

An evolutionary approach to road following: a simulated case study

Sebastian Clarke¹, Frédéric Labrosse², Vito Trianni³ and Elio Tuci⁴

^{1,2,4}Aberystwyth University, Llandinam Building, Aberystwyth, SY23 1RE, UK
CNR-ISTC, Via S. M. della Battaglia 44, 00185, Rome, Italy
sec14@aber.ac.uk¹, fl@aber.ac.uk², vito.trianni@istc.cnr.it³, elt7@aber.ac.uk⁴

Abstract

Autonomous robot road following has been widely investigated since the early 1980s and, whilst much progress has been shown, there is still no system which displays 100% generality across all types of problem. This work shows a novel approach to the problem, using the methodology of Evolutionary Robotics to facilitate the autonomous emergence of flexible, robust and general behaviours. One of the unique aspects of this approach is to encourage the evolution of a dynamic strategy of colour perception: facilitating the combination of different channels of the colour space to perceive contrast across a range of scenes where this would otherwise be impossible. The results described herein demonstrate the capability of this methodology to produce controllers capable of generalising across a broad range of road shapes to which the agents have not been previously exposed. They also vindicate the effectiveness of a dynamic colour perception strategy, enabling the controllers to perceive contrast in a challenging variety of situations.

Introduction

In the realm of autonomous vehicles, road following, i.e., the ability to detect and traverse a road surface without straying from the boundaries, is clearly an important problem. As such, it has received much attention from artificial intelligence and autonomous robotics researchers over the last thirty years (Dickmanns, 2002).

The earlier attempts employed hand-crafted controllers focusing on roads that are clearly demarcated and delineated (e.g., those that have either white lines or a clear high contrast delineation between the road and non-road surfaces, see Wallace et al., 1986; Waxman, 1985; Turk et al., 1988; Kuan et al., 1988; Dickinson and Davis, 1988). The general thrust of these approaches is to establish some sort of model based on the *a priori* assumption that the agent is already situated on the road, and by sampling its sensory information accordingly. This model is then maintained by monitoring prominent road features such as edges and lane demarcations. In order to improve the flexibility and the capability of a solution to follow more road types, later research looked at roads which have no clear demarcation, have amorphous or unclear delineation of the road edges

and have low contrast between the road and the background surface, and are subject to changing road conditions due to e.g., shadows or reflections (see Crisman and Thorpe, 1988; Kluge and Thorpe, 1990). To further address this problem, researchers started to investigate more adaptive, learning and connectionist based, rather than hand-crafted approaches (see Jochem et al., 1993; Pomerleau, 1997; Dickmanns, 2002). Recent success in this field has also been gained by combining these approaches with higher level, more complex models and reasoning, facilitated by the increase in available computing power over the last decade and spurred by high profile involvement from the military and commercial sectors (see Chen and Tsai, 1997; Aufrere et al., 2000; Urmson et al., 2008). Most recent approaches to the road following problem then, range from these complex, high-level models updated from multiple sensory sources and requiring significant computational power, to more simple, reactive and robust systems with lower model complexity and hence computational requirements (see Katramados et al., 2009; Ososinski and Labrosse, 2012).

The road following problem, in common with many others, has at its root a strong visual perception and feature extraction component. One of the problems in such visual discrimination tasks is a method of processing the input images in such a way as to reduce environmental distractions (in this case for example, those arising from shadows or reflections) so as to allow extraction of the features relevant to the problem, and to show clear contrast between the features to be extracted. One such technique could be to examine and combine different representational components of the image when transformed into colour spaces that separate luminosity (brightness) information from chrominance (colour) information. This can allow, for example, the efficient removal of shadows and effective dimensionality reduction of the problem by combining the remaining colour components in ratios that provide good performance over a range of possible scenes (see Woodland and Labrosse, 2005; Benedek and Szirányi, 2007; Finlayson et al., 2006).

However, it should be obvious that a fixed combination of colour components will never be able to show good con-

trast for all possible road following scenarios: we may well encounter scenes for which a broadly good combination of colour components actually shows very little contrast due to the nature of the colours represented in the scene. One solution to this problem may be to examine multiple components of the colour space simultaneously, dynamically choosing only those components (or combinations of components) that yield high contrast. However, for systems where input informational throughput may be limited such as low power embedded systems, or those using neural networks (i.e., where dimensionality reduction is necessary), it will not be possible to appraise all of this information concurrently. Similarly, there may be colour channels which yield very high contrast within the image, but not amongst the features that are relevant to the particular problem at hand. It should be clear then, that a method of dynamically combining various components of colour spaces in a way which aids feature detection by removing unwanted artefacts and increasing the contrast available between the relevant features for extraction is required for optimal performance of such visual perception and discrimination tasks.

With that in mind, we propose a solution to the road following problem using artificial neural network controllers synthesised by evolutionary computation techniques. We illustrate a system that is capable of successfully navigating roads in a variety of distinct simulated environments based on visual information in an integrated action-perception loop where the neural mechanisms that govern perception (identification of the road features) and action (motor activations and changes in colour perception) are not only tightly coupled, but are one and the same. In this study, we examine the combination of the standard RGB colour space components to increase contrast between the desired features, under the assumption that a similar process could be used with components of other colour spaces (that better separate luminance from chrominance) to augment this aim, and to reduce disturbances caused by shadow or reflections. We analyse how a system of dynamic colour perception contributes to the overall road-following effectiveness of the controllers and we show that without this adjunct to perception, the agent would be unable to solve all of the road following tasks presented in this simulated environment.

In summary then, the main objective of this study is to apply evolutionary robotics to the road following problem to produce a solution which will coordinate vision and action in a single, unified sensory-motor controller. Further, we aim to show that a dynamic method of dimensionality reduction, arrived at in tandem with the road following behaviour, will prove beneficial in solving the problem at hand. The evolved agents are capable of developing a general strategy for staying on poorly demarcated and delineated roads based on single camera visual input. Moreover, they are also able to dynamically adjust their colour perception, in real time, to increase the efficacy of road following in environments with



Figure 1: Pioneer robot

different or changing colour properties.

The Robot and the Simulation Environment

A simulated robot is required to navigate various types of road using visual input. Our simulation models a Pioneer 3-AT 4-wheeled skid steer all terrain research robot as shown in Fig. 1. The simulation also comprises a 3D model of an environment, rendered using OpenGL (<http://www.opengl.org>) that provides the sensory information that the robot perceives through its camera. This environment contains only 3 visual components: a tiled textured horizontal plane on which the robot travels (the ground), a textured deviated surface rendered on this plane (the road), and a sky-box to provide the illusion of sky.

The virtual camera renders the 3D scene from the point of view of a camera mounted on top of the virtual robot. It is configured to have a frustum representative of real-world cameras that may be used to capture the scene. For final evolution to produce effective solutions, this would need to match exactly the specific camera being used. As this had not been chosen at the time of evolution, representative values were used with the understanding that changing the frustum would not significantly affect the results and that we are showing that the general concept of using camera images in this way is sound.

The visual input for the robot's controller is generated by significantly reducing the resolution of each camera image. Our simulated camera renders images at 250×200 pixels. We overlaid a 5×5 grid on each image, with each square grid covering 2000 pixels. The robot sensory input is made of 25 numerical values generated as follows. For each grid square, a mean value for each colour component is calculated by summing red, green, and blue components of each pixel within this square separately, then dividing by the total number of pixels residing in this grid square. In this way we have 3 values per grid square corresponding to the contributions of red, green and blue elements in the image. We then combine these components by multiplying each by α , β , and γ respectively, and then summing to produce a final single numeric value for each square. α , β , and γ are floats in $[0, 1]$ generated by the robot controller at each time step. They are normalised so they sum to 1 hence they represent the ratios in which the red, green, and blue channels should be mixed

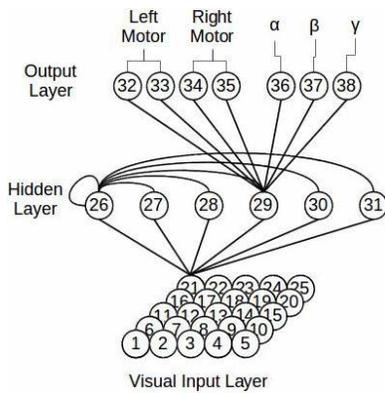


Figure 2: The neural network. The lines indicate the efferent connections for only one neuron of each layer. Each hidden neuron receives an afferent connection from each input neuron and from each hidden neuron, including a self-connection. Each output neuron receives an afferent connection from each hidden neuron.

respectively. As pixel colour values are in the range 0-255, we then divide by 255 to scale this value between 0 and 1.

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 (see Beer and Gallagher, 1992). The structure of the network is shown in Fig. 2. The states of the output neurons are used to control the speed of the left and right wheels as explained later, and they define the ratios for colour mixing. The values of sensory, internal, and output neurons are updated using equations 1, 2, and 3.

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

$$\tau_i \dot{y}_i = -y_i + \sum_{j=1}^{31} \omega_{ji} \sigma(y_j + \beta_j); \text{ for } i = \{26, \dots, 31\}; \quad (2)$$

$$y_i = \sum_{j=12}^{15} \omega_{ji} \sigma(y_j + \beta_j); \text{ for } i = \{32, \dots, 38\}; \quad (3)$$

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 $i = \{1, \dots, 11\}$ is the activation of the i^{th} sensor neuron, ω_{ji} the strength of the synaptic connection from neuron j to neuron i , β_j the bias term, $\sigma(y_j + \beta_j)$ the firing rate (hereafter, f_i). All sensory neurons share the same bias (β^I), and the same holds for all motor neurons (β^O). τ_i and β_i with $i = \{26, \dots, 31\}$, β^I , β^O , all the network connection weights ω_{ij} , and g are genetically specified networks' parameters. At each time step, the output of the left motor is $M^L = f_{33} - f_{32}$, and the right motor is $M^R = f_{35} - f_{34}$, with $M_L, M_R \in [-1, 1]$. The firing rates f_{36}, f_{37}, f_{38} are



Figure 3: Road tile construction and circle approximation

normalised such that $\alpha + \beta + \gamma = 1$. Cell potentials are set to 0 when the network is initialised or reset, and equation 2 is integrated using the forward Euler method with an integration time step $\Delta T = 0.2$.

A simple evolutionary algorithm using linear ranking is employed to set the parameters of the networks (Goldberg, 1989). The population contains 60 genotypes. Generations following the first one are produced by a combination of selection with elitism, recombination, and mutation. For each new generation, the three highest scoring individuals (“the elite”) from the previous generation are retained unchanged. The remainder of the new population is generated by fitness-proportional selection from the 30 best individuals of the old population. Each genotype is a vector comprising 243 real values (228 connections, 6 decay constants, 8 bias terms, and a gain factor). Initially, a random population of vectors is generated by initialising each component of each genotype to values chosen uniformly random from the range [0,1]. New genotypes, except “the elite”, are produced by applying recombination and mutation. Each new genotype has a 0.3 probability of being created by combining the genetic material of two parents. During recombination, one crossover point is selected. Genes from the beginning of the genotype to the crossover point are copied from one parent, the other genes are copied from the second parent. Mutation entails that a random Gaussian offset is applied to each real-valued vector component encoded in the genotype, with a probability of 0.05. The mean of the Gaussian is 0, and its standard deviation is 0.1. All vector component values are constrained to remain within the range [0,1].

The visual scene

Textures that represent real world scenarios were chosen for the road and ground surfaces from the plethora available at the multitude of free texture resources on the internet. 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.

We devised three complementary scenes, each of which featured only two of the three colour components (red, green and blue). The third component is randomly varied with noise, and hence unable to contribute to the final contrast between road and background visible in the scene. Giving

this randomly varied colour component strong weight in the colour combination would actively detract from the structured contrast visible between road and non road surfaces. In this way, the agent has to choose—by appropriately setting α , β , and γ for each environment—the colour components which assist in solving the problem, and disregard those that do not show the pattern that is being sought. This random colour variation was first undertaken at the pixel level, but later reimplemented to occur at the grid average level. In this way, with any static (i.e., not changing with respect to the environmental features) selection of colour combination there would always be one scene in which the controller was unable to detect any contrast. In addition, as there was a requirement that the agents evolve a general strategy that was able to cope with the road surface being both brighter (having higher values) than the background, and vice versa, scenes were carefully devised with the properties shown in Table 1. As long as the differences between the higher and lower values were kept constant, this should result in the following attributes. In 3 colour grey scale, all of the scenes would show no contrast between the road and non road surfaces. By choosing a single colour component, the agent would see positive contrast for the road in one scene, negative in another, and no contrast in the third. By choosing to combine two fixed colour components, the results would be the same. With this configuration, the only means by which the agent can successfully navigate all three scenes, and hence score maximum fitness, is by changing which colour components are examined between or during trials.

Each agent is evaluated against 6 different environments: two road shapes (one starting with a left bend, the other with a right bend) for each of the 3 colour scenes. At each generation, 6 different road shapes are generated using the following algorithm. First, a single square road tile is placed at the centre point of the ground plane at a world relative heading of 0° . A random angle is then chosen between two bounds (initially $\pm 20^\circ$) and a centre point coordinate position is calculated for the new tile by applying basic trigonometry to translate it forward by 0.75 times the size of the road tile along this new angle. The new tile is then placed at this position and rotated by the angle, as shown in 3. This process is repeated for the number of road tiles required (in our experiment, 20) with the centre point position and total angle of each road tile stored for later use. In rendering, only the

Table 1: Table showing road scene properties. H refers to higher values, L refers to lower values.

	Red	Green	Blue
Scene 1	Noise	L on road	H on road
Scene 2	H on road	Noise	L on road
Scene 3	L on road	H on road	Noise

angles are required: each road tile is rotated by, and translated along, this angle by OpenGL.

The Fitness Function

All of the individuals in a population are evaluated against the same 6 environments to yield a proper comparison of the agent’s performance, and new random road shapes were generated for each generation to expose them to as wide a variety of road following scenarios as possible. At the beginning of each trial (e), the robots are placed at the start of the road at a random orientation between $\pm 30^\circ$.

The fitness function used in this approach is heavily inspired by that found in (Suzuki et al., 2005). This is a function which rewards forward progress of the robot, and is calculated and tallied at each time step based on the left (S_l) and right (S_r) wheel speeds, and the naïve straight-line distance reward applied at the end of the trial. The distance from the starting point to the final agent position is calculated as a percentage of the distance from the starting point to the end point of the road. This is multiplied by a reward factor and added to the fitness to further encourage the agent to reach the end of the road. There are situations however where this strategy will be counter-productive: e.g., when a road curves round so that the end point is closer to the start point than other positions on the road where the trial might terminate. However, it was concluded that with our random strategy of road building, such occurrences are rare enough not to adversely affect the system.

To combat the behaviour of a robot travelling in tight circles at the start point in order to “game” the fitness function without traversing any of the road, a penalty was added, such that the final fitness is halved if the agent remains on the starting road tile at the end of the trial. Furthermore, to encourage even more strongly the robot to avoid leaving the confines of the road, the fitness is multiplied by a factor of 1.2 if the trial was not terminated by a failed road bounds check. This serves to make a clear distinction between behaviours where the robot gets most of the way down the road, but leaves the road at the very end of the trial, and where the robot reaches the end of the road without doing so. Therefore, the final fitness (F) for each genotype is calculated as:

$$F = \frac{1}{E} \sum_{e=0}^E f_e; \quad (4)$$

$$f_e = \eta \lambda \left(\frac{1}{T} \sum_{t=0}^{T'} g(S_l, S_r, t) + H \frac{D_a}{D_r} \right); \quad (5)$$

$$g(S_l, S_r, t) = (S_l^t + S_r^t) \left(1 - \sqrt{\frac{|S_l^t - S_r^t|}{2 \times S_{max}}} \right); \quad (6)$$

where $E = 6$ is the number of evaluations or trials per genotype, $\eta = 0.5$ if the robot remains on the first road tile at the

end of the trial, otherwise $\eta = 1.0$; $\lambda = 1.2$ if the robot does not leave the road during the trial, otherwise $\lambda = 1.0$; T is the maximum number of time steps in a trial (180 for these experiments); T' is the number of time steps experienced by the agent during this trial (for example, 100 where the trial is terminated after 100 time steps due to the robot leaving the road); D_a is the straight line distance from the start point to the final agent position at the end of the trial and D_r is the straight line distance from the start point to the centre of the final road tile; $H = 1000$ is a constant, S_{max} being the maximum allowable wheel speed setting, in this case 300.

It should be noted that this fitness function makes no mention of the road, and the requirement to stay within its bounds. Rather, the trial is stopped prematurely when a robot leaves the road surface, so maximum fitness is only available to those agents that are capable of staying on the road throughout the full duration of the trial. To check whether a robot is on the road, we used the following criteria. The road is treated as a series of overlapping circles rather than squares, as shown in Fig. 3. By checking if the centre of the robot is further away from a centre point than the circle approximation radius, we can tell if it lies inside or outside of this circle.

Results

Our objective is to synthesise controllers for autonomous robots required to visually navigate road surfaces without straying from the boundaries. We looked at roads which have unclear delineation of the edges, and situated in environments with different colour properties. The robots are required to dynamically combine—by appropriately setting α , β , and γ for each environment— various components of the colour spaces in order to detect road edges and to distinguish the road from the background.

Ten evolutionary runs, each using a different random initialisation were carried out for 2000 generations. Two evolutionary runs managed to generate robots with sufficiently high fitness to indicate that they are capable of successfully navigating all the three road scenes. The other eight runs produced only sub-optimal solutions. Due to the nature of the evolutionary process and fitness function, we cannot guarantee that the individuals with the highest fitness are those that have evolved the most robust general strategies. It is likely, in fact, that these fitness values actually represent the agents that are the “luckiest” with respect to the random variation occurring in the simulation. To deduce which are actually the most useful evolved individuals, we further evaluated a selection of the most fit genotypes across a broader range of tests with systematic variation.

Post-evaluation test I

In this test, a suite of road following scenarios is generated, with parameters pertaining to road shapes and starting angles systematically adjusted between them. The fittest in-

dividual from the 100 fittest generations from each of the two successful evolutionary runs are then evaluated against the same set of conditions, allowing for a side by side comparison of their general effectiveness at such road following tasks. Together, the evaluations performed in Test I represent an examination of the effectiveness of the agent’s road following behaviour across a set of scenarios to which it has not been exposed during evolution. This test, therefore, is performed with the aim of demonstrating the generality of the road following solution produced.

The evaluation scenarios are produced by varying the allowable bounds between which angles (θ) are chosen for road tile placement. Two roads are generated for each of the following four configurations (for a total of 8 scenarios), where θ is a randomly selected angle that each tile is placed at, between the following bounds: 1) ± 20 , 2) ± 30 , 3) ± 40 , and 4) ± 50 . 6 further scenarios are generated using roads featuring smooth, contiguous bend, where the tile placement angle is kept constant and uniform between road tiles. Three constant tile placement angle values are used to generate these roads, corresponding to shallow, medium and sharp corners: 20° , 30° and 40° . Angles greater than these resulted in unrealistic looking roads with tighter corners than one would reasonably expect such a road-following vehicle to be capable of traversing. To avoid the road looping back on itself to form a circle, the direction of the placement angle is reversed once the total corner angle reaches 110° . Two roads are generated for each of the three cornering angles described (for a total of 6 scenarios), the first starting with a left turn, the second with a right. Finally, we included a straight road without corners to ensure that agents are effective on simpler tasks.

All of the above 15 evaluation scenarios (8+6+1) are generated with a fixed tile width of 140cm. Visually, they are of mostly uniform but slightly variable width due to the noisy fading to background of the road edges, representing more ill-defined roads. The visible road width is therefore roughly 120cm, or roughly twice the diameter of our simulated robot. Accordingly, the circle radius used for the road bounds approximation test is 60cm. Each road has a total traversable length of 28m. Each of the above road shapes is rendered in the three colour and texture combinations used for evolution, resulting in 45 evaluation scenes. Individuals are evaluated against each scene 5 times with different initial robot headings. Relative to the first road tile placement angle, these are: -45° , -22.5° , 0° , 22.5° and 45° . A trial is considered successful if the robot successfully navigates to the penultimate road tile, to take into account erratic behaviour caused by the road ending in the robot’s field of view. In unsuccessful trials, the percentage of navigable road tiles successfully traversed is recorded.

The results of test I are shown in Table 2. We can see from this table that the best performers succeed in getting to the end of the road in almost 85% of the 225 individual evalua-

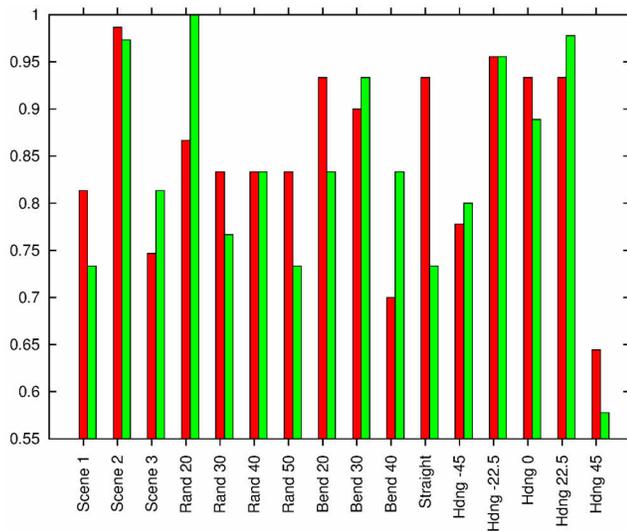


Figure 4: The performance of the best 2 individuals broken down by test type. Red bars for agent 2-1834, green bars for agent 6-1802.

tions. This is no mean feat, as they not only contain a challenging set of environments featuring diverse colours and textures and some obtuse, complex road shapes with sharp turns and currently unnavigable portions of the road in the field of view, but also place robots at a variety of starting headings, pointing more towards the sides of the road than the agents were evolved to cope with. It is also clear that both successful evolutionary runs produced roughly equally effective solutions.

Using the best individual from each run, measured by success percent, we looked at how they fare with the different problem types. In Fig. 4 the percentage success is broken down across the three coloured road scenes, the different angled random and bendy roads, the straight road, and different starting headings. Comparing the three coloured scenes

from Fig. 4 we can see that both agents perform better on scene 2 than the others. This could be due to it being a slightly easier colour/texture combination than the others, with slightly more contrast visible between road and non-road areas, but it may also be down to random occurrences in the pattern of evolution. The agents likely evolve to solve a particular coloured scene first, before learning to change their colour perception—it might simply be that both agents happened to learn this scene first, and therefore had more “practice” completing it.

Comparing the performance across the set of random roads with different placement angles, we can see that the agent from run 2 (hereafter, Agent 2) showed roughly uniform performance of around 85% across all 4 road types. This serves to show that it has evolved a highly general, effective road following behaviour, with difference in success rates largely unaffected by the coarseness of random road angles. The difference in performance between Agent 2 and the agent from run 6 (hereafter, Agent 6), at the 20° random road is striking and somewhat surprising. One might expect both agents to perform very well on this road as it most closely resembles those that they were evolved against. We can also see that Agent 6’s performance drops off on the other angled roads, in contrast to Agent 2’s broadly uniform performance, suggesting that Agent 6 has specialised on the 20° road to the detriment of the more difficult roads, whilst Agent 2 represents the more general solution at these road types, though showing somewhat worse performance on the simpler challenges. Interestingly, this situation does appear to be reversed for straight roads, where Agent 2 significantly outperforms Agent 6.

We can also see that both agents perform uniformly in a very effective manner when started at angles in the range $-22.5^\circ \leq \theta \leq 22.5^\circ$, with a drop off in performance for both agents when placed at angles beyond this range. This can be explained by the fact that although, during evolution, all agents are placed at world relative heading of 0° , the random changes in road tile angles will subject them to situations where they are pointing up to 20° from the centre line of the road, and they have hence developed strategies to mitigate this situation. What is surprising is that both agents perform worse when the agent is facing far to the left (45° starting angle) than when it is subjected to such angles in the opposite direction. There is no obvious reason for this from the attributes of the evolutionary trials, other than that, through random fluctuations, the agents may have been exposed to more left turns than right. In summary then, the agents complete these evaluation tests with high effectiveness and there is not a great deal to choose between them. Any significant out-performance by one agent in a test, is made up for, either by more generality of the other agent across more tests, or an out-performance of the other agent on a different test. This demonstrates therefore, the capability of the evolutionary process to produce effective agents that encapsulate a

Table 2: The top ten Test 1 performers

Run-Gen.	Success %	Mean distance	Std. dev.
2-1834	84.89	14.87	27.27
6-1802	84.00	5.25	12.89
2-1972	83.11	15.79	25.12
6-1939	83.11	8.92	15.27
2-1966	82.22	19.03	26.52
6-1825	82.22	15.28	20.16
2-1951	81.78	17.75	22.03
2-1838	81.33	13.10	25.68
6-1882	80.44	10.61	19.17
2-1808	80.44	7.95	16.80

general solution to the road following problem and are able to perform successfully across a wide variety of road types.

Post-evaluation test II

Having analysed the generality and effectiveness of the road following behaviour in the previous test, it would also be useful to appraise the performance of the dynamic colour selection strategy evolved to show contrast in a variety of colour and texture scene combinations. To this end, the agent is exposed to 30 different scene texture combinations, representing all the possible combinations of the 6 textures shown in Fig. 5, using each texture as both road and ground surface. In order that the variance that we see in trial performance is due to the difference in texture combinations, rather than the varying performance across road shapes and starting angles, each agent is tested against roads generated using the same algorithm as that used in evolution to represent challenges of the sort with which the evolved agents should be most familiar. Similarly, rather than varying the starting heading of the agents on the road, each agent is started with a world relative heading of 0° as in evolution. 75 such roads are generated, and rendered in each of the 30 possible texture combinations, resulting in 2250 total evaluations per agent.

The results of this test, with respect to the different coloured scenes are plotted in Fig. 6. The labels for the bars of the histogram are in the form X/Y/Z where X is the agent number, Y is the first texture and Z is the second, according to the labels in Fig. 5. The bars are grouped by agent for each texture combination: the left bar being Agent 2, the right Agent 6, allowing direct comparison between the agents on different scenes. Each individual bar represents the mean percentage success rate for one agent across 150 trials: 75 road shapes with a particular road/ground texture combination, and the same 75 shapes with the reverse combination. The proportion of light and dark areas of each bar then represent the success rate for each of these individual reversals. For example, the bar labelled '2/B/T3' shows the performance of Agent 2, using a combination of 'Blue asphalt' and 'Asphalt 3' textures: the dark portion of the bar showing the success rate with 'Blue asphalt' road on 'Asphalt 3' ground, the light portion showing the reverse.

From the results of these tests, a few issues of note become apparent. Firstly it would appear that Agent 6 broadly outperforms Agent 2 in many of these trials. This likely suggests that Agent 6 has evolved a more general and effective

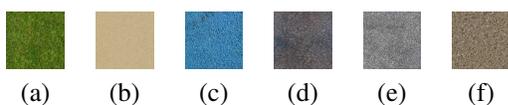


Figure 5: a) Grass (G), b) Sand (S); c) Blue asphalt (B); d) Asphalt 1 (T1); e) Asphalt 2 (T2); f) Asphalt 3 (T3)

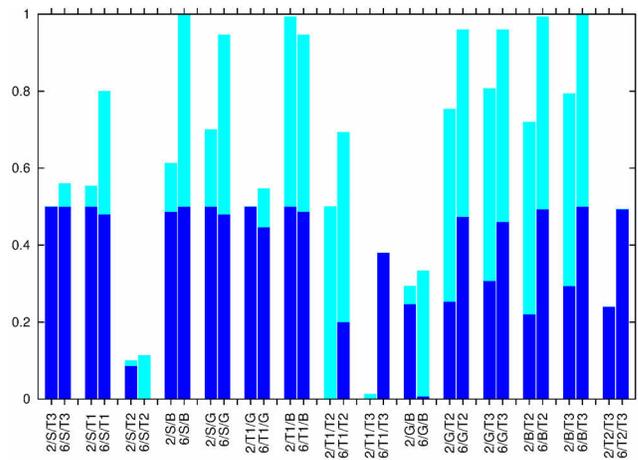


Figure 6: Comparison of both agent's performance across a range of scenes.

strategy for dynamic colour perception than Agent 2, though it is also possible that this is in some part due to it having evolved road following strategies that are more specialised to the type of road shapes present in evolution. We can also see that, for the majority of texture combinations, there is at least one reversal of road and ground textures that an agent is able to solve to a high degree of efficiency, i.e., there is normally a coloured bar portion at least 0.4 units long, a percentage success rate of 80 % for this road/ground texture combination. There is also a problem visible here though, as a number of scenes can only be solved effectively in one of the two reversals of road and ground texture. This shows a lack of generalisation in the solution. There is however another problem visible in this data. There are some scenes in which no agent can reliably detect contrast and solve the road, for either reversal of road and ground textures. This effect is most pronounced for S/T2 texture combination and it is believed this is due to the colours in both textures being too similar for either combination of them to reveal significant contrast.

Conclusion

In this work, we have demonstrated a technique, using evolutionary robotics, to design effective road-following behaviours in simulated agents controlled by artificial neural networks. We have shown that, by presenting a set of challenges with diverse colour properties, we can encourage the evolution of an autonomous, dynamic approach to colour perception which enables evolved agents to perceive contrast in scenes where this would otherwise be impossible.

The process of evolution seems to have produced a range of effective and general solutions, which encapsulate not only a robust solution to road following, but also a system of dynamic colour perception that is able to show contrast be-

tween road and non-road surfaces across a range of scenes where a constant grey-scale conversion would fail. Through performing a range of tests, we ascertained which of the solutions is more generally effective, irrespective of their fitness scores which could be influenced by luck. By breaking down the results of post-evaluation tests, we have shown that the most effective agents show good generality in their road following ability, being capable of following roads differing significantly from those that they were evolved against. Their generality with respect to starting angles is not as strong, but this is expected as they were not deliberately exposed to a representative selection of these when evolving.

In examining the most effective produced agents with respect to their colour and road perception abilities it becomes clear that there are a few limits to their generality. In some instances (though not in others) the agents are not able to follow both reversals of road and ground textures, suggesting an inability to deal with certain combinations of values demarcating the road, even when contrast is visible. We have suggested that this could be mitigated either with more diverse evolution scenes, or an extra output node to reverse the visual input values. Similarly, we believe the slightly reduced performance on a couple of scenes in post-evaluation tests can also be improved with a better strategy to noise in the simulation. However, in spite of these slight problems, it seems the broad aim of evolving controllers with a dynamic approach to colour perception has been met, and the agents are able to detect contrast in a number of scenes where this would otherwise be impossible.

This work has been undertaken as a theoretical proof-of-concept: to show that the desired road following and dynamic colour perception behaviours can be produced through artificial evolution of neural network controlled robots. The transferral of such a system onto real robotics hardware has not been broached. We are aware that there are a number of issues which may affect the ability of this evolved controller to successfully cross the "reality gap". Future work will concentrate on this challenge.

References

- Aufreere, R., Chapuis, R., and Chausse, F. (2000). A fast and robust vision based road following algorithm. In *Proc. of the 4th IEEE Intelligent Vehicles Symposium, 2000*, pages 192–197.
- Beer, R. D. and Gallagher, J. C. (1992). Evolving dynamic neural networks for adaptive behavior. *Adaptive Behavior*, 1(1):91–122.
- Benedek, C. and Szirányi, T. (2007). Study on color space selection for detecting cast shadows in video surveillance. *International Journal of Imaging Systems and Technology*.
- Chen, K.-H. and Tsai, W.-H. (1997). Vision-based autonomous land vehicle guidance in outdoor road environments using combined line and road following techniques. *Journal of Robotic Systems*, 14(10):711–728.
- Crisman, J. and Thorpe, C. (1988). Color vision for road following. In *Proc. of Conf. on Mobile Robots (SPIE)*, pages 1–10.
- Dickinson, S. and Davis, L. (1988). An expert vision system for autonomous land vehicle road following. In *Proc. of the Computer Society Conf. on Computer Vision and Pattern Recognition*, pages 826–831.
- Dickmanns, E. (2002). The development of machine vision for road vehicles in the last decade. *Intelligent Vehicle Symposium, 2002. IEEE*, 1(1):268–281.
- Finlayson, G., Hordley, S., and Drew, M. (2006). Removing shadows from images. *Computer Vision ECCV 2002*, pages 1–14.
- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, MA.
- Jochem, T., Pomerleau, D., and Thorpe, C. (1993). MANIAC: A next generation neurally based autonomous road follower. In *Proc. of Int. Conf. on Intelligence Autonomous Systems*, pages 15–18.
- Katramados, I., Crumpler, S., and Breckon, T. (2009). Real-time traversable surface detection by colour space fusion and temporal analysis. *Computer Vision Systems*, 5815(2007):265–274.
- Kluge, K. and Thorpe, C. (1990). Explicit models for robot road following. *Vision and Navigation*, pages 25–38.
- Kuan, D., Phipps, G., and Hsueh, A.-C. (1988). Autonomous robotic vehicle road following. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(5):648–658.
- Ososinski, M. and Labrosse, F. (2012). Real-time autonomous colour-based following of ill-defined roads. *LNAI: Advances in Autonomous Robotics*, pages 366–376.
- Pomerleau, D. (1997). Neural network vision for robot driving. *Intelligent Unmanned Ground Vehicles*, pages 1–22.
- Suzuki, M., Floreano, D., and Di Paolo, E. (2005). The contribution of active body movement to visual development in evolutionary robots. *Neural Networks*, 18(5-6):656–665.
- Turk, M., Morgenthaler, D., Gremban, K., and Marra, M. (1988). Vits-a vision system for autonomous land vehicle navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(3):342–361.
- Urmson, C., Anhalt, J., and Bagnell, D. (2008). Autonomous driving in urban environments: Boss and the urban challenge. *Journal of Field . . .*, 25(February):425–466.
- Wallace, R., Matsuzaki, K., Goto, Y., Crisman, J., Webb, J., and Kanade, T. (1986). Progress in robot road-following. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, volume 3, pages 1615–1621.
- Waxman, A. (1985). Visual navigation of roadways. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, volume 2, pages 862–867.
- Woodland, A. and Labrosse, F. (2005). On the separation of luminance from colour in images. In *Proc. of the Int. Conf. on Vision Video and Graphics*. The Eurographics Association.