

Simulated Road Following using Neuroevolution

Aparajit Narayan, Elio Tuci, Frédéric Labrosse

Aberystwyth University, LLandinam Building, SY23 3DB, UK
{apn3,elt7,ffl}@aber.ac.uk

Abstract. This paper describes a methodology wherein genetic algorithms were used to evolve neural network controllers for application in road-following. The simulated controllers were capable of dynamically varying the mixture of colour components in the input image to ensure the ability to perform across the entire range of possible environments. During the evolution phase, the agents were evaluated in a set of environments, carefully designed to encourage the development of flexible and general-purpose solutions. Successfully evolved controllers were capable of navigating simulated roads across challenging test environments, each with different geometric and colour distribution properties. These controllers proved to be more robust and adaptable compared to the previous work done using this evolutionary approach due to their improved dynamic colour perception capabilities where they extracted features across all three primary colour channels.

Keywords: Road-Following, Genetic Algorithm, Neural Network, Dynamic Dimensionality Reduction, Autonomous Navigation

1 Introduction

Autonomous navigation in its entirety is a vast and diverse field of study and research tends to be focussed on a number of sub-areas, such as steering control, obstacle avoidance, road-following, power management and road-sign detection. Amongst these, road-following is an essential foundation of any system with desired autonomous navigation capabilities. While it may seem a trivial problem from a human perspective, accurately extracting the desired features in the environment and using them to navigate the road successfully is indeed a significant problem in terms of an AI system. This is particularly due to the amount of variance and non-uniformity present in terms of the geometry and colour composition of the road non-road surfaces. Weather conditions such as rain, shadows, changing sunlight etc. all of which have an effect on the system's visual perception of the environment, further compound this problem.

1.1 Related Work

The design philosophy behind most engineered road-following solutions is based on maintaining an internal model of the road non-road environment which is continuously updated based on the features extracted from the world. A commonly

used technique is to use sensor fusion, combining sensory data from multiple cameras and laser range-finders, to produce a more detailed and accurate representation of the world. This approach to the road following problem, especially when adaptive techniques for maintaining the model of the environment are used has indeed been successful in real-world trials. However at the core of all such hand-crafted controllers is the issue of designer bias and the assumptions that are made of the road with regards to its geometry, contrast and colour composition. Thus successful performance may be guaranteed in environments accounted for in the design process, but not across the entire range of scenarios possible such as in the case of [3] and [8] where geometric assumptions and limited detail meant that the model was less suitable for more complex road-shapes. Even in the case of systems where the road model is adaptive and has high levels of detail and accuracy, the cost and hardware requirements often make their implementation prohibitive in most platforms. There have been a few attempts to use traditional machine learning strategies to train neural networks to provide full navigation control or at-least lateral steering control for autonomous vehicles, the foremost among them being the ALVINN project [2]. The neural network employed was a three layer feed-forward architecture with a single feedback unit. The input layer was fed in readings from camera pixels and a laser range-finder. This initial road-following controller paved the way for the ALVINN-VC [10] which was a more complete road-navigation system capable of dealing with junctions and intersections.

One of the key challenges of the project was to provide data for the back-propagation algorithm to train the network. Since in the case of road-following training on the basis of real-world conditions to account for all the variations in the road non-road environment would be logistically impossible, great effort was taken to create a simulated road-generator which would supply images based on the variations of as many as 200 parameters. Later trials involved training the network on sensor and motor inputs of on an actual human driver at the helm. The main issue with the back-propagation approach to learning in general is over-fitting to the training data and thus rendering the system less effective in new un-encountered environments. Moreover there was still a level of human bias manifesting in the choice and generation of the training environment as well as the dictating of what the desired or perfect driving output of the controller should be. Such a control system, trained on human-driving data would never be able outperform a human driving system and its best case scenario is that of matching the human driving. It would also not account for unexpected scenarios such as the ability to recover from steering errors and deviations.

Evolutionary machine learning of neural networks attempts at providing a fresh perspective on the road-following problem, by further reducing this dependency of human-foresight and allowing the AI control system to be in-charge of learning its own feature extraction and control strategies. The authors of [1] who first implemented this approach, made use of a neural network with architecture similar to the one used in [2], but instead of supervised learning the authors used evolutionary computation. Apart from having outputs for controlling mo-

tor actions, the network had a further three outputs which were fed back to the input layer and were capable of influencing the perception of the input image. Instead of having separate modules for action and perception, the paper proposed a unified motor-sensory unit. This model bears similarity to the learning methodologies of biological organisms where functional behaviour is developed through interactions with the environment and a clear link is present between actions and their effect on the perception of the scene.

Thus the aim was to not only evolve a controller capable of road following behaviour, but also the ability to dynamically change its perception of the road as needed in order to be a truly general purpose solution capable of performing across a variety of environments. In addition the computational and hardware requirements of implementing this system are minimal, requiring only a fixed camera and a low power embedded controller. Because of the prohibitive logistics of carrying out the learning on real-world platform, the evolution needed to take place in a simulated environment with the option of later transferring a successfully evolved controller to a real-world platform. As an initial proof of concept the experiment was successful in showing that such controllers can indeed be evolved to successfully carry out road following across a number of simulated environments. However there were limitations with regards to their dynamic colour perception abilities and as a result their performance in certain types of scenes which they had not experienced during evolution. This paper details further progress of solving the road-following problem in simulated environments using this active vision evolutionary robotics approach and aims at addressing the limitations of the previous methodology, techniques to ensure increased robustness and adaptability of the evolved neural networks, as well as further analysing and evaluating their behaviour. It is hypothesized that the strategies outlined in this paper would enable the evolution of controllers which would be capable of ultimately performing in real-world poorly delineated and unstructured roads.

2 Neural Network Controller

A Continuous Time Recurrent Neural Network (CTRNN) is used to control the robot as shown in figure 1. Equations 1, 2, and 3 define the activation values for the 25 input, 6 hidden and 7 output neurons. In these equations, y_i represents the cell-potential, τ_i the decay constant, g the gain factor, I_i the activation of the i^{th} sensor neuron, w_{ji} the weight of synaptic connection from neuron j to neuron i , β_j the bias term and $\sigma(y_j + \beta_j)$ the firing rate. All input neurons share the same bias (β^I); the same being true for output neurons (β^O). $\sigma_x = (1 + e^{-x})^{-1}$ is the sigmoid function. The decay constants, bias terms, weights and gain-factor

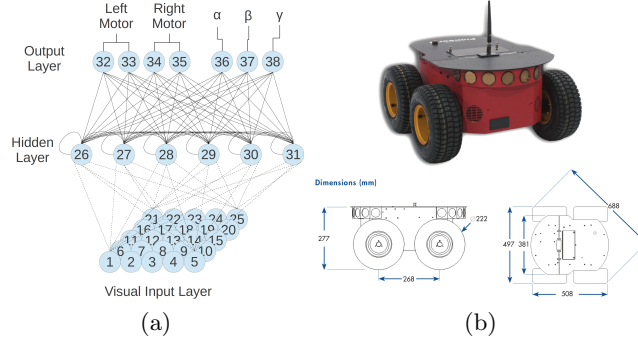


Fig. 1. (a) Architecture of the neural network controller. (b) The Pioneer robot.

are all genetically specified network parameters.

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

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

$$y_i = \sum_{j=26}^{j=31} \omega_{ji} \sigma(y_j + \beta_j); i \in \{32, \dots, 38\} \quad (3)$$

Due to the computational overheads associated with updating neural networks with large input layers, the number of input neurons was limited at 25. The image from the camera is divided into 25 equal-sized blocks. For each block, we compute the averaged red (\bar{R}), green (\bar{G}) and blue (\bar{B}) (i.e., average pixel value). Each block is associated with an input neuron and the final value I_i fed into an input neuron is computed in the following: $I_i = \alpha \bar{R} + \beta \bar{G} + \gamma \bar{B}$. The parameters α , β , and γ are generated by the network at each updating cycle, and normalised in a way that $\alpha + \beta + \gamma = 1$. These parameters give the system its dynamic dimensionality reduction properties. Each output neuron can increase or decrease the magnitude of these parameters to enhance or diminish the colour channel it is associated with, while at the same time having the opposite effect on the other two channels. For example, in an environment where red is the channel which shows contrast between road and non-road, having α at a maximum and the other channels at a minimum would enable the network to be presented with the best possible contrast from the scene.

The motion control is based on the 2D two-wheeled differential drive kinematics model for mobile robots detailed in [12]. This model takes into account the robots structural parameters i.e. radius, wheel distance and speed-limits to give an output in terms of the robots updated position and orientation. The output of neuron 32 to 35 (Figure 1) are used to set the the left and the right wheel speeds. Complex dynamical properties such as friction are not accounted for in

this model. The author in [5] highlights examples of the successful portability of this model from simulated to real-world platforms.

3 Genetic Algorithm

A population size of 52 individual chromosomes is used, with a generational limit of 3000. Trials involved the network controllers trying to perform road-following in either six or twelve simulated environments (Section ??). The best individual of each generation is guaranteed a place in the next generation, whereas the one which performed the worst is truncated and made unavailable for breeding. The rest 51 individuals of the new generation are generated by breeding with the parent chromosomes selected using the roulette-wheel method. Crossover and mutation probabilities are set at 50% and 5% respectively. These operators remain static and non-adaptive throughout the evolution. Carrying out this process of artificial evolution over 3000 generations in a sequential process would mean an unreasonably high training time. Thus the genetic algorithm is parallelized using MPI and implemented on the HPC Wales computing cluster. Each individual run its evaluations as a separate process and the respective fitness values are communicated to a root process which in turn carries out the evolution, generating the new generation of controllers.

4 Simulation Scenes

The evaluation scenes are the virtual environment where each solution (i.e., chromosome) is evaluated. These scenes form the basis for the networks learning process, and the importance of this aspect needs to be stressed. These scenes have been designed to facilitate the evolution of dynamic colour perception strategies (i.e., the adaptive variation of α , β , and γ). The evolution scene graphics are rendered using OpenGL and are designed to simulate a camera pointing down at the ground such that the road and surroundings on either side are visible till a vanishing point further away.

The road is rendered using a modified version of the road generation algorithm employed in [1]. A total of 11 tiles are used each 160 cm long and 100 cm wide. The length of the road the robot need to travel is 17.6 metres. The virtual robot model has a diameter of approximately 54 cm. The road starts off with a smooth bend; each tile rotated 30° left or right. The direction of this turn alternates for consecutive trials. This is followed by a similar smooth bend, with greater probability ($6/7$) of it being in the opposite direction as the first one. This provision allows an agent 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. The scene in each trial varies in terms of the colour of the road and non-road surfaces as shown in Table 1. These scenes are created such that no contrast can be perceived between the road and non-road surfaces unless the robot is able to vary the value of α , β ,

Table 1. Colour combinations of the twelve evaluation scenes.

Scene	Road	Non Road	Random (Noise)
1	Bright Blue	Dark Blue	Red and Green
2	Bright Green	Dark Green	Blue and Red
3	Bright Red	Dark Red	Blue and Green
4	Bright Red, Dark Green	Dark Red, Bright Green	Blue
5	Bright Blue, Dark Red	Dark Blue, Bright Red	Green
6	Bright Green, Dark Blue	Dark Green, Bright Blue	Red
7	Dark Blue	Bright Blue	Red and Green
8	Dark Green	Bright Green	Blue and Red
9	Dark Red	Bright Red	Blue and Green
10	Dark Red, Bright Green	Bright Red, Dark Green	Blue
11	Dark Blue, Bright Red	Bright Blue, Dark Red	Green
12	Dark Green, Bright Blue	Bright Green, Dark Blue	Red

and γ in an adaptive way. The 12 scenes can appear in three different format, which differs in terms of the intensity difference between the dark and the bright colours (see Table 2).

To simulate the effect of poorly delineated roads, the edges of the textures were blended together such that there would not be a clear demarcating line between the road and non-road areas. An additional road-tile with higher levels of delineation and uneven geometry was created to be used in the testing period to assess the robustness of the evolved controllers.

5 Road Bounds Checking and Fitness Function

Each trial was allowed a maximum of 250 iterations with a check being carried out after the end of each iteration (update) to see if the robot was still on the road. If the robot was detected to have moved off the road, the trial was terminated. At the end of each trial the value of the distance travelled was calculated by the number of road-tiles traversed thus far and the position in the current tile. In case of the trial being terminated due to the agent going off the road, the current score value was divided by 5, to make the contribution of progress in the current tile negligible. This distance value $d(i)$ for each evaluation was further normalized to the range of 0.5-1.0 to present the final product, which would be in the powers of twelve or six, in an acceptable range. The

Table 2. Contrast and colour distribution characteristics for the three sets of scenes.

Set	Contrast between mean intensities of road and non-road (0 - 255)	Range of distribution of intensities (0 - 255)
A	120 for all scenes	120 for all scenes
B	150 for mono-colour, 120 for dual-colour	10 for mono-colour, 30 for dual-colour
C	80 for all scenes	80 for all scenes

final fitness function (Equation 4) comprised of two components multiplied with each other, the product of distance values of each evaluation and the other a colour term Δ . In initial experiments, it was observed that best individuals in early stages of evolution were able to solve only a subset of the set of 12 scenes. However, these individuals dominated the population over generations, resulting in a local maxima wherein the ability to solve the other scenes did not evolve. This happened in the case when the fitness was determined simply by the average distance value across all the trials. Thus having the fitness comprised of the individual distance values multiplied with each other ensured that such skewed solutions could not dominate the population disproportionately and only individuals which perform consistently well in all the scenes would be rewarded. Furthermore the Δ term was introduced to further aid or guide the final solution in the correct direction by rewarding the correct activation of the colour outputs in each of the evaluation scenes. Populations initialized with the same random seed were tested in evolutionary runs with and without this colour term Δ to study its effect, and successful evolution was observed only in those runs where it was included.

$$F = \Delta_{final} \times \frac{1}{E} \prod_{e=1}^E (0.5 + (\frac{d(e)}{22})); \quad (4)$$

$$d(e) = NT + CS \quad (5)$$

$$CS = TL - \mu; \quad (6)$$

$$\Delta_{final} = \frac{1}{E} \sum_{t=1}^{t=E} C(e); \quad (7)$$

$$C_{1,2,3,7,8,9} = \sum_{s=50}^{s=S} |OR_s - OW_s^1| + |OR_s - OW_s^2| \quad (8)$$

$$C_{4,5,6,10,11,12} = \sum_{s=50}^{s=S} 2 \times OW_s \quad (9)$$

with $E = 12$ being the total number of trial; NT equal to the number of tiles crossed; CS equal to the score on the current tile; TL equal to the tile length; μ equal to the error vector from mid-point of the tile the the robot current position; C_e corresponding to the quality of the dynamic color perception strategy in trial e ; OR_s being the value of the colour parameters (i.e., α , β , or γ) that has to be used to discriminate between road and non-road; OW_s^1 and OW_s^2 being the values of the colour parameters (i.e., a combination of α , β , and γ) that do not discriminate between road and non-road in mono-colour scenes. OW_s being the value of the colour parameters (i.e., α , β , or γ) that does not discriminate between road and non-road in dual-colour scenes.

A final effect of Δ_{final} was that since it was calculated only after the 50th iteration to allow the agent time to settle on a steady sequence of colour output values for the trial, any agents leaving the road before the 50th iteration would get a 0 for the colour score of that trial. Thus agents which left the road before

the 50th iteration for all the trials would receive a 0 as the final fitness irrespective of any distance values gained.

6 Results and Observations

The first round of evolutionary runs was done with six scenes, three basic mono-colour (1, 2, 3) and three dual-colour (4, 5, 6) scenes. The scenes were chosen from Set *A*, thus having an average contrast of 120 (on the scale of 0-255) and the colour distributions having an approximate deviation of 60 on either side of the mean. Based on the results of this stage, the experiment was extended to all 12 scenes with colour distributions from Sets *A*, *B* and *C*. Scenes from Set *B* had the most inherent contrast while those in Set *C* were the most challenging in this aspect. A set of 10 seeds was used to run a total of 40 experiments; 10 with 6 scenes using Set *A* and 30 with 12 scenes using scenes from Set *A*, *B* and *C*. Due to the nature of genetic algorithms and the complexity of the problem, not all seeds were able to evolve successfully, i.e. output fitness values were not high enough to indicate their ability to solve atleast half of the evaluation scenes they were evolved in. Only those runs which had values higher than this threshold were selected for subsequent rounds of testing and evaluation.

6.1 Testing Round 1

In this first testing round the best individuals from the last 500 generations of eleven successful runs were subject to a uniform set of eight road shapes in each of the twelve scenes with the colour distributions being the same as what they were evolved in. The road shapes consisted of two basic types, an “S” shaped course where the robot needed to make turns in both directions to reach the end and the other where there was a constant turn in one direction followed by a straightening of the path. Each of these was generated twice with initial left and right turns across two different angles (20° and 30°) dictating the curvature of the turns, thus giving the total of 8 possible road shapes. During evolution the angle of curvature was always 30° and the road generation algorithm ensured that the overwhelming majority (6 out of 7) of shapes generated would be of the first “S” shaped type. The rationale behind generating this fixed set of roads was to ascertain the best performing individuals in the population, as it was possible that some controllers which obtained high fitness values could have simply been lucky and not really possess the ability to navigate multiple road shapes across all the environments. Moreover, it would provide a break-up of the controllers’ performance in each of the scenes and match this against their dynamic colour perception strategies. The agents were rewarded on the basis of a normalized distance score, which assesses their progress along a 15 tile course (24 metres) with agents reaching the end of the last tile getting the highest score of 10.

From the test results (Figure ??) of the agents evolved in six scenes, three seeds were successful at evolving agents which could solve not only the six basic mono-colour scenes (road always brighter than non-road), but also six of the

dual-colour scenes including scenes 10, 11 and 12 which they had not experienced during evolution. This is testament to the generality and adaptability of the solutions evolved. Not surprisingly they failed in the three reversed mono-colour scenes as the entire basis of their learning was dependent on the road being brighter than the non-road. However agent S3 actually managed to partially solve the scene where blue on the road was darker than the non-road, with an average normalized score of 4.22. This was because it was correct in identifying the feature-differentiating channel (blue), but could not process the negative contrast for too long. On investigating the dynamic colour perception strategies of these agents it was observed that the colour outputs for the three mono-colour scenes (as expected from their Δ values being in the range of 1.4-1.8) were more or less steady and above 0.85 throughout the trials. Only in periods where sharp turns or course-corrections needed to be made was there a period of oscillation between the outputs, where each of them would alternate between approximately 0-0.9 every 2 time steps.

The inclusion of the mono colour scenes and the colour term Δ ensured that an adaptive strategy where all three colour outputs are utilized was developed, and it was this strategy that enabled the network to solve all combinations of textures in the dual-colour scenes. It could be argued that including Δ in the fitness function was in a sense dictating a solution to the controllers, rather than truly allowing them to evolve their own strategy. However as seen from the results and during evolution, it was indeed a necessary inclusion and the network did not completely adopt this enforced strategy as the presence of the periods of oscillation between the colour three outputs suggests. It is interesting to note that the motion in terms of dynamics was smoother and quicker when the correct colour output was constantly at a high value (≈ 0.9) while during the oscillating phases constant course-corrections were being made, taking more time to complete the same distance.

The results of the twelve-scene experiments (Figure ??) were not as uniform, with solutions showing greater variability in terms of seed and the colour distribution set they were evolved in. The majority of solutions (like S5 and S6) only evolved the ability to dynamically vary two of their three colour outputs, and simply did not use the third. Which one of the three this unused output was, varied from seed to seed, meaning two out of the six mono-colour scenes (basic and reversed) could not be solved. However these agents did manage to solve all six dual-colour scenes because as mentioned earlier having the ability to dynamically vary only two colour outputs would be sufficient in these cases. In fact a few of these agents were tested in more challenging road shapes (curvature of 35°) and across scenes where the average contrast was 90, less than what they had been exposed to during evolution. They were indeed successful in solving all the scenes except the two mono-colour ones whose corresponding output they had not evolved to vary. In terms of colour-outputs, of the two that were being varied, only one (depending on the seed) displayed perfect behaviour of having a constantly high activation value. In other scenes, the agents relied

on the oscillating behaviour to solve them, except the oscillation now took place between only the two outputs that were being used.

Only two solutions successfully evolved to show capability of solving all twelve scenes. Of these S4 evolved in scenes with colour distribution of Set *B* and S7 with distribution values of Set *A*. It is interesting to note the effect of these distribution values on the evolved solutions, as the seed for S4 when evolved with distributions of Set *A* could develop only a sub-par solution where the agents could not navigate the green mono-colour scenes. The seed for S7 when evolved in Set *B*, which could be said to be a less challenging environment, could only solve two scenes. Also unsurprisingly none of these seeds when tried with Set *C* could produce any successful solutions, as the contrast values were much lower and the distributions themselves were more spread out across the spectrum.

Agents of S7, developed a strategy wherein their ability to differentiate on basis of the green channel was much more enhanced than the other two channels. The β output was constant and near maximum for all scenes where bright green could be made the differentiating channel. For all other scenes the oscillatory behaviour described earlier was observed, although for scene 12 (blue and green), the oscillation took place only between the two relevant channels despite the possessed ability to vary the α term. While the agents did demonstrate the ability to traverse scenes 1 and 2, the navigation was slower and often error-prone at the beginning contributing to the lower average scores. S4 evolved agents where the α , β and γ terms were near maximum for the majority of the time for scenes 3, 8 and 9 respectively, while being a mixture of stable and oscillatory for the rest.

6.2 Testing Round 2

Four agents, two each from the two best six-scene and two best twelve-scene runs, were then chosen to be subject to a further round of testing where the aim was to investigate the robustness and generality of their road-following strategies by observing their behaviour in environments they had not encountered during the evolution phase. The twelve scenes were recreated with a colour distribution having average contrast of 90 and deviations from mean of around 40 (on a scale of 0-255). In each of these scenes, the range of distribution of the random noise channels was set at 0-0.80 for one case and 0-0.25 in another. In the evolutionary runs, the distribution of the random noise channels always varied from 0-1 with uniform probability. However it was observed that narrowing this range to less than 0.5 during the testing phase caused a few randomly selected agents to fail (even though they had successfully solved similar scenes during evolution) and thus it was decided to also evaluate the agents against this parameter. In theory agents with the correct feature extraction strategy would be able to completely discard the random channels, as despite the range of the distribution it had no contribution towards highlighting the desired features. The road was set to be of the ‘‘S’’ shaped type with an angle of curvature of 25° in both left and right initial starting directions, and each of these shapes was generated twice, giving a

total of 4 trials for each random noise distribution value for each of the 24 scenes. Thus each agent in this second round of testing was evaluated over 96 trials. In order to further enhance the effect of presenting an unfamiliar environment to the controllers, the road tile used in this testing phase represented a more delineated and unstructured course, having a maximum width of 110 cm at places but with only 85-90 cm consistently visible throughout.

The results of this second round of testing (Figure ??) showed that agents of S4 (twelve scenes) and S7 (twelve scenes) had developed the most robust and general-purpose solution. Despite receiving lower scores (below 7) for a few scenes, only these solutions had the capability of solving all twelve scenes across all the evaluation parameters, i.e all road shapes with reduced contrast and varying random noise values. The performance of S1 (six scenes) in identifying features in the blue or γ channel was affected by the reduced contrast in the colour distribution. This in turn not only meant failure in the corresponding mono-colour scenes but also in the two dual-colour scenes where the blue channel was brighter on the road. The agent could still use the other two channels successfully across both ranges of the random noise variation. The agent was later tested in a scene with average contrast for the blue channel at 109 (still a new environment), and in this case it was able to navigate the corresponding scenes successfully. While agent S2 was able to solve all nine scenes (excluding reversed mono-coloured) when the random noise was in the range of 0-0.80, it failed to differentiate on the basis of both blue and green channels when this range was reduced to 0-0.25 thus resulting in lower average scores for scenes 1, 2, 6 and 12. The agents inability to perform in these scenes was due to it constantly relying on the red channel for contrast, activating only the α term throughout the trials. It incorrectly associated the low distribution range of red values with the availability of features, causing it to fail in scenes where these numbers were random.

For the two successful agents in this round, it can be seen that in both cases performance in all but one mono-colour scene deteriorated compared to the earlier round of testing. They were still capable reaching the end of the road in these scenes, but with less consistency compared to the earlier tests contributing to the lower overall score. While scores in dual-colour scenes still remained close to 9, the one exception to this was S7 in scene 5 where the agent suffered a number of early failures, attributed to the absence of relevant features in the green channel which it was most sensitive to. Interestingly despite being subject to higher contrasts than S7 during evolution, S4 was still able to match or exceed its performance in eleven out of twelve scenes. Moreover it was observed that the agents changed their colour perception strategies compared to the previous higher-contrast environment, increasingly relying on oscillating the activations of the colour output neurons. However when the one channel they were most sensitive to was available, they used it exclusively by activating only the associated output neuron for the majority of the trial.

7 Conclusions

The methodology described in this paper was successful in evolving neural networks capable of demonstrating road-following by dynamic dimensionality reduction in a variety of challenging simulated environments. This new set of controllers have shown improvement in the dynamic colour perceptions abilities compared to those evolved earlier in [1], with the capability to now recognize features based on negative and positive contrast in all three primary colour channels used. This work is a significant step towards the hardware implementation of these controllers, as real-world environments would in majority consist of colour combinations similar to those present in the simulated scenes. However it is acknowledged that the contrasts between road and non-road surfaces would be lower than what the networks were tested on. This is proposed to be mitigated by introducing a simple contrast stretching step before the processing of the inputs. Future work would also focus on representing the environment in terms of alternate colour models such as HSV, instead of the traditional RGB model used thus far. Besides this, there is a need to increase the robustness of these controllers by minimizing the disparity in the feature extraction capabilities across the three channels. On the whole however, the findings of this paper strengthen the potential of using these controllers as a viable alternative road-following solution and further efforts would focus on transferring these evolved controllers to a mobile robotic platform.

References

1. Clarke, S., Labrosse, F., Trianni, V., Tuci, E.: An evolutionary approach to road following: a simulated case study. In: 12th European Conference on Artificial Life, pp. 1017–1024. MIT Press, Taormina (2013)
2. Pomerleau, D., Neural network vision for robot driving, Intelligent Unmanned Ground Vehicles, pp 1–22 (1997)
3. Sotelo, M., Rodriguez, F., Magdalena, L., Bergassa, L., Boquete, L.: A Color Vision-Based Lane Tracking System for Autonomous Driving on Unmarked Roads. *Autonomous Robots* 16-1, 95–116 (2004)
4. Suzuki, M., Floreano, D.: The contribution of active body movement to visual development in evolutionary robots. *Neural Networks* 18, 656–665 (2005)
5. Clarke, S.: Thesis: An Evolutionary Approach to Road Following. Aberystwyth University, Aberystwyth (2012).
6. Bown, Oliver and Lexer, Sebastian: Continuous-Time Recurrent Neural Networks for Generative and Interactive Musical Performance. In: Rothlauf, Franz and Branke, Jrgen and Cagnoni, Stefano and Costa, Ernesto and Cotta, Carlos and Drechsler, Rolf and Lutton, Evelyne and Machado, Penousal and Moore, JasonH. and Romero, Juan and Smith, GeorgeD. and Squillero, Giovanni and Takagi, Hideyuki (Eds.), *Applications of Evolutionary Computing*. LNCS, vol. 3907, pp. 652–663. Springer Berlin Heidelberg (2006)
7. Ososinski, M., Labrosse, F.: Real-Time Autonomous Colour-Based Following of Ill-Defined Roads. In: Herrmann, G., Studley, M., Pearson, M., Conn, A., Melhuish, C., Witkowski, M., Kim, J.-H., Vadakkepat, P. (Eds.), *Advances in Autonomous Robotics*. LNCS, vol. 7429, pp. 336–376. Springer (2012)

8. Ramström, O., and Christensen, H.: A method for following of unmarked roads. IEEE Intelligent Vehicles Symposium Proceedings. IEEE (2005)
9. Tuci, E., Harvey, I., Todd, P.: Using a net to catch a mate: evolving CTRNNs for the dowry problem. In: From, Animals to Animats: Proceedings of the Seventh international conference on simulation of adaptive behavior, pp. 1017–1024. MIT Press, Edinburgh (2002)
10. Pomerleau, D.A., Thorpe, C.E.: Vision-based Neural Network Road and Intersection Detection and Traversal. In: Proceedings of the International Conference on Intelligent Robots and Systems-Volume 3, pp. 3344. IEEE (1995)
11. Montemerlo, M., Thrun, S., Dahlkamp, H., Stavens, D.: Winning the DARPA Grand Challenge with an AI robot. In: Proceedings of the AAAI National Conference on Artificial Intelligence, pp. 17–20. AAAI Press (2006)
12. Dudek, G., Jenkin, M.: Computational Principles of Mobile Robotics. Cambridge University Press (2000). ISBN 0521568765.