

Bee-Inspired Landmark Recognition in Robotic Navigation

Kodi Cumbo
Dept. of Computer Science
University of Idaho
Moscow, ID 83844 USA
cumb0597 *

Travis DeVault
Dept. of Computer Science
University of Idaho
Moscow, ID 83844 USA
zerill@gmail.com

Samantha Heck
Dept. of Computer Science
University of Idaho
Moscow, ID 83844 USA
heck9873

Robert Heckendorn
Dept. of Computer Science
University of Idaho
Moscow, ID 83844 USA
heckendo@uidaho.edu

Ian Tanimoto
Dept. of Computer Science
University of Idaho
Moscow, ID 83844 USA
tani0633

Terence Soule
Dept. of Computer Science
University of Idaho
Moscow, ID 83844 USA
tsoule@uidaho.edu

ABSTRACT

Research has shown that honeybees have a remarkable ability to use landmarks to travel miles to find — and return to — a food source [1, 2]. The long-term goal of this research is to use physical robots, incorporating a variety of image processing approaches, to examine this ability. The results may also improve our ability to develop effective landmark guidance systems for robots.

This research examines whether robots using a simple genetic algorithm (GA) and neural network (NN) that are trained to search for a target object adopt the use of landmarks to aid in the search — even when landmarks are not explicitly considered. This is a first step towards more complex experiments in less controlled environments. The simplified environment used in these experiments is controllable, making it easier to determine and understand the robot's behavior before moving to a more complex external environment. The robot uses a feed-forward, episodic NN for navigation — there is no inherent use of a memory bank of images. It is trained using Learning from Demonstration (LfD) to follow a simple search pattern independent of landmarks, but visual cues that could act as landmarks are present in the environment. Our results show that even under these conditions visual cues, i.e. landmarks, are incorporated in the learned search.

CCS Concepts

•Computing methodologies → Reinforcement learning; •Computer systems organization → Evolutionary robotics;

* Authors 1, 2, and 3 emails: @vandals.uidaho.edu

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

GECCO'16 Companion, July 20-24, 2016, Denver, CO, USA

© 2016 ACM. ISBN 978-1-4503-4323-7/16/07...\$15.00

DOI: <http://dx.doi.org/10.1145/2908961.2931669>

Keywords

Evolutionary robotics, neural networks, navigation

1. METHODS

This experiment trains a robot to find a target near a set of potential landmarks in a controlled environment. The robot is then tested in the same environment under four conditions: with the target and landmarks present, with only the target present, with only the landmarks present, and with both the landmarks and the target present, but the target moved.

The experiment environment consists of a 2.4m × 2.4m arena surrounded by white walls to allow control over the visual environment. Blue tape was placed on the top and bottom of the walls to mark its extent for the robot. The target object was a green tower approximately 10cm × 10cm × 16cm placed approximately 36cm from one corner of the arena. A 46cm × 30cm piece of red paper was placed on one wall approximately 45cm to one side of the target and a 46cm × 30cm piece of blue paper was placed approximately 45cm from the other side of the target. Note that when the robot approached within approximately 75cm of the target these pieces of paper were no longer in the field of view. These colored pieces of paper represent potential landmarks that the robot may learn to use to navigate. They are referred to as landmarks in the rest of the paper, but they were never explicitly treated as such — they are simply potential background cues.

In the demonstration phase, the robot is placed in each of four different starting locations; from each position, the robot is steered to the left until the target object is centered in the field of view and then driven toward the target until within 3cm of it, at which point the robot is moved to the next starting position. Trainers focus only on the target and ignore potential landmarks.

Evolution is done on-line. Evolution begins after the robot has received at least one demonstration each of a forward, a left, and a right command. Once evolution has begun each new command from the trainer creates a new training case that is added to the set of training cases followed by 5 generations of evolution. The training process typically resulted in 100 training cases and 350 generations (because evolution doesn't begin until the robot has received at least one demonstration each of a forward, a left, and a right com-

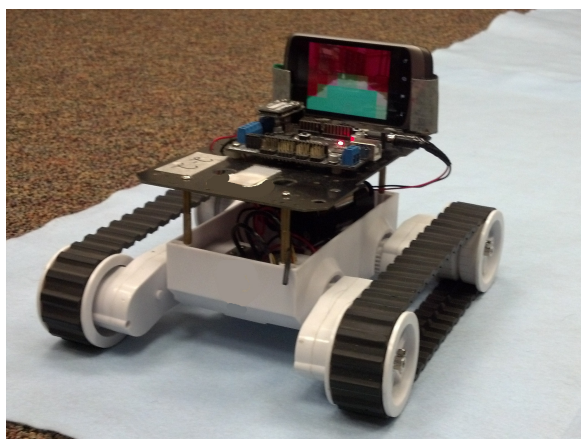


Figure 1: The robot. The smart phone receives wireless commands from the user and forwards the commands via bluetooth to an Arduino type microcontroller that controls the robot's motors. The smart phone can perform image processing and on-board learning.

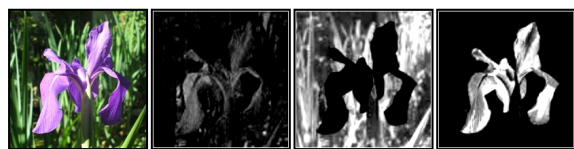


Figure 2: Sample image with the Red, Green, and Blue components enhanced.

mand many of the early training cases are collected before evolution begins).

1.1 The Research Robots

The robots used in these experiments were designed and built at the University of Idaho following Commodity Off the Shelf (COTS) design principles [3]. The robots consist of three basic components: an Android smart phone, an Arduino microcontroller, and a robot platform. The complete robot¹ is shown in Figure 1.

The robot uses two states during this experiment:

Learning from Demonstration: In this state the robot receives drive instructions from a human operator with a remote control (a second phone). When the robot receives an instruction, it creates a training case. Each training case is the image the robot sees and the action taken. The commands that the trainer has access to are: forward, left, and right. Each command results in a discrete action, e.g. forward causes the robot to move forward approximately 6cm.

Autonomous Operation: In this state the robot does not receive commands from the trainer. The robot evaluates the current image from the smartphone's camera and uses an evolved neural network (NN) to make a decision.

1.2 Image Processing

The input the neural network receives is processed visual input from the phone's camera. The camera's image is divided into 40 cells in an 8×5 grid. For each cell the relative intensity of red, green, and blue is calculated and becomes

¹Rover 5 Tank chassis <http://www.dfrobot.com>

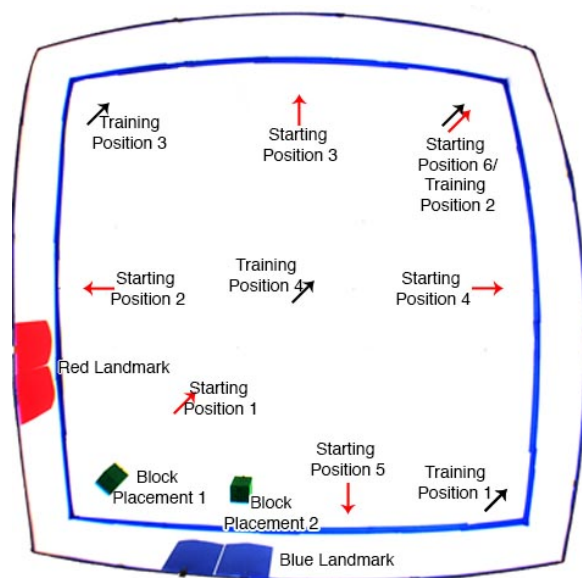


Figure 3: This is a labeled picture showing the starting positions and directions used in both training and testing. It was taken by the overhead camera used for data collection. The camera uses a fisheye lens in order to capture the entire arena, which causes some distortion.

an input to the NN. Figure 2 shows how an enhanced image's primary color components stand out.

1.3 The Genetic Algorithm

1.3.1 Neural Network Architecture

The neural network has 120 input nodes, a hidden layer with 5 nodes, and an output layer with 3 nodes. The inputs come from the processed image received from the smart phone camera. The 3 color components from each of the 40 cells are used as individual inputs in the neural network.

The neural network has three output nodes, one for forward, left, and right. The robot always takes the action corresponding to the highest output. The fitness function is designed to favor separating the highest output value (if it is the correct action) of the NN from the other lower values.

1.3.2 The Evolutionary Algorithm

A generational genetic algorithm (GA) was used to evolve the neural network weights. Individuals encode the weights of a complete multilayer perceptron neural network; each individual has 623 weights (120 input nodes×5 hidden nodes = 600, and 5 hidden nodes to 3 output nodes = 15 + 8 bias nodes). Initially the weights were randomly set in the range -1.0 to 1.0. An elitism of one is used.

Fitness was determined by comparing the correct (trainer supplied) action on each of the training cases to the output of the NN on the training cases. The training cases are categorized based on the correct action (left, right, and forward). The network can earn an equal amount of fitness for getting all the cases in a category correct, regardless of the number of cases within the category, i.e. the fitness value of each case is normalized by the number of other cases with the same action. The other parameters of the evolutionary algorithm are summarized in Table 1. This simple GA was

Population size	40
Generations	300
Mutation rate	.2
Crossover rate	100% Uniform
Selection method	Tournament(size 3)

Table 1: Summary of the evolutionary algorithm.

used to determine if a general GA and neural network are sufficient to learn to use landmarks.

1.4 Testing

The evolved neural networks are tested under four different conditions (described below). For each test the robot is placed in one of 6 different starting positions, facing away from the target object (Figure 3). Starting locations 1 - 5 differ from those used during training; starting position 6 is the same as used in training. The movement of the robot during testing is recorded using an overhead camera that records the robot's (x,y) coordinates and the motion tracks are presented in the results. *A test was halted either when the robot was within 3cm of the target or one minute passed.*

The four test cases:

- Test 1: target and landmarks are both present and in the same positions as during training – used to determine how well the robot learned the task.
- Test 2: only the target is present, in the same position as during training – used to determine whether the robot learned the target object.
- Test 3: only the landmarks are present, in the same positions as during training – used to determine whether the robot learned to use the landmarks.
- Test 4: target and landmarks are both present, but the target has been shifted by approximately 75cm - used to determine how the robot uses the combination of target and landmarks.

2. RESULTS

Figure 4 shows the paths the robot took during testing. If the robot found the target (i.e. moved to within 3cm of it), the test was terminated. Otherwise, the robot continued to move for one minute before being stopped.

In every starting position in Test 1 (Figure 4a) the robot turns until it sees the target or a landmark and then approaches it, eventually refocusing on the target and reaching it.

In Test 2 (Figure 4b), the robot only locates the target from starting position 1, the only position in which the landmarks would not be in the robot's field of view. From the other positions, the robot turns left for the duration of the test. Most movement away from the starting position is a result of drifting caused by differences in the treads and motor speeds of the robot. Trial 2 exhibits behavior in position 6 where it occasionally orients on the corner. In Trial 5, the robot would sometimes move forward. These cases suggest that the robot sometimes learns to use the blue tape around the border of the arena as a landmark.

In Test 3 (Figure 4c), the robot goes to the corner of the arena and then begins searching for the target. The robot

tends to approach the first landmark that becomes visible, and then turns to search the immediate area. The robot does not leave the corner after reaching it, instead searching until the test ends. Trial 2 again exhibits an additional fixation: after reaching the corner, the robot remains in that position, rather than searching the rest of the area.

In Test 4 (Figure 4d), initial behavior resembles Test 3: the robot approaches the corner. Then the robot begins to search the immediate area. If the target becomes visible, the robot approaches it.

2.1 Analysis

The robot learned to approach the corner with the landmarks and target from novel, untrained positions. Once the robots gets close to the corner, the robot finds and approaches the target.

Test 2 shows that the trained neural network doesn't find the target from a distance without the landmarks. The target is smaller than the landmarks, so the robot does not learn to react to it from far away. Test 3 shows that the robot has a strong preference for the corner with the landmarks, even with the target not present — cuing off the landmarks to stay within a small distance from where the target should be. Test 4 shows that, if the target becomes visible, the robot prioritizes it over other visual indicators. These behaviors closely mimic those observed in bees [2].

3. CONCLUSIONS

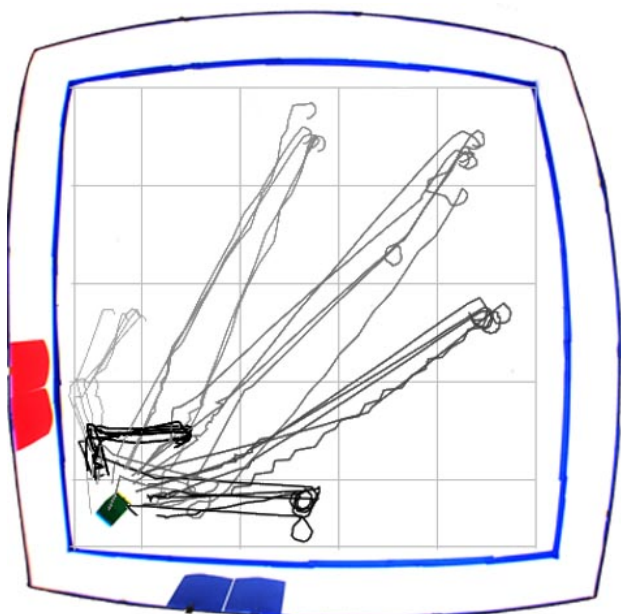
This research shows that a NN trained with LfD can learn to use landmarks even when they are not explicitly part of the training process. Where the visual processing of the robot makes it difficult for the robot to differentiate target objects from a distance, evolution takes advantage of larger background objects to allow the robot to locate the area the target is in. When the robot is close enough to easily notice the target object, the target takes precedence over background cues, even when they are contradictory. The next step is to examine the behavior of robots trained to find a target with and without intermediate target landmarks in an noisy outside environment more similar to those typically encountered by bees.

4. ACKNOWLEDGMENTS

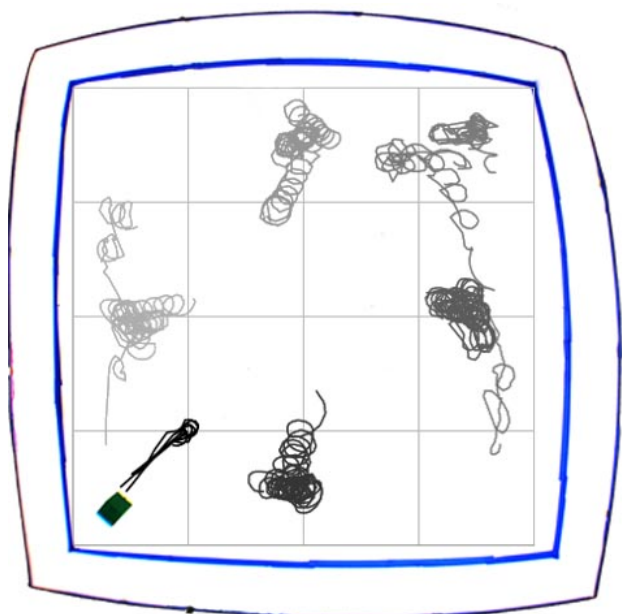
This material is based in part upon work supported by the National Science Foundation under Cooperative Agreement No. DBI-0939454. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

5. REFERENCES

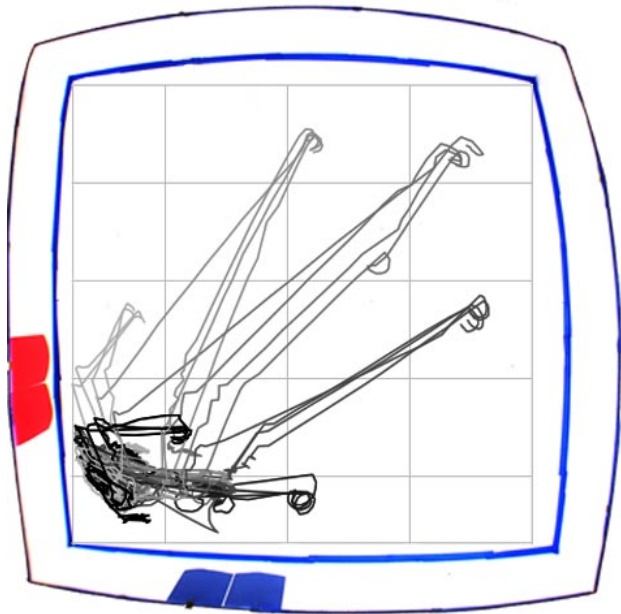
- [1] J. E. Eckert. The flight range of the honeybee. *Journal of Agricultural Research*, 47:257–285, 1933.
- [2] F. N. B. III. *How Honey Bees use Visual Landmarks during Goal-directed Navigation: Wayfinding Strategies as Sequential Decision Making Processes*. PhD thesis, Michigan State University, 2006.
- [3] T. Soule and R. B. Heckendorn. Cotsbots: computationally powerful, low-cost robots for computer science curriculums. *Journal of Computing Sciences in Colleges*, 27(1):180–187, 2011.



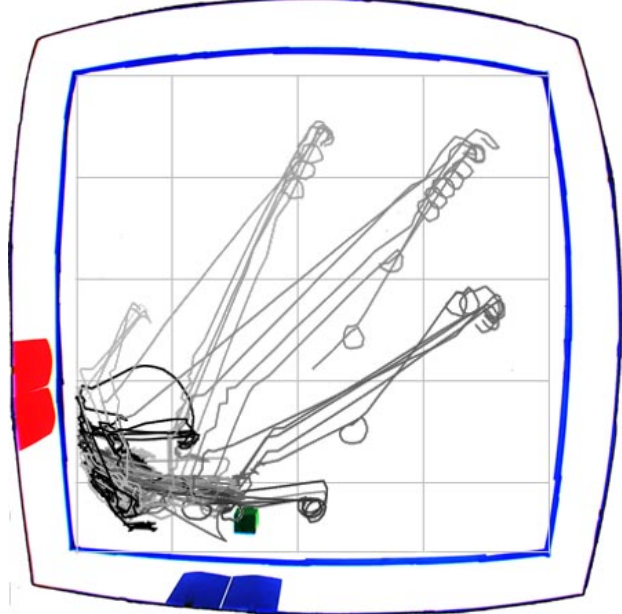
(a) Target and landmarks.



(b) Target only.



(c) Landmarks only.



(d) Landmarks and shifted target.

Figure 4: (a) The robot consistently finds the target, often using the landmarks for guidance. (b) Without landmarks the robot only finds the target when starting very close to it. (c) Without the target the robot uses the landmarks to find the right area. (d) The robot can usually find a shifted target by using the landmarks to reach the correct area.