information is used the robot needs longer to reach the target as it is later turning towards the target in contrast to a straighter trajectory when the additional information is taken into account. We also simulated the behavior within a scenario where an obstacle impedes the direct information towards the target. The dynamics are extended by an additive component $u_O$ that describes the repellent effect of the obstacle (adapted from [14], $k_O = 198$, $c_3 = 6.5$, and $c_4 = 0.8$):

$$u_O = k_O \left(1 + w \|v\|\right) \left(e^{-c_3d_T} \left(e^{-c_4d_O} \left(\Phi - \Psi_O\right)\right) - e^{-c_4d_O} \left(\Phi - \Psi_O\right)\right).$$

In Fig. 6C the trajectory for this scenario is shown, also in this case the additional optic flow information leads to a straighter pathway, the robot starts earlier to avoid the obstacle and is faster turning back after it is passed.

V. DISCUSSION AND CONCLUSION

We presented a robotic system that consists of different processing parts that are all inspired from a biological background. This system is used to imitate different ways of biological behavior aiming at the two following issues: First, the behavior of the robot can be investigated from a technical point of view focusing on the successful performance of challenging navigation tasks; second, what we consider as even more important, we replicate and verify psychophysical experiments in the domain of navigation, to test the plausibility of models for animal and human behavior. The simulation of these experiments on a robot or in a virtual environment has the advantage that the interplay and the role of individual features for the task can be investigated by manipulating the connections of different model parts. Here, we presented the results for rather simple navigation tasks that will be extended in the next months to more complex scenarios step by step. In particular, we are interested in the role the optic flow plays in these navigation tasks. The fact that this information is used by animals is well established, e.g., for navigation [1], [2]. Several attempts have been made in the past years to replicate the results of such a control system. In contrast to our approach, most works employed traditional computer vision algorithms for the computation of the optic flow itself, as well as standard control techniques [3]–[6]. A few papers focused on studying and designing systems inspired on the neural system of flies and bees [7], [8]. The platform we developed is based, instead, on the nervous system of human and primates: the computation of the optic flow based on area V1 and MT of the visual dorsal pathway achieves in real-time optic flow estimates that can be used as reliable input for the decision module. We have tested the effectiveness of the calculated optic flow to guide
the robot using a standard control technique. Then, starting from a basic neural decision module, we have developed a fully bioinspired controller. Despite all the limitations imposed by controlling a robot employing only the optic flow information, the device was able to reach performances similar to those of platforms with signal analysis and control algorithms [3]–[6] in a corridor navigation task.

In the second experiment we addressed the navigational behavior of humans that want to reach a certain target. Warren and his collaborators showed that for this task both the direction of the target and the optic flow influence the chosen trajectory. We applied a slightly adapted version of the dynamics that they proposed [2], [14] in a simulation using a first person perspective view, i.e. we use input images corresponding to the input a person gets when he/she is walking in an environment. Using the optic flow of the sequence as additional input to these dynamics, we replicated the qualitative behavior of their model: When optic flow is available, it helps to find a straight way towards the target by stronger and earlier turns in target direction followed by a rather straight line of target heading. When an obstacle is blocking the way, the robot starts earlier to sidestep the obstacle and returns quickly to a straight path once the obstacle is passed.

In the future, we are planning to repeat the same experiment on the robotic platform itself. Furthermore, we want to increase the difficulty of the navigation task to investigate navigational behavior when objects of the scenario are moving themselves: Both the currently stationary obstacle and the target could be moving independently, resulting in a moving object avoidance and interception task, respectively. Subsequently, the data can be compared with psychophysical data from humans performing these tasks [18].

To conclude, we presented a robotic framework for navigation tasks that is provided with neuroinspired modules for visual information processing, in particular optic flow, and for decision-making. The system demonstrates navigational behavior for the replication of navigation based on a bee-like strategy and the use of optic flow for target reaching like measured for humans.

VI. ACKNOWLEDGMENTS

The authors would like to thank Edoardo Sotgiu and Florian Raudies for their valuable collaboration.

REFERENCES