AUTONOMOUS NAVIGATION USING VISUAL LANDMARKS

Cameron Morland
Undergraduate Student, Mechanical Engineering, Mechatronics Option
University of Waterloo
Email: cameron@morland.ca

ABSTRACT

A method whereby an autonomous robot can navigate in a known, unmodified environment is presented. The method uses visual landmarks, matching previously stored “snapshots” of the environment to recognise its location. This landmark detection method, augmented with other systems such as ultrasonics and a memory of previous location, permits reliable location in an environment. By performing simple actions such as turning and following walls, navigation is made possible. The Tao 7 experimental wheelchair project, using the concepts developed in this paper, is described. The wheelchair demonstrates that these simple behaviours can generate a creature capable of navigating in an unmodified environment. Problems encountered and possible resolutions are considered.
Revision: 2.4

1 INTRODUCTION

This paper examines a means whereby a mobile robot can automatically and unambiguously determine its position in a known environment, using a system composed of behaviours that give ambiguous results. This landmark detection permits the robot to navigate using actions that link together points on a map.

This work is based loosely on that described by Edwards [1]. It extends previous navigation work by relying to a large extent on uncertain visual landmark detection.

1.1 BACKGROUND

The Behaviour Based approach to machine intelligence is the basis for the work described in this document. Behaviour Based robotics was developed by Rodney Brooks [2]; in this design philosophy, the Subsumption Architecture permits a handful of very simple behaviours to guide the robot. These behaviours link sensory inputs to outputs, using no more processing than simple finite-state machines, augmented with timing elements [3].

Much work has been done developing simple behaviours permitting the robot to avoid collisions with static or moving obstacles and to wander aimlessly around the unmodified environment. The focus of this paper is an additional behaviour: a navigation behaviour, by which the robot can arrive at a given location in the environment. The navigation behaviour relies on previously existing collision avoidance behaviours, but does not modify them except by selectively overriding them.

1.2 Animal Navigation

A number of methods are used by animals to navigate in their environment. It is worth considering them as inspiration, because evolved solutions can often be both effective and simple.

Digger wasps use landmarks to return to the nest after foraging [4, pg 435]. Although the wasp purposely obscures and camou?ages the entrance, it can re-find it by memorizing the location in space of the entrance with respect to nearby landmarks.

Orientation for the purpose of migration is performed by birds using a large number of approaches: star patterns, the magnetic field of the earth, the position of the sun with respect to time of day, and other approaches [4, pg 422].

2 Landmark Detection

The navigation method will require a method to detect the current landmark. How is this to be achieved?

2.1 Visual Landmark Detection

A very simple approach to landmark detection is described by Edwards [1], using first a neural network approach, then an algorithm known as the pattern vector method. Poor recognition was demonstrated by the neural network, so only the pattern vector method will be considered. The detecting is performed in the same way on separate grey and colour data; for clarity, only the grey segment will be discussed.

When a landmark is captured, a very low resolution (eg 8 × 8) image is taken of a region in the middle of the field of view. The grey and colour values of this image are stored in memory. Now when the robot is searching for a match, it compares each "snapshot" with the image currently in view. Figure 1 shows a typical landmark.

Figure 1: A typical landmark

The comparison involves simply taking the absolute difference in intensity for each pixel in the snapshot. These values are given weights proportional to height in the image, as pixels which are higher (and hence closer to the horizon) tend to change less as discussed by Gibson [5].

Once the pixels have been multiplied by their coefficients, they are summed then stored as the error associated with detecting that landmark. A smaller error indicates a closer match. Where a, s, p, and e are the coefficients, stored snapshot, currently-visible picture, and error, respectively:

$$e = \sum_{j=1}^{m} \alpha_j \sum_{i=1}^{n} \left| s_{i,j} - p_{i,j} \right|$$

Where n and m are the width and height of the snapshot, in pixels.

The smallest error for each landmark is measured by comparing the landmark with numerous points in the visible image, "sliding" the comparison across to get a close match. This enlarges the region of space at which the robot can detect each landmark.

The error is calculated for any number of landmarks. Under the Subsumption Architecture each landmark conceptually becomes its own Augmented Finite State Machine (AFSM), so detection is performed in parallel, and hence in constant time.5

3 Autonomous Navigation

With the landmark detection method described in Section 2. the question arises of how to navigate using this information. One achievable method is to join the landmarks together in a topological map, and to program different actions at each landmark which cause the robot to arrive at a specific nearby landmark. By then joining together a sequence of small goals, the ultimate destination can be reached.

3.1 The Map

The map is stored in a data structure isomorphic to a directed graph [6]. Each node corresponds to a landmark; by knowing which landmark the robot came from and which is has arrived at, it can be unambiguously etermined which directed edge of the graph the robot has crossed. In other words, orientation of the robot ollows naturally from the current and previous positions.

Each node contains information about incoming connections, and how to get from each incoming connection o the outgoing connections using an action.

Presently, a hand-generated map is used. Ultimately the robot must also be able to generate and extend he map by exploration.

3.2 Actions

When a landmark is detected, the robot evaluates the map by checking each connection from the current landmark, then checking each connection from that landmark, and so on, determining the topologically shortest route (through the smallest number of landmarks) to the goal. It can then determine the next landmark to head towards. Once the short term goal has been determined, it simply initiates the action at that landmark which links the previous landmark with

5 Actual implementations rarely use separate hardware for each behaviour.

the next landmark. These actions are simple motions, such as "turn left" or "follow a wall on the right." The robot performs these actions for a specified time, then continues wandering. The environment guides it to the next landmark; it will try to move straight, but if constrained between two walls, for example, it will keep itself between them, even if they curve.

4 Improved Landmark Detection

The pattern vector method described in Section 2, although computationally very simple, doesn't work very well. An error of zero almost never occurs, and many false positives are encountered. By itself, the pattern vector method of landmark detection is inadequate.

The first thought to remedy the situation is to improve the complexity of the vision processing algorithm, and to make each landmark contain more information about the image. A higher resolution image could be used, or an attempt could be made to match under graphical operations such as rotate, scale, and skew. However, even a human provided with a high resolution image of two similar landmarks is unable to tell the difference. Figure 2 shows two similar landmarks encountered in a typical office environment. (These were not selected because they are unusually difficult to tell apart; they are real, different landmarks which must be distinguished to navigate successfully in the office.)

Clearly, visual discrimination of such images cannot by done by the robot; it cannot even be done by a human. Although improvements in visual discrimination are quite possible and worth considering, other approaches to landmark detection must be considered.

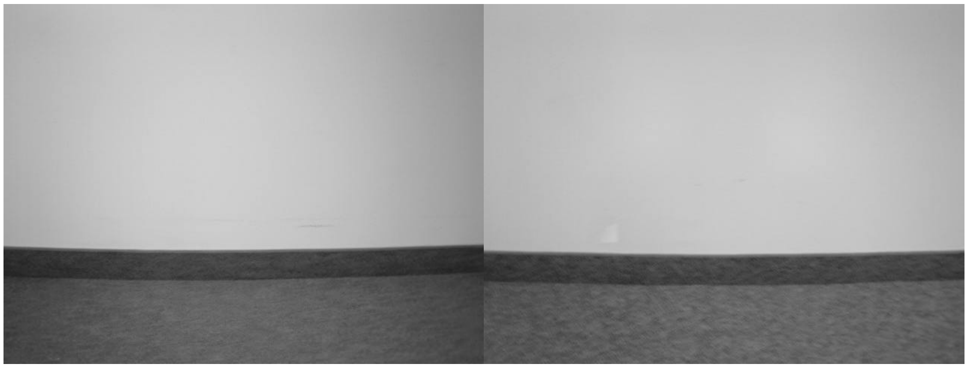

Figure 2: Two similar photos; how can the robot distinguish these locations?

Brooks [7] presents the idea of making easily distinguishable landmarks, such as bar coded signs. This idea is unacceptable to many researchers, including himself, because the robot must be capable of working in a unmodified environment. Some animals are successful at navigating without changing their environment, as discussed in Section 1.2, so this is not an impossible task.

So the problem is that the robot cannot recognise a landmark simply by looking at a photo of it. Therefore, in addition to comparing the image to old photos, we need some other detection circuitry. This can be done in number of distinct ways:

1. Evaluation of connectivity of the map
2. Additional (non-visual) perception
3. Timing between landmarks

Each deserves some attention.

4.1 Map Connectivity Analysis

Given any landmark in the map, only adjacent landmarks can be reached, provided no landmarks are missed. For this reason there is no need to consider matches of any landmarks except for those which are adjacent to the current landmark. This decreases the possibility of a false positive, as only positive results from adjacent landmarks are considered. Four connections is sufficient for most office environments, as shown by Mataric [8]. When the robot is seeking a goal, only positive detection results for the target landmark are considered. Confusion is then only possible if a scene that the robot cannot distinguish from the target is encountered while travelling towards the target.

4.2 Non-Visual Perception

There is no reason that a landmark must solely be a visual cue. Scent is used by dogs and other mammals, honey-bees and others [4, pg 486-490]; numerous air-breathing water creatures, such as amphibians and insects, use gravity for orientation [4, pg 433]. A number of other approaches for detecting landmarks are worth considering.

As shown by Mataric [8], compass readings can be of assistance in detecting landmarks. Even landmarks which are visually very similar but at different angles with respect to north can be distinguished. The images in Figure 2, for example, are at the ends of perpendicular hallways.

Using the Global Position System [9] is not a good solution, as the system is generally unusable indoors.

It would be acceptable for certain applications, including navigation of automobiles, aircraft, and military devices such as autonomous cruise missiles.

Many Subsumption Architecture robots have multiple sensors and types of sensors. It has been shown [8] that navigation can be done entirely using ultrasonic sensors and compass orientation. This was demonstrated by building up confidences in the state of the surroundings: the presence of a wall on either side, a curve, or other features large enough to be detected using ultrasonic sensors.

By considering the state of additional sensors when detecting landmarks, many false detections can be discounted. A behaviour can prevent landmarks from becoming active if the state is incorrect. If, for example, the landmark should have a wall on the left but one cannot be detected, that landmark would not become active.

4.3 Timing

Measuring the time elapsed between landmarks can also be used for verifying correct position. If the trip between landmarks typically takes place in a particular length of time, then if the second landmark is detected a very short time after the first, it can be considered a false match; likewise, if the second landmark has not been detected after a very long time, the robot can consider itself lost, and attempt to reorient itself in the map. Introspective experience suggests that humans do this sort of thing. If following a poorly known trail, one will eventually feel “I should have passed that ugly office tower by now,” and re-evaluate one’s position.

Likewise, if a landmark similar to the expected landmark is detected well before it is expected, it can be ignored.

4.4 Visual Improvements

Once these other methods have been considered, it is worth considering possible improvements in the vision based landmark recognition. Landmark recognition by the pattern-vector method has problems with lighting levels. If the blinds are opened or the lights switched off, successful recognition drops. This is because a small error is introduced into every pixel of the image. When these small errors are summed, the result is quite large. This problem could be solved by examining the distribution of the error. It is almost perfectly ?at; It should be possible to recognise this sort of pattern in the difference.

Distinct landmarks under similar lighting will be very different; instead of having the difference spread evenly across the image, the pixels will have large differences in some areas, and smaller differences in others.

5 A Physical Robot

Landmark detection and navigation using the concepts described above was implemented on the Tao 7 autonomous wheelchair.

5.1 Hardware

The Tao 7 is based on a Jazzy 1120 2000 purchased from Pride Mobility Products. A box inside contains a handful of computer boards. The main CPU board contains a Motorola 68332 processor running at 32 MHz. One megabyte of memory is available on this board. A Texas Instruments 6202 Digital Signal Processor running at 200 MHz performs vision calculation, and communicates with the main board over an RS-232 serial link. Figure 3 shows the Tao 7.

Eleven active infrared sensors and eight ultrasonic sensors are mounted around the robot, of which the majority are at the front.

The Tao 7 has several behaviours, listed in decreasing priority: infrared sensors, ultrasonic sensors, vision-based collision avoidance, and landmark detection/map following.

The avoidance and navigation behaviour source code occupies 7244 lines; the vision landmark detection code is 4621 lines. Clearly landmark detection and navigation can be performed without highly involved algorithms or powerful computers.

5.2 Environment

Extensive testing was performed in an unmodified office environment. The walls were blank white, and the floors were dark carpet; coloured desks, plants, people, and other obstruction were present. See Figure 4.

Figure 3: The Tao 7 Autonomous Wheelchair

A map was created with 24 landmarks covering a figure-eight shaped region in the office. Figure 5 shows the layout of the office. The landmarks were chosen to be at turns and important point in the corridors.

5.3 Map

Each landmark had at least two outgoing connections, one for continuing and one for returning; the only exceptions are a small number of landmarks added to ensure that a nearby landmark is approached from a consistent angle. The landmarks clustered around the reception/co-ops crossover point had extra connections to handle the crossing.

5.4 Landmark Detection

The three approaches described above for increasing landmark detection accuracy were implemented on the Tao 7.

5.4.1 Map Connectivity

Trial and error modification of the map was used to generate a usable map. Under certain circumstances it was difficult to have the robot approach landmarks in a sufficiently consistent manner. By arriving at an unusual angle the robot could either not recognise where it was, or could recognise the landmark, but the action required for one location would not function for another area within the same landmark. This problem could be handled partially by adding intermediate landmarks. Near reception, for example, landmarks were added for this purpose. By guiding the robot to the left wall it could arrive reliably at the next landmark.

5.4.2 Non-visual perception

Using the sensors already present on the Tao 7, a weak ultrasonic landmark recognition system was added alongside the visual landmark recognition; a behaviour was added which marks as invalid landmark guesses which do not make sense considering the state of the walls on either side of the robot. Each landmark includes information as to where walls should be to detect it: on the left, right, both or neither. If there exists a landmark with a larger (and still sufficiently small) visual error but with wall states matching the current condition, it will not be marked as invalid by the wall-checking behaviour, and will be successfully detected.

Figure 4: A corridor in the office

This technique improved landmark detection accuracy enormously. It is not without its problems — if the corridor is wide, the position of the wheelchair can vary considerably from run to run. A landmark that had a wall on the left on one test may be approached farther to the right in the corridor, and have a wall instead on the right, or no detectable walls at all. For these situations code was added so that a landmark may instead indicate that, for its walls, “the answer is unclear.” The wall-checking behaviour is then suppressed when detecting that landmark.

5.4.3 Timing

Timing between landmarks was a considerable problem, especially when driving clockwise around the northern loop or counter-clockwise around the southern loop; this is where the images of Figure 2 originate. After detecting the first landmark it would accept detection of the second, which, due to its visual similarity, would be immediately activated. By suppressing detection of the next landmark until the timing of the action of the previous landmark had expired, and by ensuring this action moved the robot out of the region of visual similarity, the timing issue was resolved.

5.5 Limitations

5.5.1 Open Space

Numerous problems were encountered dealing with the open space near reception. This is why a great deal of effort was required to program the robot to reliably navigate in this open area; if the robot closely follows the wall it can get guidance from this wall. In general, the navigation methods described above do not function well when there is little guidance from the environment.

Guidance from the environment can take many forms. In the office, the robot was restricted by walls, cubicle dividers, desks, chairs, potted plants, and other objects. A robot designed to navigate in a different environment would need to take path guidance cues from other sources.

Driving outside in a city, for example, the chair would need to be able to extract guidance from sidewalks, by detecting the edges. An autonomous automobile would extract path cues from the edges of the road or lane and the presence of other vehicles. Some driving “aids” have been described in IEEE Spectrum [10].

The concept of intelligence being based on interaction in the world is expressed by Brooks [2], but even more clearly by Chris Malcolm [11].

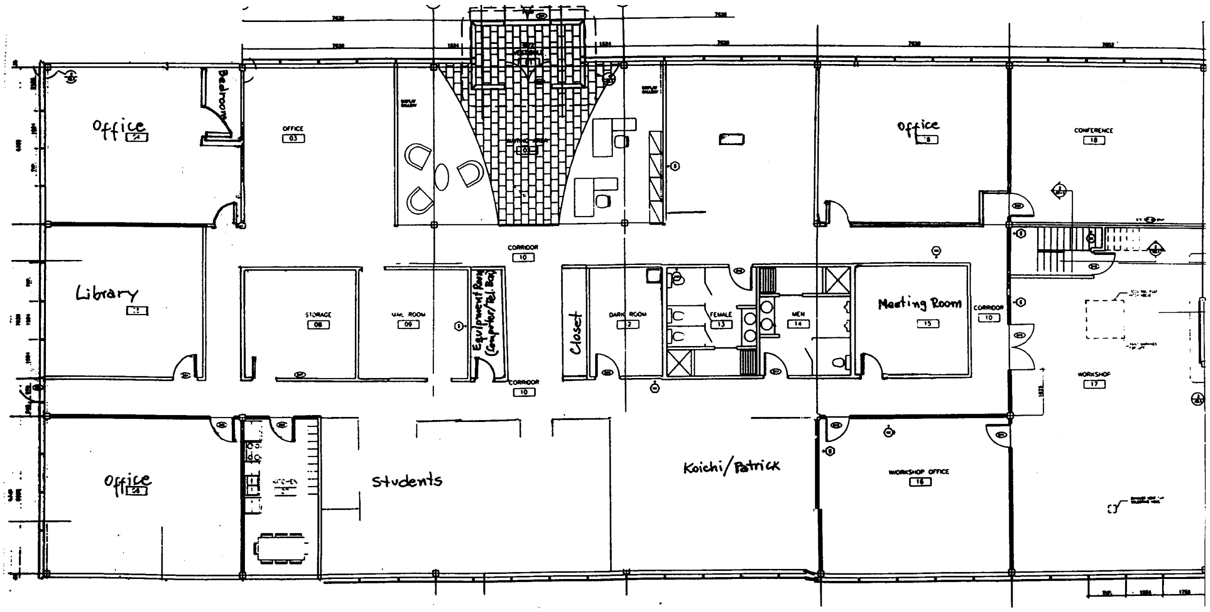

Figure 5: Layout of Tao 7's environment

It is interesting to note that humans also cannot navigate successfully without guidance from some overriding system. Directional and positional techniques and instruments (star navigation, the magnetic compass, sextant, and GPS) were required to allow navigation in an environment that is otherwise guidance free.

5.5.2 Detection Failure

The landmark detection behaviours still occasionally miss landmarks. Although this is very rare in the office environment, it ultimately cannot be avoided. Sometimes a landmark is obscured and the robot drives past it, it will eventually get to another landmark, but will not expect it, and so will not recognise it. It will also continue to expect the landmark it missed, regardless how much time elapses without finding it.

Due to the nature of Behaviour Based systems, these additions provide little benefit when considered individually, but when combined produce large improvements.

6 Conclusions

A typical office environment can successfully be navigated by an autonomous robot. Because landmarks only affect nearby landmarks, the map can be expanded without any limit besides available memory.

Where the environment provides very little guidance, in large empty rooms, fields, or lakes and oceans, landmark based navigation cannot easily be implemented. This matches biological models, which use other techniques such as directional guidance for non-organised environments.

7 RECOMMENDATIONS

Navigation is a broad topic, and there is much to be done before robots will be able to navigate in arbitrary environments. Some of the next steps are apparent; some suggestions are presented.

7.1 DESIGN IMPROVEMENTS

The robot, after finding a landmark, expects the next landmark. Currently, this expectation never ends, even if a very long time passes. Code should be added to permit the robot to determine when it is lost, and respond appropriately.

When creating the map, a large amount of "guess and check" was required. This means that switching to a different building is a difficult task; the map must be re-developed, a task nearly as difficult as making the first map. Manual map generation should be simplified drastically.

7.2 FURTHER STUDY

Detection failures can be reduced by expecting a larger number of landmarks: not only the next landmark, but other nearby landmarks. This may be a difficult task. In general the robot should have a level of confidence in its location, which is modified by seeing landmarks and timing. From certainty in its position the confidence would gradually decrease until it finds another landmark.

Free space navigation, although partially handled by adding landmarks, is quite a different task from environmentally-guided navigation. It requires as a map either a much more thickly connected graph or some other representation of space. Considering that the majority of human environments are well delineated (eg corridors by walls, roads by painted lines) this does not significantly limit applications of the ideas developed above.

It will not be acceptable to require a human to manually generate a map for every environment. The robot must be developed so that it can autonomously generate a map by wandering around. Giving it the name of each location, eg “this is the kitchen,” “this is my office,” is the most that the human should be forced to do.

REFERENCES

[1] Richard Edwards. New developments in an intelligent wheelchair. University of Waterloo Work Report, August 1996.

[2] Rodney A. Brooks. Intelligence without reason. In Proceedings of the 1991 International Joint Conference on Artificial Intelligence, pages 569–595, 1991.

[3] Rodney A. Brooks. A robust layered control system for a mobile robot. IEEE Journal of Robotics and Automation, pages 14–23, April 1986.

[4] David McFarland, editor. The Oxford Companion to Animal Behaviour. Oxford University Press, 1987.

[5] James J. Gibson. The Ecological Approach to Visual Perception. Houghton Mi? in Company, 1979.

[6] Reinhard Diestel. Graph Theory. Graduate Texts in Mathematics. Springer, second edition, 2000.

[7] Stuart Russel and Peter Norvig. Artificial Intelligence : A Modern Approach. Prentice-Hall, Inc., 1995.

[8] Maja J. Mataric. Learning a distributed map representation based on navigation behaviors. In Proceedings of the 1990 USA-Japan Symposium on Flexible Automation, pages 499–506, 1990.

[9] Peter H. Dana. Global positioning system overview. http://www.colorado.Edu/geography/gcraft/notes/gps/gps.html.

[10] Willie D. Jones. Keeping cars from crashing. IEEE Spectrum, pages 40–45, September 2001.

[11] Chris Malcolm. The organism-environment and robot-environment theories. http://www.mcgill.ca/douglas/fdg/kjf/22-C20MA.htm, 2000.