

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/37427230>

Active Vision and Feature Selection in Evolutionary Behavioral Systems

Article · January 2002

Source: OAI

CITATIONS

53

READS

381

2 authors, including:



[Davide Marocco](#)

University of Naples Federico II

144 PUBLICATIONS 1,513 CITATIONS

SEE PROFILE

Active Vision and Feature Selection in Evolutionary Behavioral Systems

Davide Marocco

“Centro Interdip. della Comunicazione”
University of Calabria (UNICAL)
Arcavacata di Rende (CS), Italy
d.marocco@unical.it

Dario Floreano

Institute of Systems Engineering
Swiss Federal Institute of Technology (EPFL)
Lausanne, Switzerland
Dario.Floreano@epfl.ch

Abstract

We describe an evolutionary approach to active vision systems for dynamic feature selection. After summarizing recent work on evolution of a simulated active retina for complex shape discrimination, we describe in detail experiments that extend this approach to an all-terrain mobile robot equipped with a mobile camera. We show that evolved robots are capable of selecting simple visual features and actively maintaining them on the same retinal position, which largely simplifies the “recognition” task, in order to generate efficient navigation trajectories with an extremely simple neural control system. Analysis of evolved solutions indicates that robots develop a simple and yet very efficient version of edge detection and visual looming to detect obstacles and move away from them. Two evolved sensory-motor strategies are described, one where the mobile camera is actively used throughout the entire navigation and one where it is used only at the beginning to point towards relevant environmental features. The relationship between these two strategies are discussed in the context of the underlying visuo-motor mechanisms and of the evolutionary conditions.

1. Active Vision and Feature Selection

Visual processing for navigation in complex environments can be significantly simplified by selecting and paying attention only to a reduced set of environmental features. In computational vision, there are two major approaches to feature selection. One consists of equipping the behavioral system with a library of predefined feature extraction mechanisms, such as edge and motion detectors, depth from stereo-disparity, shape from shading, and landmark detection, to mention a few (Mallot, 2000, e.g.). The other consists of using some form of unsupervised learning in or-

der to reduce the large amount of visual information to a subset of statistically invariant and optimal features¹ (Hinton and Sejnowski, 1999). The latter approach seems more powerful when the environment where a robot operates is largely unknown and may significantly change over time, but currently available algorithms produce stable results only for well-behaved probability distribution functions of the input data or require off-line learning on a large set of available visual data.

None of these two approaches takes into account the fact that the number and type of visual features that an organism is sensitive to depend also on the sensory-motor and behavioral characteristics of the organism in its environment (Gibson, 1979). From the perspective of a designer, an interesting complication of behavioral systems is that behavior is both determined by visual information and at the same time affects what type of visual information is gathered. In fact, this apparent “egg-and-chicken” issue can be exploited to turn hard sensory discrimination problems into simpler ones. For example, it has been shown that the fruit fly *Drosophila*, which is equipped with a very simple neural circuit, is capable of performing complex shape recognition by moving in order to shift the perceived image to a certain location of the visual field (Dill et al., 1993). The process of selecting by motor actions sensory patterns which are easy to discriminate is usually referred to as *active perception* (Bajcsy, 1988). In computer vision, the importance of active gaze control to facilitate object recognition in complex scenes has been pointed out by (Ballard, 1991) and (Rimey and Brown, 1994). The latter paper, in particular, explores the use of Bayes nets and decision theory to optimally position a vision sensor in an image, but it takes advantage of prior knowledge of environmental relations and geometrical structure. More recently, in a series of experimental results, Nolfi (1998) and Scheier et al. (1998) have shown that autonomous robots that au-

¹So that the mean square root error between the original image and the *reconstructed* image is minimized.

tonomously evolve their behavior while freely interacting with the environment exploit active perception to turn hard classification problems into simpler ones. Using a problem classification theory developed by Clark and Thornton (1997), the authors showed that such evolved robots can indeed turn sensory classification problems of “type 2” (difficult ones) into problems of “type 1” (simple ones).

Within the context of vision processing, artificial evolution of behavioral systems is a powerful method to co-evolve feature-selection mechanisms and behavior of autonomous robots because it does not separate visual perception from behavior as in conventional system engineering methods (Cliff and Noble, 1997, Harvey et al., 1994). However, in all evolutionary experiments conducted so far the vision system is aligned with the body of the robot and cannot independently explore the environment while the robot moves around. A mechanically independent vision system complicates the sensory-motor coordination of the robot, but may provide improved navigation abilities by actively searching for simple features that can be useful for maintaining a smooth trajectory or maintaining in sight some cues in the visual field during navigation.

After summarizing the results of our recent work on evolutionary active vision for shape discrimination, we describe preliminary experiments that extend this approach to an all-terrain mobile robot equipped with an active vision system. We show that evolved robots are capable of selecting simple visual features and actively maintaining them on the same retinal position, which largely simplifies the “recognition” task of the system and generation of efficient navigation trajectories with an extremely simple neural control system.

2. Evolutionary Active Vision for Shape Discrimination

Recently, we have explored the idea of evolving an active retina capable of autonomously scanning the visual field in order to discriminate shapes with very limited computational resources (Kato and Floreano, 2001). Since the robotics experiments described in this paper represent an extension of that line of research, in this section we briefly summarize the main results.

Our system was inspired upon the evidence that humans and other animals asked to recognize shapes take their time to explore images with several rapid saccadic movements (figure 1) (Krupinski and Nishikawa, 1997, e.g.) instead of providing an immediate answer based on a single snapshot of the entire image (as most computer vision techniques do). These saccadic movements tend to sequentially foveate over salient areas of the image and we speculated that they might be useful (also) for simplifying the recognition problem by checking for the presence of simple features.

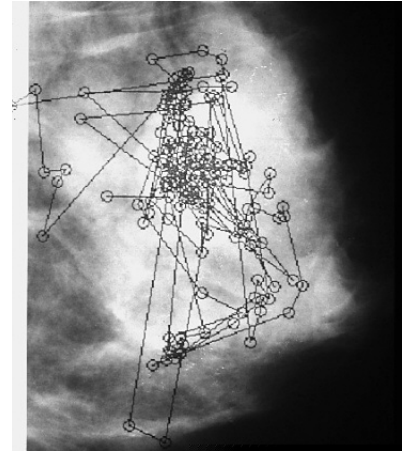


Figure 1: Patterns of eye movements of doctors scanning an X-ray image for the presence of breast cancer. Dots represent fixation points (Krupinski and Nishikawa, 1997). These and other images of human eye scans are available at <http://www.radiology.arizona.edu/eye-mo/mainpage.htm>

We devised a simulated neuron-based vision system where a subset of output neurons can move the vision acquisition device around an image, zoom in and out, and dynamically select the filtering strategy used for pre-processing (figure 2). The system consisted of a small simulated retina composed of a 3 by 3 matrix of visual cells whose receptive fields received input from a limited area of the image. Since the images were artificially created by the computer and had a limited size, an additional input neuron detected when the retina hit a border of the image (this neuron would not be necessary in a mobile robot immersed in an environment). The activations of the retinal cells were fed into a recurrent neural network without hidden units. The output units controlled the zooming factor (image area covered by the retina), the movements of the retina over the image (expressed as direction and distance from the current position), and the filtering strategy (average values of the pixels spanned by a vision cell or value of the the top leftmost pixel in the receptive field of a vision cell). In addition, two output units coded the shape discrimination response of the vision system, in this case being the presence of a triangle or of a square.

The synaptic weights of the network were genetically encoded and evolved to discriminate between squares and triangles. Each individual was presented with 20 noisy images where a triangle or a square appeared at random positions and had a random size. The system was free to explore each image for a maximum of 50 cycles (input computation, network activation, retinal displacement, zooming and filtering, and recording of shape discrimination). The fitness of each individual was proportional to the number of correct responses for each im-

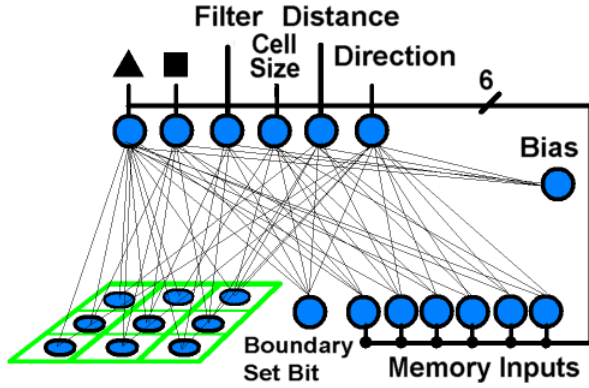


Figure 2: Neural architecture of the active vision system. Six output units receive signals from the retina cells and from a unit signalling whether the retina is against a border. The output units have recurrent connections, here represented as memory units that hold the activation of the output units at the previous time step (Elman, 1990).

age. In other words, this fitness encouraged the retinal system to provide the correct answer as soon as possible while exploring the image.

Best evolved individuals reported 80% correct response (100% correct response was impossible because for each image it takes some time for the retina to find where the shape is) and managed to discriminate correctly all shapes. Evolved systems started with a fixed response (square or triangle, depending on the evolutionary run) and then moved towards the shape. Once over the shape, the retina slid back and forth along one of its vertical edges. If the edge was straight, it set its response to square, otherwise to triangle. Figure 3 shows the trajectories of the retina in the case of two squares and figure 4 shows the trajectories in the case of two triangles. A variation on this basic strategy consisted of scanning the corners of the shapes, instead of the edges, to detect whether it forms an acute or rectangular angle. Once the shape had been correctly recognized, sometimes the vision system moved away from the shape towards a border of the image maintaining the correct response (this behavior was made possible by the recurrent connections). These selected features, edge inclination and angle, turned out to be invariant to size and location of the shape (the angle feature is also invariant to rotation).

In addition, we showed that a feed-forward neural network (whose input was the entire image and whose output encoded the shape class) trained with the back-propagation algorithm on the same set of images could not solve the discrimination task, not even when equipped with a variable number of hidden units. Although we do not intend to claim that back-propagation networks with hidden units in general cannot solve this

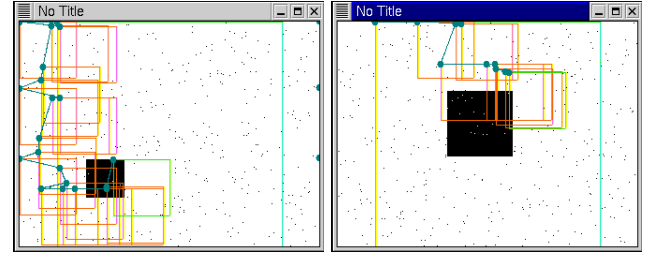


Figure 3: Examples of trajectories of an evolved individual. The retina always zooms and moves with respect to its top leftmost corner, here marked by a dot. The dots drawn after every retina movement are connected by a line. For graphical clarity, the values of the cells are not shown, only the retinal perimeter. *Left*: The retina starts with its initial size at the center of the image signalling “triangle”. It then shrinks to the top left corner and moves down towards the square where it slides along its left edge and starts signalling “square”. Finally, it explores the other three sides of the square maintaining the correct response. *Right*: The same individual begins signalling triangle and then moves towards the square where it visits the top and right edge changing the response into “square”.

task, the experiments showed that evolved active vision and feature selection turned non-linearly separable problems (type 2) into linearly separable ones (type 1).

3. Robotic Experimental Setup

The aim of these experiments was to extend the approach described above to a mobile robot equipped with a CCD pan/tilt camera (figure 5). The robot was positioned in a square arena and asked to navigate as far as possible without hitting the walls. The robot was controlled by a recurrent neural network whose input was a filtered black-and-white image coming from its mobile camera and whose output determined the movement of the robot, of the camera, and the type of filtering technique applied to the camera image. The goals of this experiment were to *a*) investigate the type of visual features exploited by a simple evolutionary neurocontroller without hidden units; *b*) study the emerging type of sensory-motor coordination used to control navigation and movement of the vision system; *c*) explore the emerging interactions between active vision and feature selection.

We used a mobile robot Koala by K-Team SA equipped with a Sony EVI-D31 mobile video camera and on-board PC-104 computer board. The Koala robot (figure 6) has three soft rubber wheels on each side, but only two motors each connected to the wheel in the middle which is slightly lower in order to generate more traction. The remaining two wheels on each side provide only physical support. This mechanical solution allows nav-

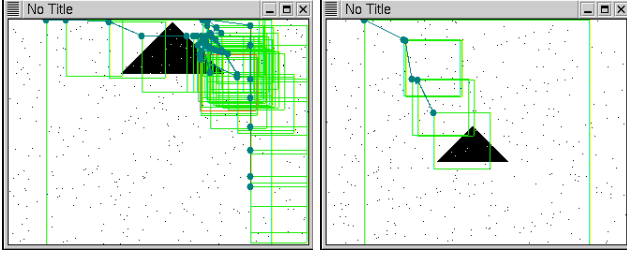


Figure 4: Examples of trajectories of an evolved individual, as in figure 3 above. *Left*: The recognition of a triangle is made by exploring its right corner and then drifting away while maintaining the correct response. *Right*: The recognition is performed by looking at the left edge of the triangle.



Figure 5: The Koala robot with the mobile camera and on-board computer board. The robot base is 30 cm w, 32 cm l, and 20 cm h; its weight is 4 kg and can carry a payload of 3 kg (camera and on-board computer are approximately 2 kg).

igation on rough terrain while maintaining the simplicity of a two-wheeled robot like the Khepera. The video camera is equipped with two motors that allow both horizontal movement (*pan*) in the range $[-100^\circ, 100^\circ]$ and vertical movement (*tilt*) in the range $[-25^\circ, 25^\circ]$.² The camera returns to the on-board computer rectangular video frames which are then cropped to a square matrix of 240 by 240 pixels and color information is discarded. The on-board computer performs image pre-processing, activation of the neural network, control of the motors of the robot and of the camera, the evolutionary algorithm, as well as fitness computation and data storage for off-line analysis. Details of the communication protocol and software are given in the appendix. The Koala robot also has 16 infrared sensors distributed around the body that can detect obstacles at a distance of approximately 20

²The Sony EVI-D31 camera also allows motorized control of zoom and focus, but we did not use this option in current experiments.

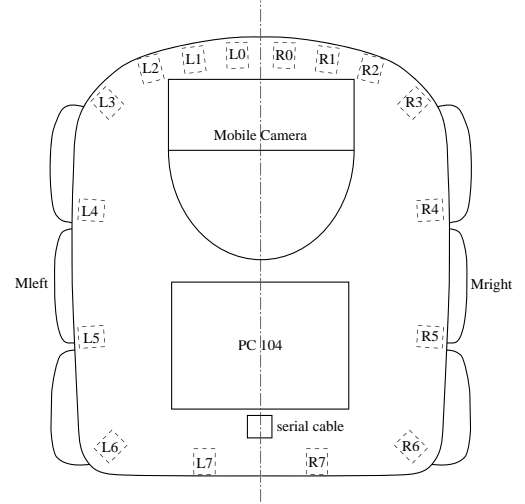


Figure 6: Sensory-motor layout of the Koala robot showing the position of the camera, of infrared sensors on left (L) and right (R) side, of the connection with the serial aerial line, and of the driving wheels (M_{left} , M_{right}).

cm. These proximity infrared sensors were used to reposition the robot to a random location between trials, but their activations were not given to the evolutionary neural controller and not even used for fitness computation. In order to provide continuous electrical power without using time-consuming homing algorithms or heavy-duty batteries, the robot was connected to a power supply through an aerial serial cable attached to the rear side of the robot and suspended rotating contacts.

The robot was evolved in a square arena measuring 200 cm by each side surrounded by 30 cm high white walls (figure 7). The arena was positioned in an office with a dark carpet floor and walls cover by posters, electric heaters, and windows. The office lights were kept on day and night. Near the arena was a desk with one or more researchers and visitors wandering around during the day while the robot evolved. All this visual information was potentially available to the mobile camera, but the robot was constrained to move only within its arena.

4. Neural Architecture and Evolution

Since a major goal of the experiment was to investigate the advantage provided by evolutionary active vision and feature selection with respect to computational complexity and resources, we employed a simple perceptron with recurrent connections to map visual input into motor commands. The neural network consists of 27 input units and 5 output units with discrete-time recurrent connections (figure 8). The input layer is an artificial retina of 5 by 5 visual neurons that receive input from a gray level image of 240 by 240 pixels. Visual neurons have non-overlapping receptive fields that receive



Figure 7: The evolutionary environment and the robot. The robot has visual access to the whole environment, but it can move only within the white arena. Lights were turned on day and night and researchers and visitors were free to come to the office during the evolutionary process. The pole on the back of the robot prevents the aerial cable from being trapped in the mobile camera. The other cable visible in this picture is used only after the evolutionary process to download data through ethernet from the PC-104 to a desktop computer for analysis.

information from a 48 by 48 pixels (240/5) area of the image. The state of visual neurons is determined by the activation of the *filter* output unit of the network. For activation values below 0.5, the state of each visual neuron corresponds to average grey level (scaled in the range $[0, 1]$) of the corresponding image patch (averaging filter). For activation values equal to or above 0.5, the state of each visual neuron corresponds to the grey level (scaled in the range $[0, 1]$) of the top leftmost pixel of the corresponding image patch (sampling filter). In addition, two proprioceptive input neurons encode the measured horizontal (pan) and vertical (tilt) angles of the camera. These values are in the interval $[-100, 100]$ and $[-25, 25]$ degrees for pan and tilt, respectively. Each value is scaled in the interval $[0, 1]$ so that activation 0.5 corresponds to 0 degrees (camera pointing forward parallel to the floor).

The activations of the output units are passed through the logistic function. Two output units determine the speeds of the wheels of the robot. In these experiments the speeds were set in the range $[-8, 8]$ cm/s. Activation values above 0.5 stand for forward rotational speed whereas activation values below 0.5 stand for backward rotational speed. Two output units encode the speed of the motor of the camera on the horizontal (pan) and vertical (tilt) planes in the same way described above. In this case, the maximum speed reachable in the hori-

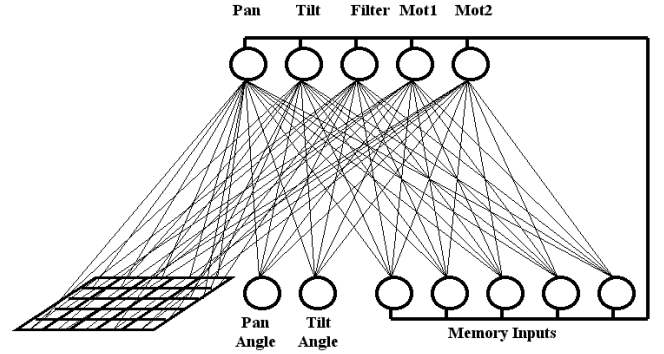


Figure 8: Architecture of neural network used in this experiment

zontal plane is 80 degrees/sec and in the vertical plane is 50 degrees/sec. If the camera has reached a maximum allowed position ($-100, 100$ and $-25, 25$ degrees for pan and tilt, respectively), output speeds in the same direction have no effect. The remaining output unit encodes the filtering strategy, as described above. Discrete-time recurrent connections are implemented using 5 memory units that maintain a copy of the activations of output units at the previous sensory-motor cycle (Elman, 1990).

The architecture of the neural network is fixed and its connection strengths and neuron thresholds are evolved using a simple genetic algorithm described in (Goldberg, 1989). Connection strengths and thresholds can take values in the range $[-4.0, 4.0]$ and are each encoded on 5 bits. The architecture described above has 160 weights and 5 thresholds. A population of 40 individuals is evolved using truncated rank-based selection with a selection rate of 0.2 (the best 10 individuals make 4 copies each) and elitism (a randomly chosen individual of the population is replaced by the best individual of the previous generation). One-point crossover probability is 0.1 and bit-toggling mutation probability is 0.01 per bit.

Each individual of the population is tested on the same robot, one at a time, for 2 trials each consisting of at maximum 200 sensory-motor cycles. Each sensory-motor cycle lasts 300 ms (during which the wheels move at the latest computed speed) during which the following sequence of operations is performed: infrared sensor reading, image acquisition and pre-processing according to the value of the output filter (most of the time and computational resources are spent on this step), activation of the network, motion of the wheels and of the camera, real wheel speed reading and fitness computation.

At the beginning of each trial the robot is relocated in the environment at a random position and orientation by means of a motor procedure during which the robot moves forward and turns in a random direction for 20 seconds. During a trial, if the robot gets too close to a wall (at a distance of less than 10 cm), the trial is

stopped and the robot is repositioned using the random motion procedure described above. This strategy is useful to prevent shocks that can damage the mechanics of the camera, the electronic contacts between the PC-104 and other components, and the head of the hard disk on the PC-104. In addition, it accelerates significantly the evolutionary process during early generations when most individuals tend to crash into walls and stay there.

The fitness function was conceived to select individuals capable of moving as fast forward as possible during the time allocated in each trial. Since the fitness function is computed and accumulated after every sensory-motor cycle (300 ms), robots whose trials are truncated earlier report lower fitness values.

The fitness criterion $\mathcal{F}(S_{right}, S_{left}, t)$ is a function of the measured speeds of the right wheel S_{right} and left wheel S_{left} , and of time t :

$$\mathcal{F}(S_{right}, S_{left}, t) = \frac{1}{E \cdot T} \sum_{e=0}^E \sum_{t=0}^{T'} ((S_{right}^t + S_{left}^t) - |S_{right}^t - S_{left}^t|)$$

where S_{right}, S_{left} are in the range $[-8, 8]$ cm/s and $\mathcal{F}(S_{right}, S_{left}, t) = 0$ if S_{right}^t or S_{left}^t are less than 0 (backward motion of the robot); E is the number of trials (2 in these experiments), T is the maximum number of sensory-motor cycles per trial (200 in these experiments), and T' is the observed number of sensory-motor cycles (for example 34 for a robot whose trial is truncated after 34 steps to prevent collision with a wall).

5. Results

The entire evolutionary process has been carried out on the real robot, each generation taking about 1.5 hrs. Both average and best fitness values gradually increased and reached a stable level after only 8 generations (figure 9). We ended the evolution at 15th generation because the observed behaviors were very similar across all individuals of the population. After the initial 8 generations, we noticed an alternation of two behavioral strategies across generations, one where the robot camera and body movements and one where the camera is maintained at a fixed position. The former behavioral strategy disappears after 12 generations although its fitness performance is equal to the latter strategy. Since the former strategy is quite interesting, we will describe both of them and discuss why it disappears in the discussion section below. We will start describing the strategy that uses the motion of the camera because the other strategy is a reduced and less general one.

Figure 10 shows the trajectory, camera displacement, and visual input of the best individual of generation 12. The robot starts in the position marked by the star. The *horizontal direction* (pan) of the camera is shown by long arrows plotted at each sensory-motor cycle. The greyscale matrices show the activations of the visual neu-

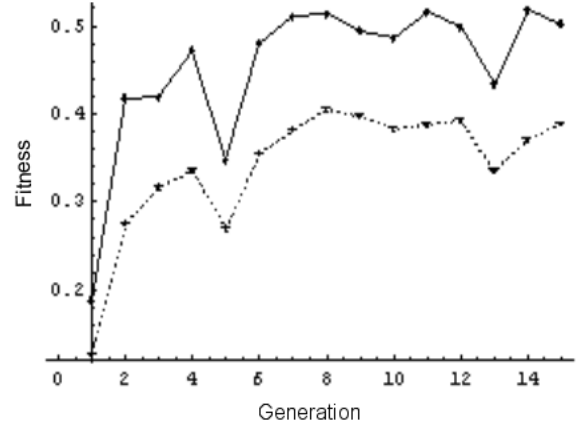


Figure 9: Fitness data scaled in the range $[0, 1]$. Fitness value is calculated from the speed of the each wheels of the robot according to function (1). The full line represent the performance of the best individual of generation. The dashed line represent the average performance of the entire population. Notice: the value 1.0 is not reachable, considering the characteristics of the fitness function, because the robot has to avoid the obstacles.

rons and, for sake of clarity, are plotted only before, during, and after avoidance of a wall and/or rotation of the camera to the opposite direction. This plot should be compared with figure 11 showing the horizontal (pan) and vertical (tilt) position of the camera as well as the output of the pan neuron and of the right motor neuron. The overall strategy consists of pointing and maintaining the camera downwards (thin continuous line in figure 11) so that the visual system can detect the edge between the dark floor and the white walls. In addition, the neural controller selects and maintains a *sampling* visual filtering (data not shown) so to enhance this brightness contrast (averaging filtering would blur out such contrast). This edge is clearly visible in the matrix plots of visual activations. The robot always follows a clockwise trajectory. The camera is moved to the left when the robot is approaching a wall on its left and then moved to right when the wall is sufficiently far away. While the camera is pointing to the right, it slowly scans back and forth and if a wall is detected at a certain distances, it moves to the left.

The values of output units encoding pan motor and right wheel motor are strongly correlated, as shown in figure 11, and are determined by the amount of white on the top two lines of the visual matrix. The closer the robot gets to a wall, the larger is the white area on the top portion of the visual field. This information is used to point the camera to the left facing the wall and slow down the rotation of the right wheel in order to turn right. As soon as the wall is sufficiently far away, the

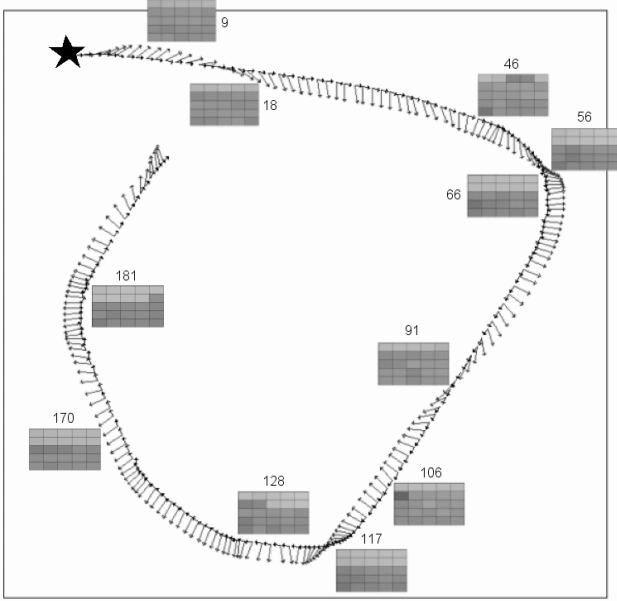


Figure 10: Robot trajectory (short arrows) and horizontal camera displacement (long arrows) of the best individual of generation 12 tested for one trial. Greyscale matrices represent the activations of the visual neurons (black = 0, white = 1) plotted before, during and after wall avoidance and camera movement. The numbers indicate the sensory-motor cycle corresponding to the visual plot and are should be compared with the graphs of figure 11.

robot shifts the camera to the front and then to right with slow scanning movements.

The alternative behavioral strategy (which becomes dominant after generation 12) is similar to that described above, but the camera is not actively used throughout the whole trial. At the beginning of the trial the robot points the camera downwards and to its left, and it keeps it there for the duration of the whole trial. The movement of the body is then sufficient to maintain the edge between the floor and the walls in sight and slow down the right wheel when it gets closer to a wall (signalled by the visual expansion of the white area on the top portion of the retinal image).

6. Discussion

The mobile camera allowed evolved robots to select two powerful, and yet computationally simple, visual features: *edge detection* and *visual looming*.

During the first couple of generations, robots were sensitive to a large variety of environmental features. For example, some of them watched the window while others were more “interested” in tall features (curious people watching the robot). However, selective attention to those features disappeared from the population because they were not sufficiently constant (window bright-

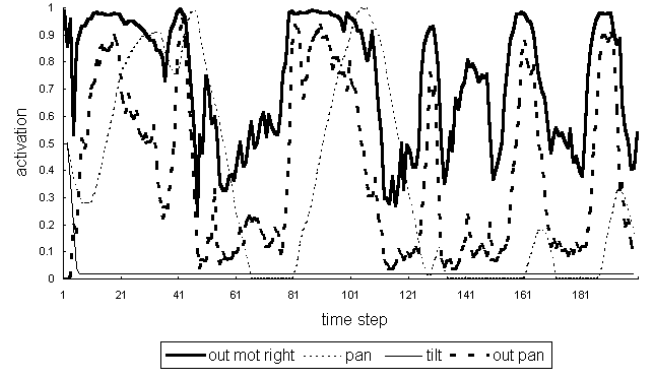


Figure 11: Pan and tilt angles scaled in the range $[0, 1]$ (0.5 means 0 degrees), and output activations of the right wheel motor and of pan motor across one trial. Tilt angle is always negative (-25°), meaning that the camera points downward. The values of the output units encoding the right wheel motor and the pan motor are correlated with the retinal input provided by the camera (see figure 10).

ness changes throughout the day, people come and go). Instead, the edge between the dark floor and white arena walls remained constant across generations and was therefore quickly selected becoming a dominant feature for all individuals of the population.

Wall avoidance was performed by correlating the expansion of the white area on the retinal projection with the rotational speed of the right wheel, which moved the robot away from the walls. This simple visual feature is known as “visual looming”, has been shown to be an innate feature of the avoidance behavior in rhesus monkeys (Caviness et al., 1962), and has been recently used as a computational strategy to measure distances in mobile robots (Sahin and Gaudiano, 1998).

Both edge detection and visual looming are instances of linearly-separable mappings between input and output space and therefore do not require multiple non-linear transformations. These results add to the growing literature showing that behavioral systems can exploit sensory-motor self-organization to actively select simple sensory stimuli in order to remain operational (Nolfi and Floreano, 2000).

Finally, it is interesting to notice that whereas evolved active retinas for shape discrimination selected *sampling* visual filtering only in 61% of the cases (Kato and Floreano, 2001), all evolved robots always used it in these experiments. The reason is that in the former experiments the contrast between black shapes and white background was much higher than in the images captured by the robot camera. In this case, the sampling strategy provides a sharper contrast between the brown floor and the walls.

Evolved robots displayed two behavioral strategies

across generations, one where the camera is actively used throughout the whole navigation and one where the camera is maintained at a fixed position. Although both strategies exploit the mobile camera to select the area of the visual field, use the same visual features, and display similar fitness values, the latter strategy does not require sensory-motor coordination between the movement of the robot and that of the camera. There may be two reasons why it becomes the dominant strategy after 12 generations. The first reason is that the environment does not change and therefore evolved individuals can select a simpler navigation strategy that fits exactly the environment where they have been evolved. The second reason is that the set of possible connection strengths that support suitable coordination between the movements of the robot and of the camera may be much smaller than that of connection strengths that control only the movement of the robot. Therefore, small random mutations are more likely to shift selected individuals towards the simpler strategy.

The first strategy, where the camera is moved throughout the whole trial, is initially selected and maintained as long as the trajectory of the robot is not tuned to the geometry of the environment. Since during early generations some individuals turn more sharply and may end up encountering a wall on their right side, it pays off to visually check both sides of the body. This strategy appeared to be quite efficient for more complex environments, as we observed in a preliminary test of the robot with 8 cm high wood bricks scattered on the floor, although we didn't make a comparison respect of the strategy without camera movement, but as we discussed above the evolutionary environment did not impose sufficient selection pressure to maintain it. Interestingly, this behavioral strategy, whereby the robot points the camera towards the direction where it is going (and not simply in front), is similar to that used by humans when steering a car at a turn. In those cases, we do not watch straight ahead, but always in the direction the car will take us (Lee and Lishman, 1977).

7. Conclusion

We have described an evolutionary active vision system capable of dynamically selecting relevant visual features for generation of suitable behaviors. Using reverse engineering of evolved mechanisms, the strategies used by the robot described here could be translated into a very simple and efficient algorithm for robot indoor navigation.

The results indicate that sensory-motor coordination in a computationally simple system can turn difficult recognition problems into simpler ones. Notice that this holds also for the evolved strategy where the robot moves the camera only at the beginning of the trial. Indeed, if the camera is not mobile, it is up to the experimenter to

design the environment or set the visual field so that the robot can perceive useful information.

We believe that the approach described in this paper may be quite powerful for more complex and rough terrains where the robot must actively search for and maintain selected features in sight as it moves. Preliminary tests on rough terrain with evolved individuals displaying camera/body coordination showed that robots successfully maintained the floor/wall edge in sight while moving up and down wood bricks. Rough terrains may also put stronger selection pressure on the development and maintenance of camera/body coordination.

Acknowledgements

The authors thank Claudio Mattiussi, Jesper Blynell, and Shloke Hajela for help with technical setup of the experiments. Thanks also to two anonymous referees for constructive questions and some bibliographic suggestion. This work was partially supported by the Swiss National Science Foundation, grant nr. 620-58049.

Appendix

The robot and the camera was controlled by an on-board PC-104 computer with 64 Mb of RAM and a CPU Pentium at 166 MHz. The operating system was Linux Mandrake 7.0. Low-level routines, such as PID motor control and sensor reading was performed by a Motorola 68332 microprocessor interfaced to the PC-104 by means of a MMA cable that allowed power supply and data exchange between the two processors. The command protocol used for motor control and sensor reading was provided by K-Team SA and is compatible with that used on the Khepera robot.

The camera was interfaced to the PC-104 board by means of an RS-232C serial communication cable using *VISCATM* protocol (*VISCATM* is an acronym of Video System Control Architecture). It is a network protocol designed to interface a wide variety of video equipment to computers. Under *VISCATM*, up to 7 EVI-D31 cameras can be connected to one controller using RS-232C communication. Protocol management was based on an informal technical report written by Thomas B. Moeslund (<http://www.vision.auc.dk/~tbm/Sony/>) because the documentation provided by the distributor of the camera was very poor. A framegrabber module installed on the PC-104 managed image acquisition and a modified version of Videodog software (<http://planeta.terra.com.br/informatica/gleicon/video4linux/videodog.html>) was used to format the data.

The entire algorithm and data storage was performed by the PC-104 board. At the end of an evolutionary run, the PC-104 was connected via ethernet cable to a desktop computer running Linux RedHat 6.0 in order to

download the data for analysis.

Software code and video clips are available at <http://dmtwww.epfl.ch/isr/east/>.

References

- Bajcsy, R. (1988). Active Perception. *Proceedings of the IEEE*, 76:996–1005.
- Ballard, D. (1991). Animate vision. *Artificial Intelligence*, 48:57–86.
- Caviness, J. A., Schiff, W., and Gibson, J. J. (1962). Persistent fear responses in rhesus monkeys to the optical stimulation of “looming”. *Science*, 136:982–983.
- Clark, A. and Thornton, C. (1997). Trading spaces: Computation, representation, and the limits of unified learning. *Behavioral and Brain Sciences*, 20:57–90.
- Cliff, D. T. and Noble, J. (1997). Knowledge-based vision and simple vision machines. *Philosophical Transactions of the Royal Society of London: Series B*, 352:1165–1175.
- Dill, M., Wolf, R., and Heisenberg, M. (1993). Visual pattern recognition in drosophila involves retinotopic matching. *Nature*, 355:751–753.
- Elman, J. L. (1990). Finding Structure in Time. *Cognitive Science*, 14:179–211.
- Gibson, J. J. (1979). *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston.
- Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Addison-Wesley, Redwood City, CA.
- Harvey, I., Husbands, P., and Cliff, D. (1994). Seeing the light: Artificial evolution, real vision. In Cliff, D., Husbands, P., Meyer, J., and Wilson, S. W., (Eds.), *From Animals to Animats III: Proceedings of the Third International Conference on Simulation of Adaptive Behavior*, pages 392–401. MIT Press-Bradford Books, Cambridge, MA.
- Hinton, G. E. and Sejnowski, T. J., (Eds.) (1999). *Unsupervised Learning*. MIT Press, Cambridge, MA.
- Kato, T. and Floreano, D. (2001). An evolutionary active-vision system. In *Proceedings of the Congress on Evolutionary Computation (CEC01)*, Piscataway. IEEE Press.
- Krupinski, E. A. and Nishikawa, R. M. (1997). Comparison of eye position versus computer identified microcalcification clusters on mammograms. *Medical Physics*, 24:17–23.
- Lee, N. D. and Lishman, R. (1977). Visual control of locomotion. *Scandinavian Journal of Psychology*, 18:224–230.
- Mallot, H. A. (2000). *Computational Vision*. MIT Press, Cambridge, MA.
- Nolfi, S. (1998). Evolutionary robotics: Exploiting the full power of self-organization. *Connection Science*, 10:167–183.
- Nolfi, S. and Floreano, D. (2000). *Evolutionary Robotics: Biology, Intelligence, and Technology of Self-Organizing Machines*. MIT Press, Cambridge, MA.
- Rimey, R. D. and Brown, C. M. (1994). Control of selective perception using bayes nets and decision theory. *International Journal of Computer Vision*, 12:2/3:173–207.
- Sahin, E. and Gaudiano, P. (1998). Visual Looming as a range sensor for mobile robots. In Pfeifer, R., Blumberg, B., Meyer, J., and Wilson, S., (Eds.), *From Animals to Animats V: Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior*. MIT Press-Bradford Books, Cambridge, MA.
- Scheier, C., Pfeifer, R., and Kunyoshi, Y. (1998). Embedded neural networks: Exploiting constraints. *Neural Networks*, 11:1551–1596.