

Obstacle avoidance and target acquisition in mobile robots equipped with neuromorphic sensory-processing systems

Moritz B. Milde, Alexander Dietmüller, Hermann Blum, Giacomo Indiveri, and Yulia Sandamirskaya
Institute of Neuroinformatics, University and ETH Zurich
Winterthurerstrasse 190, 8057 Zurich, Switzerland
Email: ysandamirskaya@ini.uzh.ch

Abstract—Event based sensors and neural processing architectures represent a promising technology for implementing low power and low latency robotic control systems. However, the implementation of robust and reliable control architectures using neuromorphic devices is challenging, due to their limited precision and variable nature of their underlying computing elements. In this paper we demonstrate robust obstacle avoidance and target acquisition behaviors in a compact mobile platform controlled by a neuromorphic sensory-processing system and validate its performance in a number of robotic experiments.

I. INTRODUCTION

Neuromorphic circuits harness some of the outstanding properties of biological neuronal networks, such as massively parallel distributed processing, clock-less event-based nature of computation, and interwoven co-localized memory and computing. These properties and the resulting speed and efficiency of computation are particularly well-suited for real-world robotic applications, in which large amount of sensory information has to be processed in real time.

Along with purely digital neuromorphic approaches [1], [2], neuromorphic processors designed using mixed signal analog-digital neuromorphic electronic circuits combine the energy efficiency and compact features of analog devices with the reliability of digital processing [3], [4]. However, analog circuits are affected by device mismatch and variability, e.g. due to temperature changes. Therefore it is necessary to adopt computational strategies and architectures that are tolerant to variability, noise, and temporal fluctuations. A possible solution to this problem that appears to be employed by biological neuronal systems lies in the use of attractor dynamics and population dynamics. Population dynamics provide robustness against mismatch and fluctuations in properties of neural circuits. Similarly, attractor neural dynamics have been shown to provide the required stability of the behaviorally relevant states – such states of the neural network that correspond to behavior – against sensory and neuronal noise [5], [6].

We present a proof of concept realization of these dynamics implemented using a mixed-signal neuromorphic processor, to process visual data sensed from a Dynamic Vision Sensor (DVS) [7], and to produce obstacle avoidance and target acquisition behaviors in a compact mobile robot. The neuromorphic processor makes use of the Real-time On-Line Learning Spiking (ROLLS) neural network chip, originally proposed and fully characterized in [3]. The robotic platform is a “PushBot” robot, developed by Jörg Conradt.

While similar recent attempts of using neurally inspired controllers have been proposed to learn sensory-motor associations with robots [8], [9], to plan routes in an unknown environment [10], or to control a robotic arm [11], this work represents the first attempt to realize obstacle avoidance and target acquisition behaviors on a real robot using a mixed signal analog-digital neuromorphic device and a neuromorphic vision sensor.

II. METHODS

A. The mixed-signal neuromorphic processor chip

The ROLLS neuromorphic processor comprises 256 Adaptive-Exponential integrate and fire (AdExp IF) silicon neurons [12], implemented using analog electronic circuits [13]. The neurons express biologically plausible neural dynamics including configurable refractory period, spike frequency adaptation, and time constant of integration. The 256 neurons on the ROLLS chip can be connected to each other and to external signals via three sets of synapses: each neuron has 256 programmable (non-plastic) synapses, 256 learning (plastic) synapses, and 4 auxiliary (“virtual”) synapses. The synapse dynamics are expressed using a current-mode Differential Pair Integrator (DPI) circuit, that behaves as a linear filter: incoming pre-synaptic spikes produce currents that have an amplitude proportional to the synaptic weight and that decay with a time-constant that is set by the DPI. The programmable and on-chip routing on the ROLLS that supports all-to-all connectivity allows us to implement any arbitrary neural architecture. However, the synapses can assume only one of 4 possible synaptic weight values, that can be programmed via a 12-bit temperature compensated bias-generator. An extra digital circuit allows the user to specify if the synapse is excitatory (positive weights) or inhibitory (negative weights). Careful analog design allowed to reduce the effect of device mismatch to an average of about 10 to 20% variability, depending on the bias settings chosen.

B. The vision sensor

The PushBot mobile robot is equipped with a DVS silicon retina. Each pixel of the DVS reacts asynchronously to a local change in luminance and sends out an event using the address-event representation (AER) protocol [14]. Every event contains the coordinates of the sending pixel (x, y), the time of event occurrence (t), and its polarity (pol : “on-event” or “off-event”). Due to the asynchronous sampling, the DVS is characterized by an extreme low latency, which results in μs time resolution.

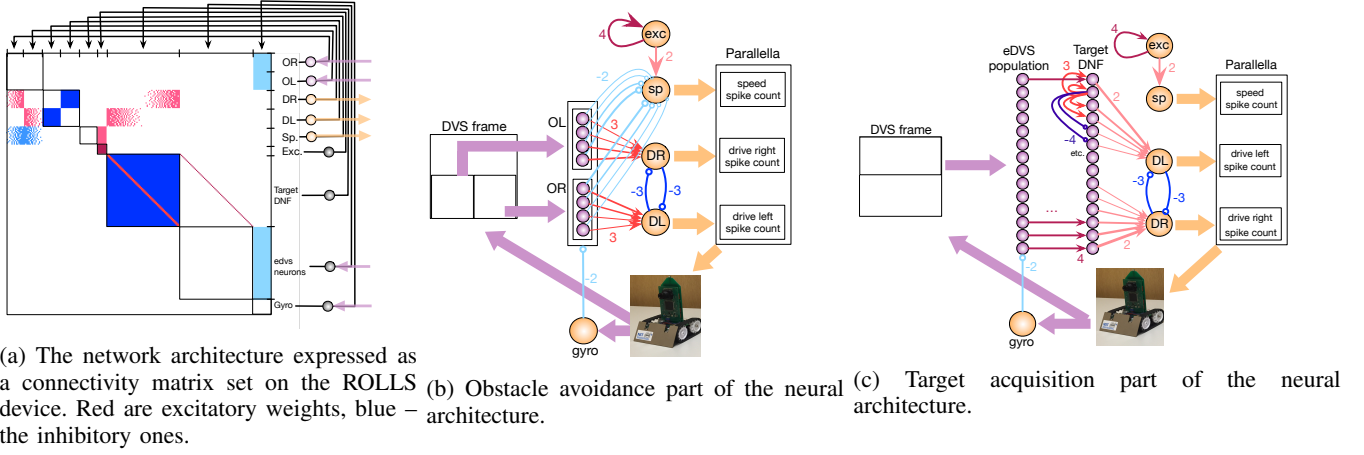


Fig. 1: The neural architecture for obstacle avoidance and target acquisition realized on the ROLLS chip.

As the DVS detects the spatio-temporal changes in a visual scene, a static camera only perceives moving objects. However, on a moving robot, the DVS produces a continuous stream of events at the objects’ boundaries, where change is induced by the sensor motion. The fact that the DVS also emits a fairly large amount of input-dependent noise makes the use of this sensor particularly challenging in navigation scenarios.

C. Neuronal architecture for obstacle avoidance and target acquisition

We designed a neural architecture that can cope with both the DVS noise and the ROLLS device mismatch in the same way. Each event of the DVS is mapped onto a virtual synapse of one of the neurons of neuronal populations on the ROLLS. This mapping is realised on a miniature computer (“parallela” board), used for the purpose of data logging and DVS to ROLLS event mapping. In the longer term, this board can be replaced by a more direct hardware interface between the two neuromorphic devices. The neural obstacle avoidance and target acquisition architecture is shown in Fig. 1. The figure shows the connectivity matrix which realizes the network architecture on the ROLLS chip, as well as a “connectionists” scheme of the obstacle avoidance and the target acquisition parts of the neural architecture.

1) *Motor outputs of the ROLLS chip:* We defined three populations of 16 neurons each that represent three corresponding robot commands: *speed*, *turn left*, and *turn right*. The *speed* population receives constant input from an additional constantly active population that represents the maximum speed. Spikes from the *speed* population drive the forward motion of the robot. The *turn left* and *turn right* populations drive the respective turning behavior of the robot. The spiking output rates of the neural populations are transformed into robot commands using a first-order low-pass filter with a sampling time of 50 ms. Multiplied by a scaling factor, these firing rates (one for each command population) are sent to the robot as velocity commands. This mechanism is in principle equivalent to how, e.g., motor neurons drive muscles in biological systems.

2) *Obstacle Avoidance:* We defined a group of 32 neurons to represent obstacles in the lower half of the camera field

of view (FoV). We divided the FoV into vertical regions (containing 4 columns of the 128×128 DVS pixels), each region providing input to one of the obstacle neurons. For every received DVS event in the assigned column the respective neuron receives a spike. If enough events arrive, the neurons start emitting spikes, thus signaling detection of an object. To turn away from an obstacle, the first 16 neurons representing the left half of the FoV are connected to the *turn right* population and vice-versa for the other 16 neurons. The turn-populations inhibit each other and thus implement competitive dynamics. The decision about the driving direction is made at this point. Furthermore, the obstacle detecting neurons inhibit the speed population, slowing the robot down if obstacles are present to ensure collision-free navigation.

3) *Dynamic neural field for target representation:* Similar to the strategy above, we defined an *edvs* population of 64 neurons receiving input from the upper half of the FoV (every neuron receiving events from 2 DVS columns). This population filters out much of the sensory noise, since only activity which persists over time (in contrary to the salt and pepper noise) can activate neurons.

Every neuron in the *edvs* population is connected to one neuron in a *target* population of the same size. To represent targets of the navigation dynamics, we use the principle of Dynamic Neural Fields (DNFs) [15], which can be easily realised in neuromorphic hardware by setting a winner-takes-all (WTA) connectivity network: Every neuron has excitatory connections to close neighbours while inhibiting far off neighbours. This DNF population implements working memory, remembering the target location in the DVS output even if the target vanishes from sight. The target in our experiments is a blinking LED of a second robot which is a salient input for the DVS and simplifies the segregation from the background (which is still non-trivial, as you will see).

4) *Fine tuning:* In order to achieve robust navigation, we modulated the weights strength between the obstacle and drive populations depending on the distance of the obstacle from the camera mid-line. We set weights of different strength for synapses from different obstacle neurons to the turn populations to make the robot turn faster if an obstacle is

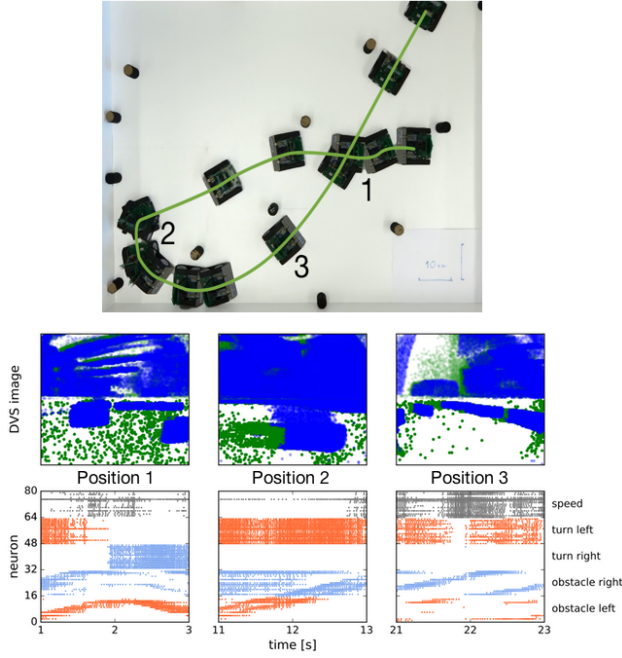


Fig. 2: Avoiding obstacles in a cluttered environment. **Top:** overlay of the overhead camera frames that depict the robot’s moving in a cluttered environment. **Middle:** DVS events at three points during the experiments (the points are marked with 1, 2, and 3 in the top pane). Green are off and blue are on events. **Bottom:** Output of the obstacle, turn, and speed populations of the ROLLS chip.

in front of the robot and slower if it is on the periphery. To overcome the limitation of the ROLLS chip in the limited (four) number of possible weight values, we used multiple sets of synapses. Thus, neurons representing an obstacle in the center project to all neurons in the turn population (16 in total), while neurons representing obstacles on the edge have only one connection. We use the same idea for the inhibition of the speed population: the robot slows down more for central than for peripheral obstacles. For target acquisition, we use an inverted scheme: neurons representing a target in the center have less connections since less turning is required. In this way, convergence on the direction towards the target can be achieved and the network effectively sets an attractor for the robot’s heading direction on the direction towards the target.

5) *Proprioception:* In this work we also used the Inertia Measurement Unit (IMU) device integrated in the eDVS to model proprioception and saccadic suppression. We read the measurement for angular velocity along the robots axis, sampled every 50 ms. This measurement was used to set the rate of stimulation for a *gyro* population on the ROLLS chip (16 neurons). Because of the nature of the DVS output, a greater number of events is produced while turning. We use the *gyro* population to inhibit all populations receiving DVS input (the obstacle and *edvs* populations), implementing a mechanism analogous to saccadic suppression in mammals.

III. EXPERIMENTAL RESULTS

Figure 2 demonstrates the obstacle avoidance behavior of the robot controlled by our neuronal architecture on the

ROLLS chip. The robot is put in an arena, in which a number of obstacles is arbitrarily distributed. The top part of the figure shows an overlay of camera frames from an overhead camera that allows to follow the robot’s trajectory as it navigates in this cluttered environment, avoiding collisions with objects and walls. The middle part of the figure shows the output of the DVS at three time points during this experiment (number 1, 2, and 3 in the top-view image). The DVS events are sampled for 1500 ms to show them as one image. The neurons on the ROLLS chip receive events from DVS pixel’s asynchronously in real time. Note the noisy and cluttered character of the output signal. The bottom plots in the figure show activity of the obstacle, turn, and speed populations on the ROLLS chip. Note how the speed population is inhibited when an obstacle is detected, how the obstacles are represented in a spatially resolved way by the obstacle left and obstacle right populations, and how the turn populations are activated to a different degree depending on the position of the obstacle relative to the midline.

Figure 3 demonstrates how the robot modulates the amplitude of the obstacle avoidance manoeuvre depending on the distance of the obstacle from the midline of the DVS frame. Here, the robot moves towards a cup, which is placed at different distances from line that would be the robot’s straight trajectory in the absence of the obstacle. The figure shows that the obstacle and the turn neuronal populations are activated stronger and for a longer time for the more central obstacle, leading to a more pronounced avoidance manoeuvre. For a peripheral obstacle, the robot only slightly changes its trajectory. This behavior emerges from the dynamics and connectivity of neuronal populations on the ROLLS chip and is not “programmed” algorithmically.

Figure 4 demonstrates the target acquisition behavior of the robot. In the presented experiment, the PushBot equipped with the ROLLS device approaches a second PushBot with a blinking LED. Since the upper part of the DVS FoV is used for target acquisition, many disturbing events are perceived by the DVS from the background objects outside the arena. The *edvs* population filters out much of the noise events, whereas the WTA connectivity of the *target DNF* population creates a stable localised representation of the single most salient (the LED) object.

IV. DISCUSSION

In this work, we presented a neuromorphic obstacle avoidance and target acquisition architecture, realized using low-power and low-latency event-based sensing and processing. This architecture allows to smoothly and reliably avoid obstacles and track the target object. We showed how redundant synaptic connectivity between populations of neurons can be used to cope with the low number of available weights values in the neuromorphic hardware used and enable to realize complex graded connectivity patterns with a limited number of synaptic weights. We also demonstrated how we can overcome the effects of device mismatch of the neuromorphic hardware by redundant computation using population dynamics and neuronal filtering. At the same time, the mismatch may be beneficial in a neuronal controller, as the variability would allow to avoid getting stuck in unstable fix-points of the dynamics, as well as facilitate exploration behavior in more complex scenarios. In summary we showed how

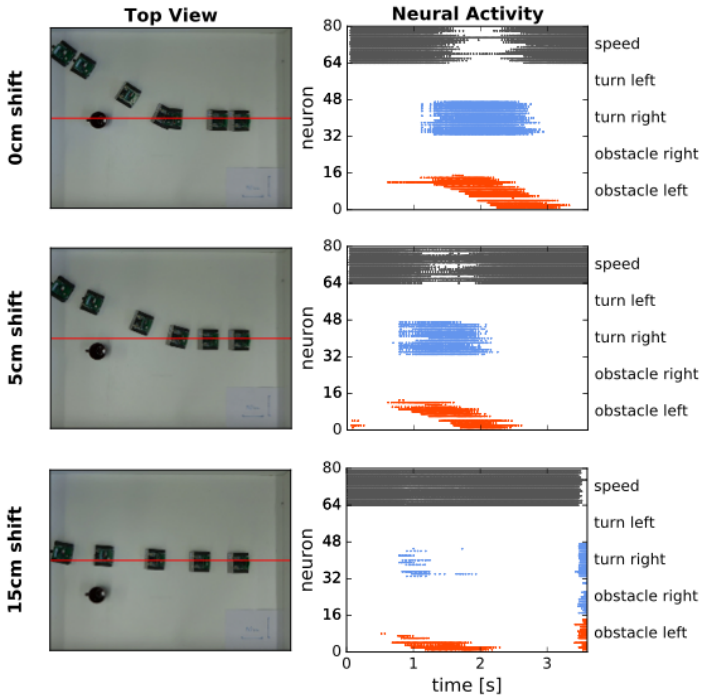


Fig. 3: Demonstrating the graded nature of the spatial representation of the obstacles in experiments with an object at different distances relative to the initial heading direction of the robot. **Left:** Overlays of the overhead camera images for three different runs of the experiment. The red line marks the line of the initial heading direction of the robot. **Right:** Activity of the obstacle, drive, and speed populations of the ROLLS neurons.

our system can produce robust robotic behaviors, despite the constraints and limitations of the neuromorphic hardware used.

ACKNOWLEDGMENT

We are grateful to Jörg Conradt for providing the PushBot robotic platforms. This work was supported by the EU H2020-MSCA-IF-2015 gr. 707373 ECogNet, the ERC-2010-StG gr. 257219 NeuroP, and the UZH grant FK-16-106.

REFERENCES

- [1] S. B. Furber, D. R. Lester, L. A. Plana, J. D. Garside, E. Painkras, S. Temple, and A. D. Brown, "Overview of the SpiNNaker System Architecture," *IEEE Transactions on Computers*, vol. 62, no. 12, pp. 2454–2467, 2012.
- [2] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, et al., "Artificial brains. A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science (New York, N.Y.)*, vol. 345, no. 6197, pp. 668–73, 2014.
- [3] N. Qiao, H. Mostafa, F. Corradi, M. Osswald, D. Sumislawska, G. Indiveri, and G. Indiveri, "A Re-configurable On-line Learning Spiking Neuromorphic Processor comprising 256 neurons and 128K synapses," *Frontiers in neuroscience*, vol. 9, no. February, 2015.
- [4] E. Chicca, F. Stefanini, C. Bartolozzi, and G. Indiveri, "Neuromorphic Electronic Circuits for Building Autonomous Cognitive Systems," *Proceedings of the IEEE*, vol. 102, no. 9, pp. 1367–1388, 2014.
- [5] W. Erlhagen, a. Bastian, D. Jancke, a. Riehle, and G. Schöner, "The distribution of neuronal population activation (DPA) as a tool to study interaction and integration in cortical representations," *Journal of neuroscience methods*, vol. 94, no. 1, pp. 53–66, dec 1999.

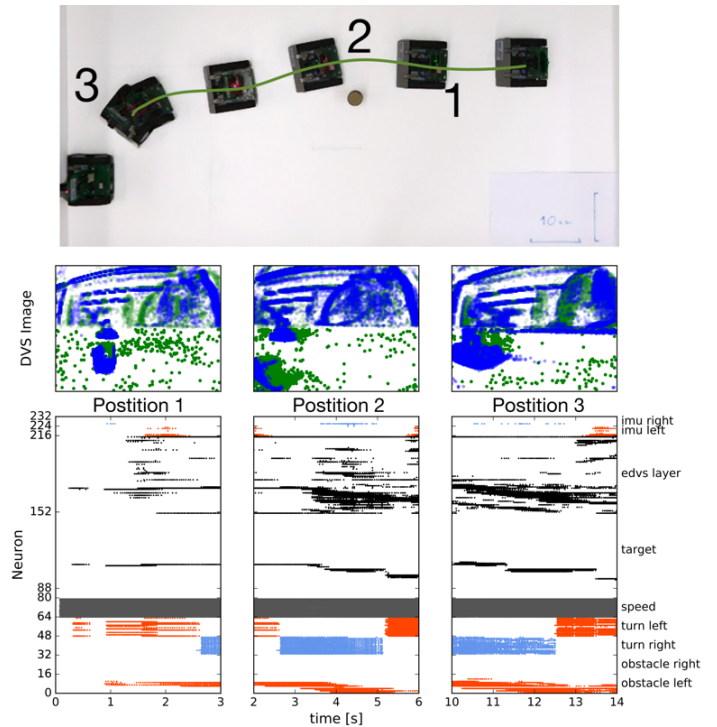


Fig. 4: Target acquisition behavior: the PushBot controlled by the ROLLS device approaches a second PushBot with a blinking LED, while avoiding a small obstacle on the way. **Top:** The overlay of the overhead camera; **middle:** the DVS events around the time points 1, 2, and 3; **bottom:** activity of all 256 neurons of the ROLLS chip.

- [6] W. Erlhagen and G. Schöner, "Dynamic field theory of movement preparation," *Psychological Review*, vol. 109, pp. 545–572, 2002.
- [7] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128 X 128 120db 30mw asynchronous vision sensor that responds to relative intensity change," *2006 IEEE International Solid State Circuits Conference - Digest of Technical Papers*, pp. 2004–2006, 2006.
- [8] J. Conradt, F. Galluppi, and T. C. Stewart, "Trainable sensorimotor mapping in a neuromorphic robot," *Robotics and Autonomous Systems*, vol. 71, pp. 60–68, 2015.
- [9] T. C. Stewart, A. Kleinhans, A. Mundy, and J. Conradt, "Serendipitous Offline Learning in a Neuromorphic Robot," *Frontiers in Neurobotics*, vol. 10, no. February, pp. 1–11, 2016.
- [10] S. Koziol, S. Brink, and J. Hasler, "A neuromorphic approach to path planning using a reconfigurable neuron array IC," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 12, pp. 2724–2737, 2014.
- [11] F. Perez-Peña, A. Morgado-Estevéz, A. Linares-Barranco, A. Jimenez-Fernandez, F. Gomez-Rodriguez, G. Jimenez-Moreno, and J. Lopez-Coronado, "Neuro-inspired spike-based motion: from dynamic vision sensor to robot motor open-loop control through spike-VITE," *Sensors (Basel, Switzerland)*, vol. 13, no. 11, pp. 15 805–15 832, 2013.
- [12] R. Brette and W. Gerstner, "Adaptive exponential integrate-and-fire model as an effective description of neuronal activity," *J. Neurophysiol.*, vol. 94, no. 5, pp. 3637–3642, 2005.
- [13] G. Indiveri, B. Linares-Barranco, T. Hamilton, A. van Schaik, et al., "Neuromorphic silicon neuron circuits," *Frontiers in Neuroscience*, vol. 5, pp. 1–23, 2011.
- [14] "The address-event representation communication protocol AER 0.02," Caltech internal memo, February 1993.
- [15] Y. Sandamirskaya, "Dynamic Neural Fields as a Step Towards Cognitive Neuromorphic Architectures," *Frontiers in Neuroscience*, vol. 7, p. 276, 2013.