A dynamic colour perception system for autonomous robot navigation on unmarked roads

Narayan, Aparajit; Tuci, Elio; Labrosse, Frederic; Alkilabi, Muhanad H. Mohammed

Published in:
Neurocomputing

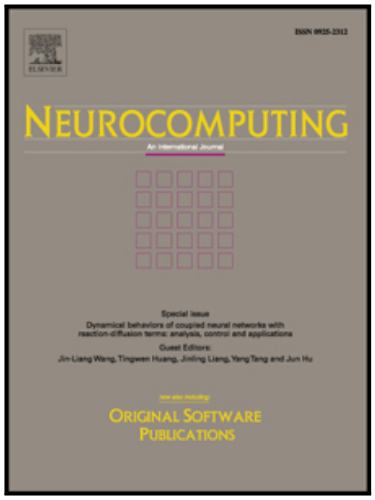
DOI:
10.1016/j.neucom.2017.11.008

Publication date:
2018

Citation for published version (APA):
Narayan, A., Tuci, E., Labrosse, F., & Alkilabi, M. H. M. (2018). A dynamic colour perception system for autonomous robot navigation on unmarked roads. *Neurocomputing*, *275*, 2251-2263. https://doi.org/10.1016/j.neucom.2017.11.008

Communicated by Prof. R. Tadeusiewicz

Accepted Manuscript



A Dynamic Colour Perception System for Autonomous Robot Navigation on Unmarked Roads

Aparajit Narayan, Elio Tuci, Frédéric Labrosse, Muhanad H. Mohammed Alkilabi

PII: S0925-2312(17)31722-8
DOI: 10.1016/j.neucom.2017.11.008
Reference: NEUCOM 19051

To appear in: *Neurocomputing*

Received date: 18 June 2017
Revised date: 31 October 2017
Accepted date: 3 November 2017

Please cite this article as: Aparajit Narayan, Elio Tuci, Frédéric Labrosse, Muhanad H. Mohammed Alkilabi, A Dynamic Colour Perception System for Autonomous Robot Navigation on Unmarked Roads, *Neurocomputing* (2017), doi: 10.1016/j.neucom.2017.11.008

A Dynamic Colour Perception System for Autonomous Robot Navigation on Unmarked Roads

Aparajit Narayan[a], Elio Tuci[b], Frédéric Labrosse[a], Muhanad H. Mohammed Alkilabi[a,c]

[a] *Aberystwyth University*
[b] *Middlesex University, London*
[c] *University of Karbala*

Abstract

Navigation on unmarked and possible poorly delineated roads where the boundaries between the road and the non-road surfaces are not clearly indicated is a particularly challenging task for autonomous vehicles. The results of this study show that fairly robust navigation strategies can be generated by a robot equipped with a dynamic active-vision based control system represented by an artificial neural network synthesized using evolutionary computation techniques. In the experiments described in this paper, a simulated Pioneer robot is required to visually navigate multiple poorly delineated roads that differ in terms of variations in luminance and/or chrominance between the road and the adjacent non-road areas. Low resolution camera images are processed by a mechanism that continuously adjusts the contribution of each component of a three dimensional colour model (e.g., R, G and B) to the generation of the robot perceptual experience. We show that the best controller can successfully drive a simulated Pioneer robot in environments with colour characteristics never encountered during the design phase, and operate with colour models never used during training. We show that the dynamic differential weighting of the colour components is underpinned by a complex pattern of neural activity that allows the robot to successfully adapt its perceptual system to the colour characteristics of different visual scenes. We also show that the controller can be easily ported onto real hardware, by showing the results of a series of tests with a physical Pioneer robot required to navigate various poorly delineated pedestrian roads.

Keywords: Active-Vision; Road-Following; Evolutionary Robotics

1. Introduction

Imagine a driverless vehicle delivering multiple packages autonomously. Such a vehicle may be required to drive the last mile in residential areas where roads may be missing their markings. Imagine another driverless vehicle carrying out inspections on unmarked roads in non-urban areas to check for irregularities of the road surface, or for the road lighting conditions. These are two potential applications for driverless vehicles that, by automating processes currently carried out by humans, can reduce costs (e.g., delivering a package over the last mile to its destination is considered to be one of the most costly stages of the entire logistics chain [36]), and improve efficiency (e.g., a driverless car could inspect roads 24/7). However, both applications face the challenge of driving on unmarked roads. How can such vehicles detect the road to safely perform their task?

We propose in this paper an answer to the above question by showing the latest results of a project focused on the design of control systems for mobile robots required

Email addresses: apn3@aber.ac.uk (Aparajit Narayan), E.Tuci@mdx.ac.uk (Elio Tuci), ffl@aber.ac.uk (Frédéric Labrosse), mhm1@aber.ac.uk (Muhanad H. Mohammed Alkilabi)

to autonomously drive on roads that are poorly demarcated. A road is considered unmarked when the boundaries between the road and the non-road surfaces are not clearly indicated, such as dirt roads or any other type of road without markings delineating traffic lanes. A major challenge for autonomous vehicles required to drive on such roads resides in the capability to distinguish the road from the non-road surfaces in order to avoid going off road. Within the broad research area of vision for mobile robotics, a large body of literature discusses solutions to the problem of autonomous road detection and road following, often focusing on the sensing aspects of the problem (i.e., the visual discrimination of the road from the non-road areas of an image), and by using a variety of methodological approaches. Some of these methods are not suited for poorly defined roads because they use features (e.g., road markings [9, 22]) that are often missing or not detectable from the visual scene. Moreover, the variability that can be faced by a vehicle that navigates a road is such that the road model needs to be constantly updated to respond to changes in illumination, and surface as well as localized features such as puddles and shadows (see [30]).

This study illustrates a method to design and test a control system that allows a Pioneer robot to autonomously navigate different types of unmarked pedestrian roads that feature a potentially detectable difference in colour between the road and the non-road area. We take an "active vision" approach based on the use of artificial neural network controllers synthesized by evolutionary computation techniques. These methods for an active approach to vision have been already described in other papers for different problems (see, e.g., [16][35]). The original contribution of our work is the design and testing of an integrated neuro-controller that processes low resolution colour camera images with a dynamic colour-perception mechanism, and drives the robot on the roads by setting the wheels speed. The dynamic colour-perception mechanism is a differential weighting mechanism that continuously adjusts the contribution of each component of a three dimensional colour model (e.g., R, G and B) to the generation of the robot perceptual experience. We demonstrate that this dynamic active-vision based control system allows a simulated robot to successfully operate in extremely variable environments (e.g., roads that differ in terms of variations in luminance and/or chrominance between the road and the adjacent non-road areas). We show that, once successfully trained with RGB, the dynamic colour perception mechanism can successfully weight the contribution of any combination of three colour components, by allowing the robot to operate with a sensory apparatus different from the one used during training. We demonstrate that this feature improves the robustness and adaptability of the robot. We show that the dynamic differential weighting of the colour components is a process underpinned by a complex pattern of neural activity that allows the robot to successfully adapt its perceptual system to the colour characteristics of different visual scenes. We also show that the controller can be easily ported onto real hardware, by showing the results of a series of tests with a physical Pioneer robot required to navigate different poorly delineated roads.

In Section 2, we briefly review the research work in the literature that we considered relevant to this paper and highlight the need to explore an alternate approach using an adaptive neuro-controller with active vision capabilities. We have kept the focus of our review on works using machine learning, neural network and active vision based methods. A more detailed review of "engineered" methods and colour models used in road detection can be found in [30]. In Section 3, we describe the simulation model used to develop the controllers, the nature of the control systems, and the evolutionary machinery used to generate controllers. In Section 4, we describe the results of the evolutionary process, and we illustrate and comment on the results of a series of tests with the best evolved controller in simulated and in real world conditions. In Section 5, we draw the conclusions, we discuss issues arising from the results of our experiments, and we identify interesting lines for future work.

2. Related work

For a truly autonomous driving control stack, it is crucial to develop a solution to robustly detect and follow roads across all environmental conditions, irrespective of the nature of road/non-road surfaces and prevalent weather and lighting conditions. This has been an area of focus for researchers across decades with a wide array of proposed solutions, some effectively demonstrating automatic driving in challenging conditions. A review

covering the entire spectrum of literature relevant to road-following is beyond the scope of this work. In this section, we illustrate some of those research works which have contributed to the state-of-the art improvements and development of promising techniques in this field. Most works rely on modifications to standard signal processing, statistical and probabilistic methods used across various computer vision problem domains, while others attempt to develop control solutions based on machine learning and artificial neural network models.

A number of works have looked at the problem of visual navigation in poorly delineated roads by segmenting the colour distribution of the input image into road and non-road areas. This approach is particularly effective since it does not depend on features such as road markings and is also flexible enough to be applied on a much wider range of road environments. In [2], the authors first transform the raw image into an illuminant-invariant colour space to reduce the effect of shadows. They then feed the histogram of the transformed image to a classifier which labels each pixel as either belonging to the road or non-road areas. The authors successfully test their method on images corresponding to a variety of dynamic lighting conditions on urban roads. The work presented in [33] details a segmentation based approach, where instead of using histograms, the authors use a mixture of Gaussians in the RGB colour space. As the robot drives, newly captured images are used to update the road-model by adding or discarding Gaussians. This method ensures that the robot can adapt to changes in the road surface and lighting conditions. This road detection method is used in conjunction with readings from a laser range finder used to detect traversable areas. This road detection algorithm is part of a larger autonomous driving system used to control the robot that won the DARPA grand challenge in 2006 by driving through a highly delineated desert course. In [30], the authors describe a similar method to allow an autonomous vehicle to successfully drive on a variety of poorly demarcated roads by using a shape-constrained geometric model of the road, rather than segmenting the entire image, along with a Gaussian model of the road colour. The authors propose an adaptive method that uses the Mahalonobis distance between the colour model of the road and pixels of the images. The geometrical model is fitted to the current image by minimising the Mahalanobis distance and maximising the width of the detected road.

The accuracy of such colour-based techniques to build a dynamic colour model of the road is influenced by the number and type of colour components used. In [30] it is shown that the higher the variability in operating conditions, the more likely any fixed set of channels fails in discriminating road from non-road areas. There also has to be considerable forethought on part of the designer to ensure robustness in both the strategies for extracting features from raw inputs and the targeted features themselves. A hand designed controller would often reflect the designers own biases failing to account for the sheer amount of variability in operational environments. Despite this, works such as [30, 33] have indeed demonstrated a high degree of accuracy in challenging delineated conditions by maintaining a dynamic and adaptive representation of the road. However room for improvement exists as there are foreseeable scenarios where strong discriminative cues necessary for these methods to extract the road-shape may not exist.

In recent years, machine learning techniques have been proposed as a potentially effective solution to the problem of road detection in autonomous driving vehicles required to operate in noisy and highly variable real world conditions. These techniques with their ability to "learn-by-example" can in theory learn a global, generalised representation of roads, without prior human biases and restrictive assumptions that limit the performance of so many of "hand-crafted" methods. In [19], image segmentation is carried out by a support vector machine (SVM) which classifies pixels as belonging to either road or non-road classes. Before the robot starts navigating, part of the road is selected from the initial frame. The pixels belonging to this region are used to train the SVM classifier. In the subsequent frames, additional road-pixels are added to the training samples (and old ones discarded), based on assumptions about the structure of the road. The classifier is thus continuously retrained and updated to adapt to changing properties of the road and to recover from poor initial misclassification. Another method relying on SVMs is described in [42], where the authors use SIFT features, Histogram of Oriented Gradients and Local Binary Patterns for classification instead of pixel colour information. Classification errors can however arise in com-

plex environments and dynamic weather conditions when the pixels in the SVM training no longer represent the entire range of pixels on the road surface. The authors acknowledge the complexity of devising a re-sampling procedure and feature representation that could make the SVMs predictions robust to sudden changes in illumination and colour properties.

A number of recent works such as [26, 41] have used particle filters for estimating road boundaries in unmarked, delineated roads. Particle filters are an approach used for dynamic state estimation, i.e, predicting state of a system based on information about the current state. Weighted probabilities are set for likely candidates (probable road regions in the case of autonomous driving) in the output-parameters search space. An interative search process follows this, through which the distribution of probable road shapes is narrowed down to a range within acceptable error limits. The road detection accuracy of such systems relying on particle filters is affected by the variance of the estimated particle distribution, which in turn depends on the reliability of their extracted feature cues. In other words, estimation accuracy and uncertainty is tied to the degree of discernible information available in the selected feature space. For example, in [26] the choice of colour saturation values as one of the feature cues is based on the assumption that the more coloured areas in the image correspond to regions outside the road. This holds for certain environments (due to the presence of vegetation for example) where this work is evaluated in (e.g. forest paths with vegetation at the sides), but may not generalise across the broader range of road following scenarios.

One of the earliest examples of using neural networks to control an autonomous vehicle on real outdoor roads is the ALVINN project [32]. The controller consists of a 3 layered feed-forward neural network taking in a grey-scale image as input and outputting the needed turning in order for the vehicle to remain on the road. The system is first trained using back-propagation on data generated by a human controller navigating in a road-simulator (based on real road images) developed by the authors. Later networks are trained "on-the-fly", using images and control commands from real-time driving on outdoor roads. After an initial period of training on a particular type of road surface the network takes over control and drives autonomously, demonstrating that it has learned the correct mappings between input images and steering directions. However, one of the major limitations with the network is its inability to generalise across road environments different to what it has been trained for. To overcome this limitation, the same authors propose a new modular architecture called MANIAC [20]. MANIAC is a module architecture consisting of several individual networks trained with back-propagation under different types of roads. After training, the modules are integrated in a single structure capable of driving a vehicle autonomously, demonstrating that it has learned the correct mappings between input images and steering directions. While being capable of driving in multiple road environments, the system is still limited in the sense that accurately representing the entire range of possible road types requires a progressively larger number of individually trained modules, and each time a new module is added to the system, the entire structure require retraining.

In [39] and [34] controllers consisting of multiple network modules operating within a larger structure are proposed. In both studies, the controller generates the robot trajectory by parsing images that are pre-processed with a feature extraction algorithm. Training is supervised and done off-line with annotated images. These neural networks, trained for specific road types, can adapt to further sections of already trained roads, but they cannot operate on previously unseen environments without further training. A more flexible solution is therefore needed to tackle the case of generic, poorly delineated, roads.

With the growing usage of deeper neural network architectures [24], and wide availability of high performance computing resources, researchers have been trying to apply deep convolution neural networks [23] to various instances of the problem of detecting and navigating on roads by autonomous vehicles (i.e., the road following problem). Most of the work using convolution neural networks (CNN) for automatic driving has been focused and evaluated on highways and urban roads (see [5], [25]). These CNN based solutions have been shown to outperform a benchmark based on predefined GIST features [29] on such environments. An exception to this is the work described in [14], where the authors developed a controller able to autonomously drive on highly delineated off-road terrains. A convolution network (trained off-line) is used as a feature extractor providing a robust representation of

complex environments. The terrain in front of the robot is segmented into multiple categories by a classifier, which is trained on-line using self-supervised data labels. Another deep convolution network based work for non-urban environments is described in [40]. The authors propose integrating feature sets extracted from separate networks dedicated to different input modalities (RGB, depth and infra-red images) for segmenting highly delineated forest paths.

In [3], a de-convolution (convolutional encoder-decoder) network is used for semantic pixel-wise image labelling. The results show that the network is capable of successfully parsing and segmenting urban road scenes into multiple categories such as roads, pavements, trees, etc. The method described in [1] also relies on scene segmentation by a convolution network detecting the road. Classification from the network is aggregated with that from a statistical colour-information based method. With this approach, the generalisation capabilities of the convolution network which has learned high-level features from other road scenes, is combined with the colour-based method which adapts to dynamic changes in the current road.

To the best of our knowledge, convolution neural networks such as those described in [3], [5] and [1] have not been evaluated on non-urban roads, and their performance in such environments is yet to be ascertained. We have tested the system described in [3] on various images of poorly delineated roads, some of which taken from the environments where we conducted the experiments described in this study. The performance of the system turned out to be relatively poor. The results of this test are available in the supplementary document (available at `https://www.aber.ac.uk/en/cs/research/ir/dss/#road-driving`). The multi-modal feature fusion approach described in [40] whilst promising, especially with the use of features extracted from multiple sources, also needs further evaluation in more varied and complex conditions (as mentioned by its authors). These networks trained specifically for forest roads and off-terrain tracks may not be able to generalise to urban and high-way road scenes. Indeed as shown by the experiment involving cross-dataset evaluation of deep convolution networks published in [37], these models tend to specialise very well to the datasets used for training/validation but may perform poorly in others despite the same labels and categories being present. While fine tuning the above mentioned CNN models may enable them to achieve low errors in images similar to those in the re-training image set, it may not result in robust detection across the entire spectrum of road scenarios.

This suggests that developing road detection strategies that can be successfully applied to a broad spectrum of real world operational conditions is a quite difficult tasks, due to the large variability in operating conditions. Designer biases related to the nature of the visual features used for discrimination, or lack of sufficient variability into the training set can significantly limit the robustness of the controllers required to perform visual discrimination tasks. Moreover, as pointed in [31], important components of advanced cognitive vision systems such as recurrent feedback and time-dependent processes tend to be omitted in deep convolution neural network based models described in the literature. As a consequence, these models lack the ability to use temporal structures to extract regularities in the environment. Observations of detection failures from ongoing experiments carried out by us for road detection with deep CNNs (see [27]) also indicate a need to investigate alternative road following/detection approaches beyond CNNs. In this study, we propose to overcome these limitations by considering an alternate line of research that considers the problem from the perspective of a controller with relatively simplistic visual apparatus (and computational requirements for implementation) that has developed behavioural properties required for "actively" extracting required sensory cues across varied real-world unmarked road environments with complexities much beyond those encountered in training.

2.1. The active vision approach

The theory of active perception is based on the assumption that any perceptual process depends as much on the sensory apparatus characterizing an organism as on its motor activity. Within an active approach to perception, active vision refers to the sequential and interactive process by which an agent "actively" selects and analyses parts of the visual scenes [28]. The famous "Kitten in the Gondola" experiment illustrated in [18] clearly demonstrated that normal visual development depends not only

on movement of the body relative to the environment, but also on self-actuated movement. In the last 30 years, a fast growing body of literature in psychology and neuroscience forwards the case of looking at action and perception as part of unified closed sensory-motor loop. In the domain of robotics, the active perception paradigm has been implemented following different approaches, in particular for vision-based tasks. For a comprehensive review of active vision in cognitive science and robotics we refer the reader to [8]. In this section, we focus mainly on a particular approach to active vision based on the use of artificial neural network controllers synthesized by evolutionary computation techniques. With this approach, the neuro-controller is an integrated action-perception system in which the camera images, largely reduced in resolution, contribute to form the controller input vector. The sensory information is propagated forward to the network output layer, which sets the speed of the robot wheels, and/or the pan and tilt orientation of the camera, and/or the level of the zoom. In this system, the agents actions contribute to altering its perception of the scene and thus in turn affects the visual inputs it receives. This is referred to as the close sensory-motor loop. Various papers have investigated the potentialities of this approach in different vision-based tasks, showing that the integrated neuro-controllers that close the action-visual perception loop manage to overcome the limitations imposed by the use of low resolution images to allow autonomous robots to perform complex visual discrimination tasks (see for example [16]). A significant contribution to the development of the active approach to robot vision has been made by the numerous research works carried out at the Laboratory of Intelligent Systems, EPFL Swiss Federal Institute of Technology (see for example [21], [11] [35]).

Our work is largely inspired by the above mentioned work carried out at EPFL. The distinctive contribution of this study is to develop in simulation and test in real hardware an integrated dynamic neural network controller with an adaptive differential weighting mechanism that determines how much each colour component (e.g., R,G,B) from the raw camera images contributes in generating the final network input vector. After evaluating the network's performance using different combinations of colour models in a vast range of simulated roads that differ in combination of colours of the road and non-road surfaces, we demonstrate that the network can reliably navigate a real robot in real world roads. The differential weighting mechanism embedded into the active control framework provides the network the plasticity required to adjust and to cope with the large environmental variability of the test conditions. An analysis of the robot controller shows that the colour parameters vary in a complex but also effective way in order to capture structures and regularities in extremely low pixel resolution images.

3. Methods

A Pioneer 3-AT 4 wheel skid steer all terrain robot is used to navigate a variety of roads using visual input. The controller is a dynamic neural network synthesised in a simulated environment using evolutionary computation techniques. In this section we describe the main features of the simulation model used to evolve the controller.

3.1. Simulated environment

In the simulation environment, the robot is modelled as a circular object (radius 25 cm) with left and right motors which can be independently driven forward or backward, allowing the robot to turn fully in any direction. The robot position is updated using the differential drive kinematic equations described in [10]. The simulated robot maximum speed is 0.8 cm/s. A virtual camera, positioned on the front-end of the robot body, looks at a 3D scene rendered using OpenGL (`http://www.opengl.org`). The field of view of the simulated camera is set in order to match the characteristics of the camera mounted on the real robot. The scene contains a tiled textured horizontal plane that represents the ground, and a texture deviated surface, rendered on the ground, that represents the road. At each simulation time-step, a 500×500 pixels image is captured from the virtual camera. The image resolution is significantly reduced by overlaying a 5×5 grid on the image. Each of the 25 grid cells covers an area of 10,000 pixels. For each cell, the normalized mean value of each RGB colour component (C_R, C_G, and C_B) is computed by first summing the respective colour component of each pixel within the cell and then dividing by the number of pixels in the cell. Each cell i generates a sensory input $I_i \in [0,1]$ by combining its mean colour components as

follows:

$$I_i = \rho \times C_R + \gamma \times C_G + \beta \times C_B. \quad (1)$$

The parameters ρ, γ, and β are real numbers in $[0,1]$ generated by the robot controller at each time-step (see Section 3.2). Their value is such that $\rho+\gamma+\beta = 1$ to represent the ratios in which the mean colour components are mixed in forming the input of each grid cell. By varying ρ, γ and β at every update cycle of the neuro-controller, the robot can dynamically adapt its perceptual system to the colour characteristics of the environment. One of the challenges of this research work is in finding the evolutionary conditions to design neuro-controllers that, by dynamically varying ρ, γ and β, generate the perceptual information required to successfully drive a robot in varying colour scenes.

The process of compressing a 3-channel full resolution input image into 25 real valued numbers is carried out to limit the neural networks input field resolution to a reasonable size of 25 nodes. The neural architectures used for mobile navigation and simulated autonomous driving in the experiments described in [35] also used a similarly sized architecture requiring 25 input values. Additional input nodes imply increased number of trainable network parameters which leads to a more complex and higher dimensional search space for the evolutionary optimization algorithm. Moreover the rather extreme reduction in input resolution also works as a filtering mechanism by representing each grid of 100×100 pixels as a single real valued number.

3.2. Controller and the evolutionary algorithm

The robot controller is composed of a continuous time recurrent neural network (CTRNN) of 25 visual input neurons, 6 inter-neurons and 7 output neurons. CTRNNs are a particular type of dynamic neural networks that have been extensively used in recent years as control systems for autonomous robots. Originally proposed in [4] as an alternative neuro-controller to classic discrete time artificial neural networks, CTRNNs are universal approximators of dynamical systems [12]. The main characteristics of this network is to provide the neural plasticity required to allow the robot to adjust its behaviour to the current operating conditions [15]. Memory and learning mechanisms are represented by the recurrent connections, as in

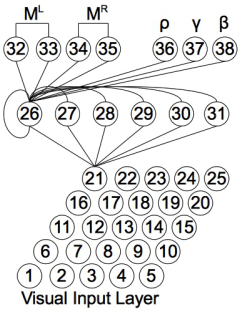


Figure 1: The neural network. The lines indicate the efferent connections for only one neuron of each layer. Each hidden neuron receives an afferent connection from each in- put neuron and from each hidden neuron, including a self-connection. Each output neuron receives an afferent connection from each hidden neuron.

classic artificial neural networks, and by the state of the network nodes which can vary overtime in response to the robot sensory experience [38]. See also [17] for an extensive discussion of the characteristics of CTRNNs as neuro-controller for autonomous agents.

The structure of the CTRNN used for this study is such that each input neuron is connected to all hidden neurons, and each hidden neuron is connected to all other hidden neurons including itself and to all output neurons (see figure 1). The states of the output neurons are used to control the speed of the left and right wheels as explained later, and to define the parameters ρ, γ and β mentioned above. The values of sensory, internal, and output neurons are updated using Equations (2), (3), and (4).

$$y_i = gI_i \ \text{ for } i \in \{1,\cdots,25\}, \quad (2)$$

$$\tau_i \dot{y}_i = -y_i + \sum_{j=1}^{31} \omega_{ji}\sigma(y_j+\mu_j) \ \text{ for } i = \{26,\cdots,31\}, \quad (3)$$

$$y_i = \sum_{j=12}^{15} \omega_{ji}\sigma(y_j+\mu_j) \ \text{ for } i = \{32,\cdots,38\}, \quad (4)$$

with $\sigma(x) = (1+e^{-x})^{-1}$. In these equations, using terms derived from an analogy with real neurons, y_i represents the cell potential, τ_i the decay constant, g is a gain factor, I_i with $i \in \{1,\cdots,25\}$ is the activation of the i^{th} sensor neuron, ω_{ji} the strength of the synaptic connection from neuron j to neuron i, μ_j the bias term, and $\sigma(y_j+\mu_j)$ the

firing rate (hereafter, η_i). All sensory neurons share the same bias (μ^I), and the same holds for all motor neurons (μ^O). τ_i and μ_i with $i = \{26, \cdots, 31\}$, μ^I, μ^O, all the network connection weights ω_{ij}, and g are genetically specified network parameters. At each time step, the output of the left motor is $M^L = \eta_{33} - \eta_{32}$, and the right motor is $M^R = \eta_{35} - \eta_{34}$, with $M_L, M_R \in [-1, 1]$. Each colour parameter is normalised in the following:

$$\rho = \frac{\eta_{36}}{\zeta}; \quad \gamma = \frac{\eta_{37}}{\zeta}; \quad \beta = \frac{\eta_{38}}{\zeta}; \tag{5}$$

with $\zeta = \eta_{36} + \eta_{37} + \eta_{38}$. Cell potentials are set to 0 when the network is initialized or reset, and Equation (3) is integrated using the forward Euler method. The integration time step is $\Delta T = 0.38$, as it takes the Pioneer robot's embedded computer 0.38 seconds to perform a complete network update cycle.

An evolutionary algorithm using linear ranking is employed to set the parameters of the networks [13]. We consider populations composed of $M = 100$ genotypes coding for the parameters of the robot controllers. At generation 0, each genotype comprising 243 real values (228 connections, 6 decay constants, 8 bias terms, and a gain factor) is chosen uniformly random from the range $[0, 1]$. Generations following the first one are produced by a combination of selection with elitism, recombination, and mutation. For each new generation, the highest scoring genotype ("the elite") from the previous generation is retained unchanged. Each of the other $M - 1$ new genotypes are generated by fitness-proportional selection from the 30 best genotypes of the old population. Each new genotype has a 0.3 probability of being created by combining the genetic material of two genotypes of the old population. During recombination, one crossover point is selected. Mutation entails that a random Gaussian offset is applied to each real-valued vector component encoded in the genotype, with a probability of 0.04. The mean of the Gaussian is 0, and its standard deviation is 0.1. During evolution, all vector component values are constrained to remain within the range $[0, 1]$. We wish to clarify that, the specific contribution of various single implementation details (e.g., structural and functional properties of the robots neuro-controllers, the proportional fitness selection algorithm to search the problem space, etc.), remain to be empirically evaluated in specifically designed future comparative studies. Within reasonable limits, we have explored various different solutions and tested different combinations of values for the large set of parameters of the system. The implementation details illustrated in this paper are those that returned the best results in terms of quality of the robot's navigation strategies and effectiveness in generating these strategies with the evolutionary algorithm. However, we cannot exclude that alternative methodological solutions would not result in the emergence of equally effective or even more robust and more adaptive strategies.

3.3. Fitness function and evaluation criteria

During evolution, each robot is evaluated on 24 trials. Each trial is a sequence of 500 time steps in each of which first the wheel speeds are computed by calculating the outputs of the network and subsequently position and orientation of the robot are updated. During evaluation, each robot experiences 12 different evolutionary scenes (see Figure 2), which differ in terms of colour characteristics of the road and non-road surfaces. In particular, scenes 1 to 6 feature only one colour component. In these scenes, either the road is darker than the non-road surface, or vice-versa. Scenes 7 to 12 feature two colour components, one which is used to represent the road, and the other to represent the non-road surface. For the third colour component, a uniformly distributed random value in the interval $[0, 255]$ is assigned to each pixel of the scene. In this way, this third colour component does not help in discriminating between the road and the non-road surface.

The scenes have average inherent contrasts between the road and non-road intensities of either 150 (scenes 1–6) or 120 (scenes 6–12) on a scale of 0 to 255. Preliminary tests showed that in scenes with a contrast below 80 and in lower contrast textures no successful controllers could be obtained. Real world environments can have contrasts much lower than this. Thus, the visual input on the real robots has been generated by applying histogram equalization to C_R, C_G and C_B in order to increase the global contrast.

The scenes have been devised in order to provide selective pressures to guide evolution towards the emergence of controllers that choose—by appropriately setting ρ, γ and β for each scene—the colour components which assist the robot in distinguishing the road from the non-road surface, and disregard those that do not show the pattern that

is being sought. With this set of scenes, the only means by which the robot can detect and successfully navigate all the roads is by varying, between trials, the colour parameters ρ, γ and β. The road is rendered using a modified version of the road generation algorithm employed in [7]. A total of 11 tiles are used each 160 cm long and 100 cm wide. The length of the road the robot needs to travel is 1760 cm.

Each road starts off with two of straight tiles followed by a smooth 30° bend left or right. This is followed by a similar smooth bend, with probability $\frac{6}{7}$ of it being in the opposite direction as the first one. This provision allows a robot to demonstrate the ability to make both kinds of turns and ensures the robot needs to be constantly maintaining its course to stay on the road. Subsequent turns are random, but checks are made to ensure no unrealistic or intersected road shapes are generated. In order to simulate roads with amorphous nondescript edges, the edges of road textures were manually faded out using noisy paintbrush tools in image manipulation software, and then alpha-blended with the underlying ground texture. During the 24 trials, each evolutionary scene is presented twice, first with a right turn followed by a left turn road, then with a left turn followed by a right turn road.

In each trial e, the robot fitness $f_e \in [1.0, 1.5]$ corresponds to the number of road tiles traversed from trial start, and the position in the last traversed tile, in case the robot does not reach the road end within 250 time steps. A trial is terminated earlier if the robot is detected to have moved off the road. The robot final fitness F is computed as:

$$F = \left(\frac{1}{E}\prod_{e=1}^{E} f_e\right)^E \times \left(\frac{1}{E}\sum_{e=1}^{e=E} C_e\right), \tag{6}$$

$$f_e = 1.0 + \frac{K+H}{22}, \quad H = \frac{Q-V}{Q}, \tag{7}$$

$$C_e = \begin{cases} \frac{1}{S}\sum_{s=50}^{s=S} |R_s - P_s^1| + |R_s - P_s^2| & 1 \le e \le 6 \\ \frac{1}{S}\sum_{s=50}^{s=S} 2 \times P_s^3 & 7 \le e \le 12 \\ 0 & \forall e, \text{if } s \le 50, \end{cases} \tag{8}$$

where $E = 24$ is the total number of trials, K the number of tiles crossed, Q the tile length, V the error vector from mid-point of the tile side closer to the road end to the robot position at the end of the trial, C_e the quality of the dynamic colour perception strategy in trial e, R_s the value of the colour parameters (i.e., either ρ, γ or β) that has to be used to discriminate between road and non-road, P_s^1 and P_s^2 the values of the colour parameters (i.e., ρ, γ or β) that do not discriminate between road and non-road in mono-colour scenes (i.e., scenes 1 to 6), P_s^3 the value of the colour parameters (i.e., either ρ, γ or β) that does not discriminate between road and non-road in dual-colour scenes (i.e., scenes 7 to 12) and S the number of time steps completed by a robot in each single trial e. C_e is computed only after time step 50, to allow the robot some time to adjust its colour parameters in an optimal way given the characteristics of the current scene. Robots leaving the road before the 50^{th} time step get a 0 fitness for that trial irrespective of the distance travelled since trial start. H is set to zero if the robot crosses the road/non-road border. The term η is introduced to guide evolution towards the emergence of solutions which vary in an adaptive way the activation of the colour parameters. We observed that without the term η the fitness function is not capable to steer evolution towards the emergence of the adaptive mechanisms required to visually discriminate the roads in all the evolutionary scenes.

4. Results

We ran a set of 15 differently seeded evolutionary simulations (or evolutionary runs), each of which lasted 4,500 generations. At the end of the evolutionary phase, we performed a first post-evaluation test in simulation. The aim of this post-evaluation test is to generate a more accurate estimate of the effectiveness of the most promising solutions in a larger set of operating conditions, and to choose what will be our best evolved controller. For each evolutionary run, we post-evaluated the best solution (i.e., the best genotype based on fitness) of each of the last 2,500 generations. Each solution has been re-evaluated on a set of 192 trials based on the 12 evolutionary scenes described above. Each scene has been presented 16 times by varying the shape of the road—we use 8 different road shapes—as well as the inherent contrast and distributions of intensities in the three RGB colour channels—for each scene, we use two different intensity distributions. The results of this first post-evaluation tests are shown in Figure 3, which shows the percentage of success of the best

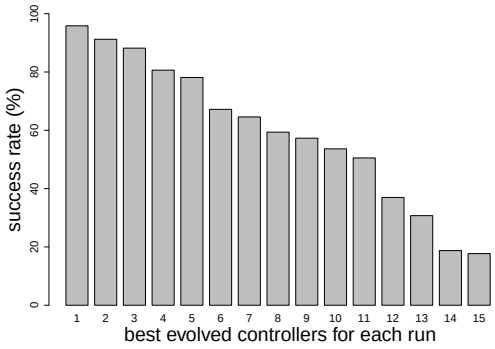
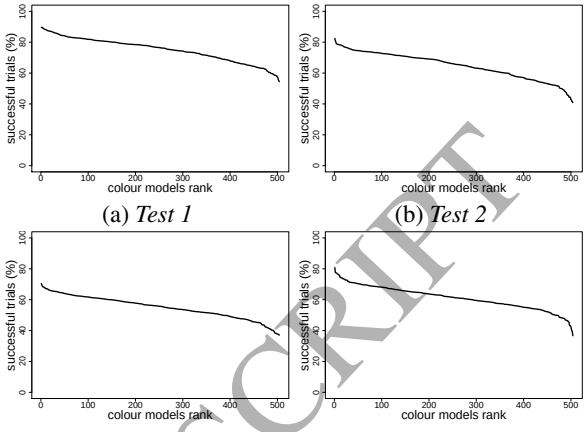


Figure 3: Percentage of success of the best evolved controllers of each evolutionary run, during 192 trials (12 scenes, 8 road shapes, 2 distributions of colour intensity). Controllers are ranked from best to worst.



(a) *Test 1* (b) *Test 2*

(c) *Test 3* (d) *Test 4*

Figure 4: Results of *Test 1*, *Test 2*, *Test 3*,*Test 4*. Each graph shows the percentage of successful trials for all 504 colour models. In each graph, the colour models are ranked from the best to the worst.

evolved controllers of each evolutionary run, during the 192 post-evaluation trials[1]. The graph shows that 4 out of 15 evolutionary runs managed to produce controllers with a success rate higher than 80%. The controller that drove the most successful robot (i.e., the one with highest number of roads navigated from start to end without leaving the road) has been chosen to be our best evolved controller.

In the following section, we show the results of a series of analyses in which the best evolved controller (hereafter, B-controller) has been tested on a larger set of simulated operating conditions generated by varying the characteristics of the scenes as well as the nature of the colour components used to generate the controller sensory input. These series of tests aim to verify the robustness and adaptability of our B-controller. In Section 4.2, we test the capability of the B-controller to cross the reality gap by showing the results of a series of tests in which the controller guides a real Pioneer robot in 5 different outdoor environments. In Section 4.3, we show the results of a series of qualitative and quantitative tests aimed at shedding light on the mechanisms underpinning the robot capability to cope with various forms of environmental variability.

4.1. Evaluation of the B-controller in simulated scenes

We illustrate here the results of five different tests that feature various forms of colour variability between and within the scenes. In *Test 1* to *Test 4*, the B-controller is evaluated under conditions in which the sensory inputs I_i (Equation 1) are computed by considering all the possible 3-permutations of the following 9 colour components: R, G, B, H, S, Y, U, a, and b (respectively from the RGB, HSV, YUV and CIE L*a*b* colour spaces). Hereafter, each arbitrary triplet of colour components is referred to as a colour model. Triplets made of same colour components, but ordered differently, differ in the way in which the colour components are associated to the colour parameters ρ, γ and β. The logic behind the use of multiple colour models is the following: a successful controller that adaptively varies the colour parameter ρ, γ and β may benefit from the use of any arbitrary combination of three components chosen from different colour models. For example, with the use of components of colour models different from RGB in which colour and luminance are not intertwined, a successful adaptive controller may more effectively respond to extra environmental variability, or dy-

[1] Videos of the robot controlled by the best evolved controller are available at `https://www.aber.ac.uk/en/cs/research/ir/dss/#road-driving`.

namic variations produced by shadows, sudden changes in luminance, etc.[2]

Test 1 aims to evaluate the B-controller on a large set of scenes (812) generated by the 2-permutations of 29 different textures. In each scene, one texture is used for the road and the other for the non-road surface. Given that each scene is presented with 8 different road shapes, and associated to the 504 colour models given by permuting the 9 colour components R, G, B, H, S, Y, U, a, and b, the total number of trials is $812 \times 8 \times 504$. The purpose of *Test 2*, *Test 3*, and *Test 4* is to evaluate the performance of the B-controller in scenes in which there is more variability in term of colour distribution compared to scenes presented during the training phase, since each scene features three (instead of two) different textures. In *Test 2*, the texture of the two non-road areas, on the left and on the right of the road, are different. In *Test 3*, left and right non-road areas are the same, but the non-road texture on both sides of the road changes colour half-way through the road. In *Test 4*, left and right non-road areas are the same, and the road texture changes colour halfway through the road. We created 720 scenes, resulting from all possible 3-permutations of 10 different textures. Each scene is presented with 8 different road shapes, and it is repeated 504 times, one for each colour model given by permuting the 9 colour components R, G, B, H, S, Y, U, a, and b. It follows that the total number of trials in *Test 2*, *Test 3*, and *Test 4* is $720 \times 8 \times 504$.

As during evolution, in all tests, the robot controller is reset at the beginning of each trial. A trial is considered successful if the robot manages to traverse all 11 tiles forming the road. A trial is terminated earlier and set as unsuccessful if the robot is detected to have moved off the road. The results of *Test 1* to *Test 4* are shown in Figure 4, where each graph indicates the percentage of successful trials of all 504 colour models. We also compare the colour models by considering these four tests as a multi-objective optimization problem, in which each model has to maximize its performance measured in terms of percentage of successful trials. Table 1 shows, for each test, the performance of the robot in trials in which the controller is linked to those colour models populating the Pareto set, plus the RGB colour model. For each test, the colour models are ranked in descending order of performance. There are various lessons to learn from this data.

We consider first *Test 1* and *Test 2* where the colour variability is only between the scenes and not within each scene. Both tests evaluate the robustness of the B-controller to deal with a larger number of road/non-road textures combinations never encountered during training. From Table 1, we notice that in *Test 1* the success rate of the robot with almost all Pareto set colour models is higher than 85%, while in *Test 2* the robot success rate with at least half of the Pareto set colour models is higher than 75%. These results indicate that, if supported by adequate colour models (i.e., models which can potentially generate the perceptual cues required to visually discriminate the road from the non-road surface in such variable operating conditions), the B-controller manages to successfully cope with a large range of different scenes, proving to possess the required robustness to deal with scenes never encountered during training. Although the data gathered from these tests do not tell us anything about how ρ, γ and β are varied between the scenes, the relatively high success rates under these testing conditions suggest that the controller copes with the colour differences of the scenes by adaptively varying the colour parameters ρ, γ and β to exploit the benefits of those components of each colour model that facilitate the navigation task. The results of *Test 1* and *Test 2* also show that not all colour models adequately support the controller in the navigation task. The graphs in Figures 4(a) and 4(b) show that for many colour models the robot success rate drops below 70%. This suggests that the adaptiveness required to cope with the environmental variability encountered by the robot in these tests is a combination of the functional properties of the controller and the characteristics of the colour model. Each colour model offers a unique perspective on the scenes from which the controller has to extract the required perceptual features to guide the robot in this navigation task. Quite surprisingly, RGB, which is the

[2]During evolution, the RGB colour model has been used to create the scenes and also to generate the input vector of the robot controllers. The reason for this is, first, that it is relatively simple altering R, G, B of an image texture rather than components of different colour models to generate the environmental variability required to select controllers capable of adaptively tuning their perception system (i.e., the ρ, γ and β) to different scenes. Second, it would have required longer in term of computational time, to evaluate controllers on the use of different components of multiple colour models.

model used during the evolutionary phase, is not among the models that allow the robot to perform reasonable well at both tests. As expected, the performances observed in *Test 2* are worse than those observed in *Test 1*. Indeed, *Test 2* uses scenes where right and left non-road surfaces have a different colour, a situation not encountered during training.

While *Test 1* and *Test 2* feature variability between the scenes, *Test 3*, and *Test 4* are definitely more challenging since they feature colour variability between and within the scenes. Recall that the controller is reset only at the beginning of each trial. Thus, in order to cope with variability within a scene, the controller has to adjust the colour parameters without being reset at the time when, during the trial, the environmental conditions change. The results of these tests show some positive aspects, and areas in which improvement is needed. The positive data are those of *Test 4*, where the robot's success rate with several Pareto set colour models is higher than 75% (see Figure 4(d) and Table 1). The B-controller can successfully cope with conditions in which the road texture abruptly changes, in spite of the fact that this event has not been contemplated during training. In *Test 3* the success rate is not as high as for the other previous tests. The change in colour texture of the non-road area seems to hinder the performance of the robot more than any other type of environmental variability (Figure 4(c)).

We also run a fifth test (*Test 5*) to evaluate the B-controller in scenes with shadows and bright spots. To keep the computational time required to run this test within reasonable limits, we used a limited set of 25 different scenes. In each scene, the colour of the non-road and of the road surfaces does not change but shadows and bright spots are added to the scenes (see supplementary document for images of these scenes). As in previous tests, each scene is presented with 8 different road shapes. However, each scene is repeated for only the 15 colour models populating the Pareto set as resulted from *Test 1* to *Test 4*, plus the RGB colour model. The total number of trials in *Test 5* is therefore $25 \times 8 \times 16$. The results are shown in Table 1. We can see that, even in these very challenging conditions, the robot's success rate with at least two Pareto set colour models (i.e., bUV and bHV) is higher than 75%.

In summary, given that the large majority of the scenes employed in these tests have been created with textures not used during training, the relatively good performances of the robot guided by the B-controller associated to several colour models bear upon the robustness of the controller. Moreover, these results demonstrate that the design methods is particularly effective in generating adaptive mechanisms for solving this visual navigation task. The next interesting point emerging from the data is that there is no single colour model that performs better than all other models in all tests. The final point to highlight is that the RGB model is not the best performing colour model in any of the tests (see Table 1).

4.2. Outdoor trials with the real robot

We illustrate the results of tests in five different outdoor environments with the real Pioneer robot guided by the B-controller. The robot is tested in 5 different environments (see Figure 5 for details). For each environment, the robot undergoes a total of 30 trials, as 3 sets of 10 trials with 3 different colour models (i.e., USH, aSH, bUV). We used these three colour models because, in previous tests, they proved to be able to adequately support the robot guided by the B-controller in a variety of simulated environmental conditions.

For each set of 10 trials, at the sixth trial the robot's starting position changes from the beginning to the end of the outdoor path, and consequently its direction of motion is inverted. A trial is successfully terminated when the robot traverses the entire length of the road without moving off the road boundaries. A trial is unsuccessfully terminated when either one set of wheels goes off the road boundaries, or for exceeding the time limits set to 5 minutes. Since Path 1, Path 2 and Path 3 are sections of a longer public path, they all shared the same road surface but they differ in terms of non-road surfaces on either side (see Figures 5(a), 5(b) and 5(c)). Extra variability not only between environments but also between trials in the same environment has been created by the varying weather and lighting conditions encountered during evaluation. Moreover, as all environments are public areas, pedestrians walking on the paths while trials were in progress represented a further challenge the robot had to deal with. The trials were not stopped for pedestrians walking in front of, or around the robot. These real environments are obviously more complex than those recreated in simulation and used for training, not only for the

presence of pedestrians on the road surface, but also in terms of the colour distribution, varying lighting conditions, and for the presence of visual features such bushes and benches. Moreover, the width of the road of these outdoor environments is slightly wider than the surfaces of road in the simulated environments. The ability of the B-controller to deal with all these new elements should be considered further testament to its robustness.

Table 2: Number of successful trials per colour model for each outdoor environment (column 3); mean and standard deviation of the robot's divergence from the centre of the road (columns 4 and 5); mean and standard deviation of the time taken to complete a trial (columns 6 and 7).

Env.	colour Model	Num. Succ.	Divergence (cm)		Time (s)	
			mean	sd	mean	sd
Path 1	USH	8/10	-9.4	39.1	168	66.1
	aSH	4/10	-19.9	39.2	186	56.1
	bUV	8/10	-43.3	74.6	190.5	15.8
Path 2	USH	8/10	-14.3	38.3	137.2	22.3
	aSH	8/10	-18.9	42.7	168	45.0
	bUV	5/10	10.3	61.6	223.5	18.6
Path 3	USH	8/10	4.4	41.1	127.8	33.6
	aSH	8/10	-8.5	52.5	152.4	21.9
	bUV	10/10	7.2	33.9	231	25.2
Path 4	USH	9/10	-6.0	37.7	170	21.5
	aSH	4/10	3.4	34.2	136.5	22.5
	bUV	10/10	-4.2	22.5	165.5	30.4
Path 5	USH	9/10	-10.5	108.3	102	17.5
	aSH	5/10	-59.6	49.3	147.6	37.6
	bUV	10/10	-14.0	68.5	226.8	54.3

The results of the outdoor tests are shown in Table 2. Considering all 150 trials across the three colour models, the overall success rate is 76%. We consider this as a good result that demonstrates the effectiveness of the evolutionary method in synthesizing a robust robot controller capable of successfully guiding a real robot engaged in this visual navigation task. Moreover, these results demonstrate the effectiveness of the embodied active approach, which generated a dynamic perception and action system capable of coping with the large environmental variability characteristic of the outdoor testing conditions.

Among the three colour models, bUV is the most successful one with an average success rate of 86%. However, bUV has a low success rate (50%) in Path 2. For all trials in which the B-controller has been associated to bUV, the robot systematically fails Path 2 for one of the two directions of motion, around the path half-way point which corresponds to a texture change on one of the non-road surfaces. When tested with the colour model USH, the robot achieves a slightly lower average success rate of 84%, but with a more constant performance across all five environments. The worst robot performance is with the colour model aSH with an average success rate of 58%.

It should be noted that during the evolutionary design phase, no particular restrictions have been imposed either to the trajectory or to the speed of motion, apart from the requirements of navigating the paths without straying from the road boundaries, and of reaching the end within a given time limit (450 update cycles or 171 s). This latter requirement creates an implicit selective pressure favouring controllers that guide the robot at a speed sufficiently high to cover the 17.6 m of the simulated roads in less than 171 s.

During the outdoor trials, we measured mean and standard deviation of the robot's divergence from the centre of the road. From the divergence values shown in Table 2, we notice that the robot controller associated with the colour model USH generates trajectories that are closer to the centre of the road than for the other two colour models. The use of USH helps the robot to develop a navigation strategy that tries to remain as far away as possible form the road edges. However, the relatively high standard deviation of the divergence for almost all colour models indicates that the trajectories have been rather sinuous, with quite a few changes in direction of motion to avoid to cross the road edges. For trials on Path 5 for all colour models, the robot tended to remain on the darkest lane (see Figure 5(e)). However, with the colour models USH and bUV, and only for one travel direction, the robot mainly uses the brightest lane (see in Figure 5(e)), and it switches between lanes towards the end of the path. This accounts for the high standard deviation for the divergence from the road centre.

During the tests with real robots, no trial failed because of exceeding the time limit of 5 minutes. However, if we look at the average trial completion time (see Table 2, column 6) we notice that trials carried out with the bUV colour model took generally longer than trials with the two other colour models. We observed large variations in trial completion time between trials with any given colour model (see Table 2, column 7). This is caused by the be-

haviour of the robot which, in some trials, reduces its speed in certain sections of the path and displayed periods of rapid changes in direction without much progression along the path.

4.3. Analysis of the mechanisms underpinning adaptivity and robustness to environmental variability

We have shown that the B-controller manages to successfully guide simulated and real robots in a variety of environments which differ in term of the colour of the road/non-road surfaces. We showed that the controllers are robust enough to deal with environmental conditions never encountered during training. From the robot performance and the kind of variability faced during post-evaluation, we deduced that the controller is able to tune the colour parameters ρ, γ and β in order to adjust the visual input to the colour characteristics of the environment in which the robot operates. In this section, we provide evidence of how the colour parameters are actually varied by the B-controller.

We begin our investigations by showing the results of a qualitative test which visualizes the variation over time of ρ, γ and β while the simulated robot navigates four different environments (i.e., four trials) in which the colour of the road surface changes half-way through the road. The simulated robot is controlled by the B-controller linked to the RGB colour model. The results are shown in Figure 6, where each column refers to a different trial. Different combinations of colours are used for the road in each trial (Figure 6, first row).

The graphs in the top row of Figure 6 are a representation of the visual scenes experienced by the robot. The graphs are constructed by taking into account, at each update cycle of the robot controller, the contribution of C_R, C_G, and C_B (see Section 3.1 for details) for each of the 25 grid cells superimposed on the camera image. At each update cycle, the 25 coloured points are distributed over the y-axis. The graphs in the second row from the top refer to the variation over time of the 25 values sensory input vector (i.e., I_i, see Equation (1)). At each update cycle, the 25 grey scale points corresponding to the vector I_i are distributed over the y-axis. The graphs in the third row from the top of Figure 6 refer to the variation over time of the colour parameters ρ (light grey), γ (dark grey) and β (black). The shades of grey indicate the activation of each colour parameter at each update cycle. The graphs in the fourth row from the top of Figure 6 refer to the output to the left and right motors (M^L and M^R respectively).

Looking at the variation of the colour parameters over time, we first notice that different types of activation patterns are used (e.g., see Figure 6, third row), and that the type of pattern tends to change when the colour characteristics of the road surface changes (e.g., see Figure 6(b), at about 60 s). We also notice that the controller employs three different types of activation patterns. We can observe an oscillatory pattern in which the controller alternates between high and low activation of all colour parameters. This can be seen in Figure 6(a) third row, where in the first half of the trial there is a prevalence of ρ (light grey), while in the second part of the trial there is a prevalence of γ (dark grey), with spikes of β (black) appearing throughout the trial. We can also observe an oscillatory pattern in which the controller mainly employs two colour parameters, and oscillations tend to have a lower temporal frequency than the previous oscillatory pattern, for example in the first part of the trial in Figure 6(b) third row. Finally, we can observe a pattern in which one colour parameter is set to its highest values while the other two are set to zero, Such as in the second part of the trial in Figure 6(b) third row, and in trials in Figures 6(c) and 6(d) third row, where β dominates over ρ and γ.

These three activation patterns of the colour parameters are quite prototypical, since they exhaustively represent all the possible types of activation patterns observed in different environmental conditions, and with the controller linked to different colour models. By looking at the variations of the colour parameters in a larger set of simulation trials extracted from *Test 1*, *Test 2*, *Test 3*, *Test 4* and *Test 5*, described in Section 4.1, we noticed that changes in activation patterns of the colour parameters are triggered by different types of environmental variability (e.g., changes in the colour of the non-road surfaces, or appearance of bright spots and shadows), and that the oscillatory patterns are the most frequent ones. We believe that the oscillatory patterns of the colour parameters are a rather effective exploratory strategy that allow the controller to tune the vision system to the characteristics of the environment, both during the initial phase of the trial, when the environment is unknown to the robot, and in response to environmental variations that require a change in ac-

tivation pattern. Future work is required to test this hypothesis. On the basis of the activation patterns observed in Figure 6 we can say that a dynamic change in the environment tends to result into a change in the nature of temporal activation of the colour parameters exhibited by the controller. If the controller cannot continue to extract a satisfactory final input vector after being exposed to a change in the environment, it changes the pattern of activation of its colour parameters to maintain the robot navigation capabilities and to stay within the road boundaries.

Figure 6 also shows an interesting relationship between the perception and the motor system of the B-controller. The fourth row of Figure 6 shows that at the points where the road colour changes there is also either a marked change (see Figure 6(b) and 6(c) fourth row) or a temporary disruption (see Figure 6(a) and 6(d) fourth row) of the nature of motor output activation. Trials with extended periods of colour, and hence motor output, oscillatory patterns take significantly longer to complete. Trials in Figures 6(c) and 6(d) where there are only short periods of oscillatory behaviour last 48 s and 45 s, respectively. This is in contrast to the trial in Figure 6(a) where the robot takes 698 s to cover the same distance (21 m). In the trial in Figure 6(b) the robot moves much slower in the first section of the road where the colour parameters oscillate, compared to the second half of the trial where it covers the half-trial distance in approximately 20 s. This indicates that the controller's colour perception and motor system are tightly coupled.

In the remaining of this section, we show the results of a further analysis that aims to quantitatively verify the hypothesis that the controller adapts to different environments by varying the activation of the colour parameters ρ, γ and β. For this we consider all 50 outdoor trials carried out with the USH colour model, across all 5 environments shown in Figure 5. In each of these trials we recorded the values of the colour parameters during the entire trial with the exclusion of the first 10 iterations, where colour parameters are not set properly yet, and the last 10 iterations, where colour parameters tend to be affected by the perception of the road end. Given that the previous qualitative tests with simulated trials indicated that the B-controller tends to develop oscillatory patterns in which each colour parameter is either close to its maximum (1) or to its minimum (0), we classify the raw data as

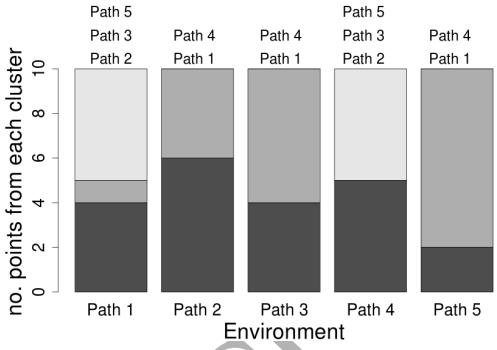


Figure 7: Distribution of elements of the 6-dimensional space among the three clusters for each outdoor environment. Black shaded parts of the bars correspond to points belonging to cluster one, dark grey shaded parts correspond to points belonging to cluster two, and light grey shaded parts correspond to points belonging to cluster three. Annotations above each bar refer to environments where the points distribution on clusters is significantly different (based on Fisher's exact test at 95% confidence level) to the distribution of the environment indicated on the x-axis.

follows. We compute the percentage of time each colour parameter is either below a low threshold of 0.2, or above a high threshold of 0.8. This reduces the raw data gathered from each trial to a point in a 6-dimensional vector space, where the six dimensions are given by the two types of categories (values either below the low threshold or above the high threshold), times the three colour parameters ρ, γ and β. We then clustered the 50 points obtained from all outdoor trials using the mean-shift unsupervised clustering algorithm (see [6]). The algorithm clustered the data into three clusters (see supplementary document available at `https://www.aber.ac.uk/en/cs/research/ir/dss/#road-driving` for a detailed description of the three clusters and for a graphical representation of how the 50 points are distributed in the 6-dimensional space, and how they are clustered). We observed that clusters one and three correspond to phases of trial where all three colour parameters are activated in an oscillatory manner, while cluster two corresponds to phases of trial where the controller primarily keeps γ set to 1.

Figure 7 shows the distribution of the points in the 6-dimensional space among the three clusters for each outdoor environment. The graphs clearly indicate that the B-controller tends to use different activation patterns in

different environments. For example, while in Path 1and in Path 4 the activation patterns corresponding to clusters one and three are more represented than the activation pattern corresponding to cluster two. In Path 2, Path 3 and Path 5 the activation patterns corresponding to clusters two and three are more represented than the activation pattern corresponding to cluster one. Considering the null hypothesis that the distribution of the elements among the clusters is independent of the environments, we carried out the Pearson's chi-squared test of independence. The test gives us a $\tilde{\chi}^2$ value of 56.44, which is higher than the critical value for significance level 0.005 (26.75). This leads us to reject the null-hypothesis and to reach the conclusion that the dynamic colour-perception strategy used by the B-controller depends on the properties of the environment in which the robot is required to operate. Furthermore, we carried out the Fisher's exact test to statistically verify in which environment the controller displays a different activation pattern of the colour parameters. The test indicates that the distribution of elements in Path 1 and in Path 4 is statistically different from the distribution of elements recorded in Path 2, Path 3 and Path 5 (see Figure 7). We conclude that the B-controller has different strategies of dynamically activating the colour parameters depending on the nature of the outdoor environment, and it adapts its strategy to suit the environment it is operating in.

5. Conclusions

In this work we tackled the challenge of designing controllers for autonomous vehicles and showed the potential benefits of a design method based on dynamic neural networks synthesized by evolutionary computation techniques. We targeted the problem of visual navigation on unmarked roads. By assuming the existence of a colour difference between the road and the non-road areas, we developed a system to design controllers that allow autonomous vehicles to navigate unmarked roads by exploiting this colour difference. We employed artificial neural network based controllers owing to their potential robustness and adaptability to successfully cope with conditions never encountered during training. Visual discrimination tasks requiring colour perception are generally tackled using sensory apparatuses (e.g., a camera) which tend to generate high-dimensional input vectors. With artificial neural networks, high-dimensional input vectors significantly contribute to increasing the dimensionality of the parameter search space, with potentially severe consequences on the effectiveness of both supervised and unsupervised training methods to generate successful solutions. We have kept the network parameters search space within reasonable limits by reducing the camera pixels density with a dimensionality reduction process that interfaces the robot camera with the artificial neural network controller (see Section 3.1). The integrated action-perception approach effectively compensates for the low resolution perceptual system and it allows the robot sensory apparatus to be tuned to the colour characteristics of the environment.

We have described a method that, first, allows to generate robot controllers that can drive autonomous vehicles on unmarked roads by distinguishing the road from the non-road area based on colour differences between the two areas. Second, this method allows the vehicle to autonomously adapt to the variability in colour of the road and non-road areas by generating adaptive mechanisms capable of tuning the robot visual perception system to the characteristics of the environment in which it is required to operate. This is a significant contribution of this study which provides an alternative to the solution described in [30], concerning the challenges faced by vision based navigation systems to cope with the enormous variability of the real world conditions. We have tested our B-controller in a limited set of real world scenes. However, the promising results described in this paper suggest that this approach can potentially be a valuable option to design control systems for autonomous driverless vehicles required to operate in more challenging conditions.

Another contribution of this work is in illustrating the evolutionary conditions, the functional and structural properties of the network, and the fitness function used to generate controllers that, when ported to a real robot, proved to be successful and robust enough to deal with complex outdoor scenes. We have extensively observed an analysed the mechanisms underpinning the capability of the B-controller to adjust to the colour characteristics of different environments. We showed that the patterns of activation of the colour parameters ρ, γ and β vary in response to different types of colour changes in the road/non-road areas, both in simulated and real sce-

narios. We have observed that, although different types of patterns can be generated, the oscillatory patterns are the most frequent ones. We have also run some tests that indicate that the colour parameter activation patterns generated by the networks are absolutely crucial for the functional integrity of the system. In particular, we carried out a variant of *Test 1* (Section 4.1), with the RGB colour model. At each time step of each trial, we substituted the values of ρ, γ and β generated by controller with other values in which the colour parameter associated to the colour component (i.e., R, G, or B) carrying the highest contrast level between road and non-road areas is set to one, and the other colour parameters are set to zero. In this way, the task of the robot should have been facilitated since, in all trials, the vision system was set to maximize the road/non-road contrast. Under this condition, the robot could only successfully terminate 41.50% of the trials, as opposed to 57% when the robot perception system is not altered. This clearly indicates that, although complex activation patterns may appear superfluous, they are indeed playing an invaluable role for the functional integrity of the controller.

There are other methods in the literature that can be used to interface artificial neural network controllers for guiding autonomous vehicles required to solve colour discrimination tasks in complex real world conditions. As mentioned in Section 2 however, these methods either rely on multiple neural networks "glued" together, each trained to specific environmental conditions, or more complex convolution neural networks. We have shown here that a single neural network designed with evolutionary computation techniques using carefully crafted simulated data can cope with the relatively large environmental variability. Convolution neural networks, as mentioned in Section 2, are a promising method that do not require any pre-processing of the camera images to interface the visual input with the neuro-controller. We think that it would be important to dedicate future work to comparative studies in which different approaches are evaluated on identical scenarios, to effectively identify advantages and disadvantages of each of them. We are also planning to extend our colour based visual perceptual system to more complex scenarios in which discrimination capabilities are not only limited to the distinction between road and non-road areas, but also include the recognition and avoidance of static and moving obstacles present on the road, such as other vehicles, or pedestrians. Finally it would be interesting to study scenarios featuring multiple decision points such as junctions, with or without traffic lights, in which the vehicles have to autonomously decided which action to take.

Acknowledgement

This work is partly sponsored by Fujitsu and HPC-Wales.

References

[1] Alvarez, J., Gevers, T., LeCun, Y., Lopez, A.: Road scene segmentation from a single image. In: A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, C. Schmid (eds.) Proc. of the European Conf. on Computer Vision, pp. 376–389. Springer Berlin Heidelberg (2012)

[2] Álvarez, J., López, A.: Road detection based on illuminant invariance. IEEE Transactions on Intelligent Transportation Systems **12**(1), 184–193 (2011)

[3] Badrinarayanan, V., Kendall, A., Cipolla, R.: SegNet: A deep convolutional encoder-decoder architecture for image segmentation. arXiv preprint arXiv:1511.00561 (2015)

[4] Beer, R.D., Gallagher, J.C.: Evolving dynamic neural networks for adaptive behavior. Adaptive Behavior **1**(1), 91–122 (1992)

[5] Chen, C., Seff, A., Kornhauser, A., Xiao, J.: DeepDriving: Learning affordance for direct perception in autonomous driving. In: Proc. of the IEEE Int. Conf. on Computer Vision (ICCV), pp. 2722–2730 (2015)

[6] Cheng, Y.: Mean shift, mode seeking, and clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence **17**(8), 790–799 (1995). DOI 10.1109/34.400568

[7] Clarke, S., Labrosse, F., Trianni, V., Tuci, E.: An evolutionary approach to road following: a simulated case study. In: Proc. of the 12^{th} European Conf. on Artificial Life (ECAL), pp. 1017–1024. MIT Press (2013)

[8] de Croon, G.: Adaptive active vision. Ph.D. thesis, Maastricht University (2008)

[9] Dickmanns, E., Mysliwetz, B.: Recursive 3-D road and relative ego-state recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **14**(2), 199–213 (1992)

[10] Dudek, G., Jenkin, M.: Computational Principles of Mobile Robotics. Cambridge University Press (2000)

[11] Floreano, D., Kato, T., Marocco, D., Sauser, E.: Coevolution of active vision and feature selection. Biological Cybernetics **90**(3), 218–228 (2004)

[12] Funahashi, K., Nakamura, Y.: Approximation of Dynamical Systems by Continuous Time Recurrent Neural Networks. Neural Networks **6**, 801–806 (1993)

[13] Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, Reading, MA (1989)

[14] Hadsell, R., Sermanet, P., Ben, J., Erkan, A., Scoffier, M., Kavukcuoglu, K., Muller, U., LeCun, Y.: Learning long-range vision for autonomous off-road driving. Journal of Field Robotics **26**(2), 120–144 (2009)

[15] Harvey, I., Di Paolo, E., Wood, R., Quinn, M., Tuci, E.: Evolutionary robotics: A new scientific tool for studying cognition. Artificial Life **11**(2), 79–98 (2005)

[16] Harvey, I., Husbands, P., Cliff, D.: Seeing the light: artificial evolution, real vision. In: D. Cliff, P. Husbands, J. Meyer, S. Wilson (eds.) Proc. of 3^{rd} Int. Conf. on Simulation of Adaptive Behavior (SAB), pp. 392–401. MIT Press/Bradford Books, Boston MA (1994)

[17] Harvey, I., Husbands, P., Cliff, D., Thompson, A., Jakobi, N.: Evolutionary robotics: the sussex approach. Robotics and Autonomous Systems **20**(2–4), 205–224 (1997)

[18] Held, R., Hein, A.: Movement-produced stimulation in the development of visually guided behavior. Journal of Comparative and Physiological Psychology **56**(5), 872–876 (1963)

[19] Hsu, C.M., Lian, F.L., Lin, Y.C., Huang, C.M., Chang, Y.S.: Road detection based on region similarity analysis. In: Proc. of the Int. Conf. on Automatic Control and Artificial Intelligence, pp. 1775–1778 (2012)

[20] Jochem, T.M., Pomerleau, D.A., Thorpe, C.E.: MANIAC: A next generation neurally based autonomous road follower. In: Proc. of the Int. Conf. on Intelligent Autonomous Systems (1993)

[21] Kato, T., Floreano, D.: An evolutionary active-vision system. In: Proc. of the Congress on Evolutionary Computation (CEC), vol. 1, pp. 107–114. IEEE (2001)

[22] Kluge, K., Thorpe, C.: The YARF system for vision-based road following. Mathematical and Computer Modelling **22**(4-7), 213–233 (1995)

[23] LeCun, Y., Bengio, Y.: Convolutional networks for images, speech, and time series. In: M. Arbib (ed.) The handbook of brain theory and neural networks, vol. 3361, pp. 255–258. MIT Press (1995)

[24] LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)

[25] Liu, Z., Wang, Y.: Deeper direct perception in autonomous driving. Technical report (2016)

[26] Manz, M., von Hundelshausen, F., Wuensche, H.: A hybrid estimation approach for autonomous dirt road following using multiple clothoid segments. In: Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 2410–2415 (2010)

[27] Narayan, A., Tuci, E., Labrosse, F., Alkilabi, M.: Road detection using convolutional neural networks.

In: Proc. of the 14^{th} European Conference on Artificial Life (ECAL), pp. 314–321 (2017)

[28] Nöe, A.: Action in Perception. MIT Press (2006)

[29] Oliva, A., Torralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. International Journal of Computer Vision **42**(3), 145–175 (2001)

[30] Ososinski, M., Labrosse, F.: Automatic driving on ill-defined roads: An adaptive, shape-constrained, color-based method. Journal of Field Robotics **32**(4), 504–533 (2015)

[31] Piekniewski, F., Laurent, P., Petre, C., Richert, M., Fisher, D., Hylton, T.: Unsupervised learning from continuous video in a scalable predictive recurrent network. arXiv preprint arXiv:1607.06854 (2016)

[32] Pomerleau, D.: Progress in neural network-based vision for autonomous robot driving. In: Proc. of the Intelligent Vehicles '92 Symposium, pp. 391–396 (1992)

[33] S. Thrun et al.: Stanley: The robot that won the DARPA grand challenge. Journal of Field Robotics **23**(9), 661–692 (2006)

[34] Shinzato, P.Y., Wolf, D.F.: A road following approach using artificial neural networks combinations. Journal of Intelligent & Robotic Systems **62**(3), 527–546 (2011)

[35] Suzuki, M., Floreano, D., Paolo, D., Ezequiel, A.: The contribution of active body movement to visual development in evolutionary robots. Neural Networks **18**(5-6), 656–665 (2005)

[36] Taniguchi, E., Thompson, R.G., Gevaers, R., Van de Voorde, E., Vanelslander, T.: Cost Modelling and Simulation of Last-mile Characteristics in an Innovative B2C Supply Chain Environment with Implications on Urban Areas and Cities. Procedia - Social and Behavioral Sciences **125**, 398 – 411 (2014)

[37] Torralba, A., Efros, A.: Unbiased look at dataset bias. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 1521–1528 (2011)

[38] Tuci, E., Quinn, M., Harvey, I.: An evolutionary ecological approach to the study of learning behaviour using a robot based model. Adaptive Behavior **10**(3/4), 201–221 (2003)

[39] Tudoran, C.T., Neagoe, V.E.: A new neural network approach for visual autonomous road following. Latest Trends on Computers **1**, 266–271 (2010)

[40] Valada, A., Oliveira, G., Brox, T., Burgard, W.: Deep multispectral semantic scene understanding of forested environments using multimodal fusion. In: Proc. of the Int. Symposium on Experimental Robotics, pp. 465–477 (2016)

[41] Yeo, Y., Xiao, X., Zhang, X.: Rural scene parsing and road boundary estimation by fusion of lidar pointcloud and eo images. In: Proc. of the Int. Conf. on Information Fusion (FUSION (2015)

[42] Yuan, Y., Jiang, Z., Wang, Q.: Video-based road detection via online structural learning. Neurocomputing **168**, 336–347 (2015)

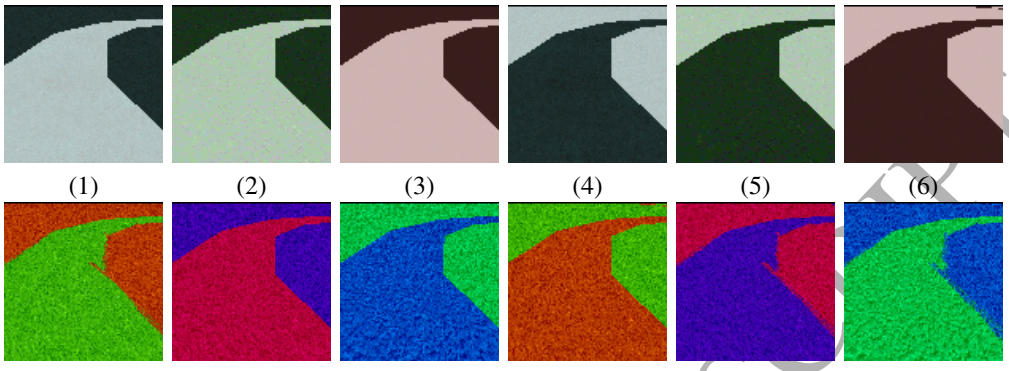


Figure 2: Snapshots of the 12 evolutionary scenes used to train the robot's neuro-controller, see also Section 3.3 for further details.

Table 1: Performance of the colour models in the Pareto set, and RGB, for *Test 1*, *Test 2*, *Test 3*, *Test 4* and *Test 5*. For each test, the colour models are ranked in descending order of performance.

Test 1		*Test 2*		*Test 3*		*Test 4*		*Test 5*	
colour model	**Success rate (%)**	**colour model**	**Success rate (%)**	**colour model**	**Success rate (%)**	**colour model**	**Success rate (%)**	**colour model**	**Success rate (%)**
VHb	89.63	bSH	82.41	bUV	70.45	bUa	80.76	bUV	75.25
VHa	89.51	aSH	80.97	UbV	69.56	Uba	77.51	bHV	75.25
bHU	89.08	HbB	79.06	aSH	68.83	Bha	77.50	bHa	73.25
bHA	88.59	USH	78.07	VHb	68.35	VHa	77.36	VHa	73.25
aSH	88.51	bHV	76.87	UbB	68.26	UbV	77.18	VHb	71.25
aHB	88.40	UbB	76.33	HbB	66.54	bUV	76.82	aHb	69.75
bHV	88.06	bHa	76.07	aHB	65.72	aHb	76.49	bUa	68.25
bUA	87.83	bHU	74.53	USH	65.71	VHb	75.88	UbV	67.00
UbV	87.74	VHb	74.44	bHV	65.67	UbB	74.16	Uba	67.00
bUV	87.74	bUV	73.26	Uba	63.26	bHV	72.74	bHU	63.75
Uba	87.11	bUa	73.26	bHU	62.46	HbB	71.47	USH	63.50
USH	86.79	aHb	72.72	bUa	61.85	bHU	71.23	aSH	62.00
UbB	85.79	VHa	72.43	bSH	59.37	aSH	70.12	bSH	60.25
HbB	81.86	UbV	70.67	bHa	58.24	USH	69.84	UbB	33.75
bSH	78.34	Uba	68.83	VHa	53.55	bSH	65.93	HbB	31.50
RGB	57.00	RGB	52.22	RGB	45.70	RGB	54.00	RGB	13.25

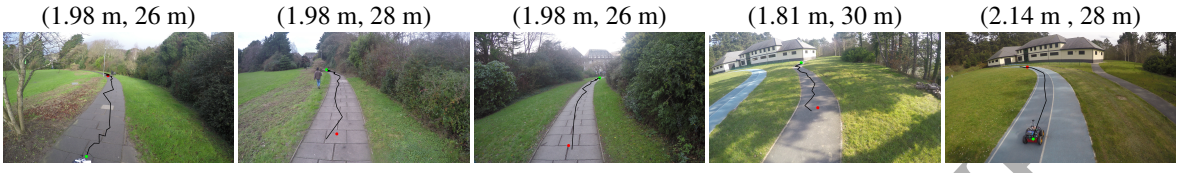


Figure 5: Outdoor environments. In each image, the black line refers to the robot's trajectory for a trial using the USH colour model. The green circle shows the starting position, and the red circle denotes the end of the trajectory. Width (m) and length (m) of each path is indicated above each image.

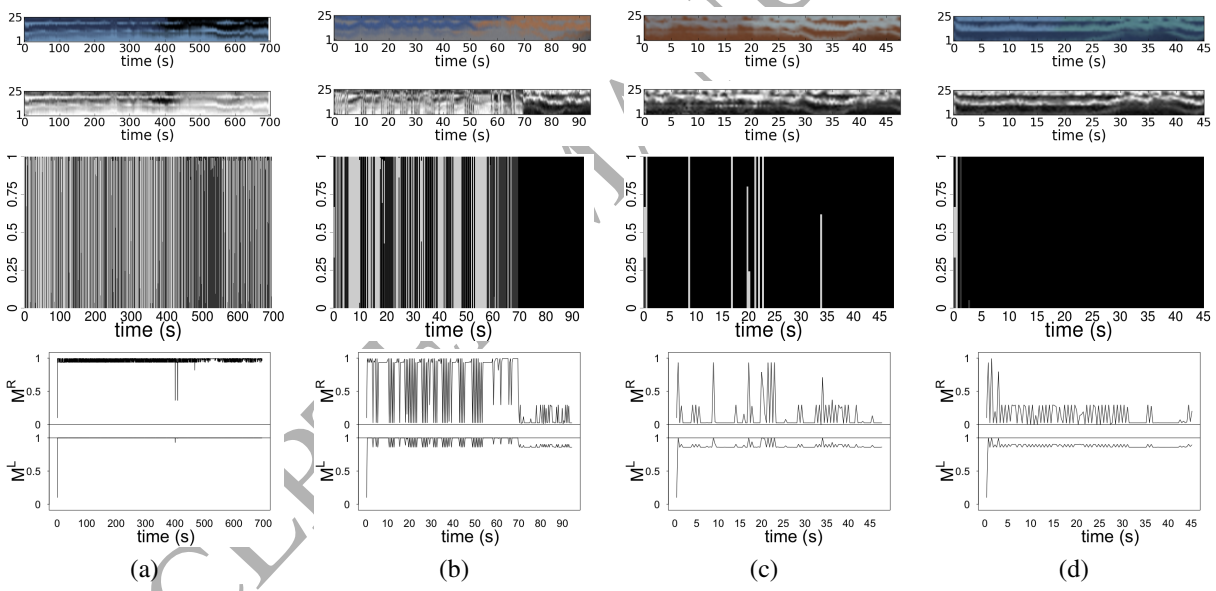


Figure 6: colour component activation. Each column refers to a different trial in a different environment, with the simulated robot controlled by the B-controller linked to the RGB colour model. The graphs in the top row are a representation of the visual scenes experienced by the robot. The graphs are constructed by taking into account, at each update cycle of the robot controller, the contribution of C_R, C_G, and C_B (see Section 3.1 for details) for each of the 25 grid cells superimposed on the camera image. At each update cycle, the 25 coloured points are distributed over the y-axis. The graphs in the second row from the top refer to the variation over time of the 25 values sensory input vector (i.e., I_i, see Equation (1)). At each update cycle, the 25 grey scale points corresponding to the vector I_i are distributed along the y-axis. The graphs in the third row from the top refer to the variation over time of the colour parameters ρ (light grey), γ (dark grey) and β (black). The shades of grey indicate the activation of each colour parameter at each update cycle. The graphs in the fourth row from the top refer to the output to the left (M^L) and right motors (M^R).

Aparajit Narayan obtained his B.Eng (Hons) in electronic engineering from the University of Sheffield in 2013. He is currently completing his PhD in computer science from Aberystwyth University (UK), working on a project funding by Fujtisu-HPC Wales. His primary research interests are in evolutionary robotics and deep-learning, especially with regards to the application of control strategies emerging from these fields to various real-world problem domains.

Dr. Elio Tuci is a Senior Lecturer in the Department of Computer Science, Middlesex University (London, UK). He received a PhD in Computer Science and Artificial Intelligence from the University of Sussex, Brighton, UK in 2003. His research interests fall into the interdisciplinary domain of bio-inspired robotics and computational intelligence, drawing inspiration from nature to design control mechanisms to allow artificial agents to operate in a complex environment and to learn from their experience in an autonomous way.

Dr Frédéric Labrosse obtained his degree from Université Paris XI, France, in 1991 and his PhD from École Polytechnique de Montréal, Canada, in 1999. He currently is a Senior Lecturer in Computer Science, Aberystwyth University, UK and his research concerns Robotics and Computer Vision, and in particular the interface between the two fields. As such he has worked on visual sensors and visual navigation methods. He is also interested in the practical applications of robotics, collaborating with many customers who need sensors in remote, dangerous, places.

Muhanad Alkilabi obtained his electical engineering bachelors degree from Al-Mustansiriyah University, Baghdad in 2003 and a Masters degree from Pune University, India in 2007. He is a member of faculty in the Computer Science department at the University of Kerbala and is currently pursuing his PhD in swarm robotics from Aberystwyth University, UK. His research interest is in bio-inspired robotics and computational intelligence.

Aparajit Narayan:

Dr Elio Tuci

Dr F. Labrosse

Muhanad Alkilabi