Defence Research and Development Canada
Recherche et développement pour la défense Canada

DEFENCE R&D DÉFENSE

Leader/Follower Behaviour Using the SIFT Algorithm for Object Recognition

J. Giesbrecht
DRDC Suffield

Technical Memorandum
DRDC Suffield TM 2006-108
June 2006

Canada

Leader/Follower Behaviour Using the SIFT Algorithm for Object Recognition

J. Giesbrecht
DRDC Suffield

Defence R&D Canada – Suffield

Technical Memorandum

DRDC Suffield TM 2006-108

June 2006

Author

J. Giesbrecht

Approved by

D. Hanna
Head/Tactical Vehicle Systems Section

Approved for release by

Dr. P. D'Agostino
Chairman/Document Review Panel

Abstract

This paper presents an application of vision-based object recognition to create a leader/follower behaviour in mobile robots. A system is developed which makes use of the Scale Invariant Feature Transform (SIFT) agorithm to recognize a leader robot or human. The follower robot then uses PID control to track the leader's movements while maintaining a fixed following distance.

Résumé

Cet article présente une application de la technique de reconnaissance visuelle d'objets visant à créer un comportement de chef et d'exécutant chez des robots mobiles. Un système en voie de mise au point utilise l'algorithme SIFT d'invariance d'échelle pour reconnaître un chef robotisé ou humain. Le robot exécutant utilise alors un régulateur PID pour suivre les mouvements du chef tout en maintenant une distance fixe de poursuite.

This page intentionally left blank.

Executive summary

Leader/Follower Behaviour Using the SIFT Algorithm for Object Recognition

J. Giesbrecht; DRDC Suffield TM 2006-108; Defence R&D Canada – Suffield; June 2006.

Background: Human/robot or multi-robot teams have a wide variety of potential applications. The ability of an autonomous mobile robot to control its position relative to team members is a key enabling component. This work focuses on a small, simple autonomous robot following a lead human or robot using only an inexpensive video camera, maintaining its position relative to the leader by recognizing an object which it has been trained to follow. This process requires two components in the follower robot: a visual recognition system which can provide information about the relative pose of the leader, and a control system which uses this information to adjust the follower robot's speed and direction.

Principal Results: The follower system relies upon object recognition using the Scale Invariant Feature Transform (SIFT) algorithm, built into the ViPR software libraries in the Evolution Robotics ERSP toolkit. This technique extracts feature points from a training image and compares these feature points to those extracted from successive camera images to recognize the leader's position. Using positional information from the SIFT object recognition code, the tracking controller used simple PID loops on the robot's translational and rotational velocity to follow the leader and maintain a safe following distance.

This system was tested in an indoor office environment, and was able to follow arbitrary leader objects at moderate walking speeds. However, due to the direct pursuit nature of the controller, the robot would bump intervening obstacles if the distance between the leader and follower was too great.

Significance of Results: These experiments proved the SIFT algorithm as a viable method of creating leader/follower behaviour, and can serve as a proof of concept for more complex convoying operations using machine-vision based leader detection.

Future Work: Given the current simplicity of this system, a large number of improvements and extensions are potentially available. Object recognition could be improved through the use of higher resolution, or zoom cameras. In order to control path following performance beyond that enabled by direct PID pursuit, global localization (such as GPS) would be required. With this, a pan/tilt unit could be added to the system to allow the vision system to track the leader's position while the robot followed it using the Pure Pursuit path tracking algorithm. The potential applications of this type of system need to be explored, such as a military convoying system. This will require adaptation to an Ackerman steered vehicle, and testing of the algorithm in less controlled, outdoor lighting conditions. Further system development would also see the introduction of a variety of robot behaviours through the recognition of a variety of fiducial objects, leading to much easier cooperation between human and machine.

This page intentionally left blank.

Sommaire

Leader/Follower Behaviour Using the SIFT Algorithm for Object Recognition

J. Giesbrecht; DRDC Suffield TM 2006-108; R & D pour la défense Canada – Suffield; juin 2006.

Contexte : Des équipes composées d'une combinaison d'humains et de robots ou bien multirobot présentent des possibilités d'applications très variées. La capacité d'un robot mobile autonome à contrôler sa position par rapport à d'autres membres de l'équipe est une composante clé de fonctionnement. Ce travail est axé sur un petit robot autonome simple poursuivant un chef humain ou robotisé ; il utilise simplement une caméra vidéo peu coûteuse pour maintenir sa position par rapport au chef en reconnaissant un objet qu'il a été entrainé à poursuivre. Ce procédé requiert deux composantes chez le robot exécutant : un système de reconnaissance visuelle qui peut fournir des informations au sujet de sa position en fonction de celle du chef et un système de contrôle qui ajuste la vitesse et la direction du robot exécutant.

Résultats principaux : Le système de l'exécutant dépend de la reconnaissance de l'objet, une technique basée sur l'algorithme SIFT réalisé dans les bibliothèques de logiciels ViPR de la trousse *Evolution Robotics ERSP***.** Cette technique extrait des points sélectionnés à partir d'une image d'entrainement et compare ces points sélectionnés à ceux extraits d'une succession d'images prises par une caméra reconnaissant la position du chef. En utilisant l'information positionnelle obtenue par le code de reconnaissance d'objet SIFT, le contrôleur de la poursuite utilise de simples boucles PID sur la vélocité translationnelle et rotationnelle du robot pour poursuivre le chef et maintenir une distance de poursuite sécuritaire.

Ce système a été testé dans un milieu de bureaux à l'intérieur et a réussi à poursuivre des objets chefs arbitraires à des vitesses de marche modérées. Le robot percutait cependant les obstacles qui interféraient si la distance entre le chef et l'exécutant était trop importante ; ceci provenait de la nature de la poursuite directe du contrôleur.

La portée des résultats : Ces expériences ont prouvé que l'algorithme SIFT était une méthode viable permettant de créer un comportement du chef et de l'exécutant et pouvant servir de validation de principe pour des opérations de convoiement plus complexes qui utiliseraient une vision artificielle basée sur la détection d'un chef.

Les travaux futurs : Étant donné la simplicité actuelle du système, il est possible d'y apporter un grand nombre d'améliorations et d'extensions. La reconnaissance d'objets pourrait être améliorée en utilisant une plus haute résolution ou un appareil photo à focale variable. Pour contrôler le rendement de la poursuite de parcours au-delà de la capacité d'une poursuite directe PID, il faudrait utiliser une localisation mondiale (telle que le GPS). On pourrait y ajouter une unité panoramique basculante qui permettrait au système de vision de poursuivre la position du chef pendant que le robot la poursuivrait en utilisant l'algorithme de poursuite mobile pure. Les applications possibles de ce type de système, telles que celles des systèmes de convoi militaires, doivent être explorées. Ceci demandera d'adapter les véhicules de type de propulsion Ackerman et de tester l'algorithme dans des conditions moins contrôlées de lumière extérieure. Une mise au point plus approfondie du système favoriserait l'introduction d'une variété de comportements de robots au moyen de la reconnaissance d'une variété d'objets de référence et permettrait une bien meilleure coopération entre l'humain et la machine.

This page intentionally left blank.

Table of contents

List of figures

This page intentionally left blank.

1 Introduction

There exist a wide variety of potential applications involving human/robot or multi-robot teams. The ability of an autonomous mobile robot to control its position relative to team members is a key enabling component. This work focuses on a small, simple autonomous robot following a lead human or robot using only an inexpensive video camera, maintaining its position relative to the leader by recognizing an object which it has been trained to follow. This process requires two components in the follower robot: a visual recognition system which can provide information about the relative pose of the leader, and a control system which uses this information to adjust the follower robot's speed and direction.

2 Background

2.1 Target Recognition

Many unmanned systems have accomplished leader/follower behaviour by sending positional information over a wireless data link[1]. This requires radio infrastructure and position finding equipment such as GPS, both of which are prone to failure and consume valuable bandwidth. Furthermore, it is advantageous for a human to be able to interact with a robot without the need for electronic equipment. Therefore, many researchers have used other detection systems to locate the leader, such as sonar[2] and laser range finders[3] which are range limited, prone to noise interference, and again require complex hardware.

A suitable solution to the problem is to use an inexpensive video camera and image processing techniques to find the leader's pose. The most basic approaches track a colour fiducial on the leader in consecutive video images, using the perceived size of the fiducial to estimate distance[4, 5]. Another approach recognizes the colour of a human leader's shirt[6]. These methods, although effective, can easily fail if a similar color object enters the robot's field of view or under changing lighting conditions. In an extension to this method, some researchers use color fiducials of a specific shape which allow additional information to be gathered, such as leader roll, pitch and yaw[2, 7, 8] . However, in addition to the previously mentioned limitations, this also makes the system sensitive to partial occlusion by obstacles between the leader and follower.

Recognizing an object on the leader human or robot to track it directly is a more robust and practical alternative. One approach used the taillights of a lead vehicle, which is effective at night[9]. Another implementation uses template based image recognition to retrieve leader distance and orientation from images[10]. This type of approach provides insensitivity to target occlusion and changes in lighting, and does not require the use of special fiducials on the lead robot or human.

2.2 Path Following

Once the position of the leader has been established, a control scheme must be implemented to actuate the robot's movements. There exist two basic options: attempt to match the leader's complete path, or simply follow the leader's current position directly. To implement the former, the robot must be able to keep a record of both its own and the leader's position

Figure 1: *The ER1 robot with camera and laptop.*

very accurately in world coordinates [1, 9]. It then repeats the leader's traverse using a path tracking algorithm such as Pure Pursuit[11]. Unfortunately, keeping track of position in world coordinates is not necessarily an easy problem. Additionally, this approach makes it more difficult for the follower to keep its vision sensor aimed at the lead robot, unless the robot is equipped with a pan/tilt camera[4, 7].

For simple robots, it is more practical to pursue the lead vehicle directly. A number of works are available on the topic. If the leader's position and orientation are known, the follower can calculate a trajectory to attain that pose using the Vector Pursuit method[12], or with Bezier Trajectories [7, 13]. Another work uses a "virtual trailer link" model to recreate the leader's motions when following closely[3]. These approaches have the advantage of following the leader's motions more accurately, but will not keep the vision sensor aimed on the leader for tracking purposes. If no orientation information is available, a "tail chase" method is often adopted whereby the leader's current position or position/velocity are considered as a target to pursue using a kinematic model of the vehicle[14, 15].

An even simpler approach reduces the problem of controlling robot motion in Cartesian coordinates to a visual servoing problem[2, 5, 16, 17]. The robot simply tries to keep the recognized image centered in its field of view by controlling the wheels of the robot. This method removes all modeling in world coordinates and ensures that the lead vehicle stays in the follower's field of view. The downside to this approach is that the leader's trajectory is not followed as accurately, and the robot will cut corners dramatically if it falls a large distance behind.

The novel approach taken in this work combines the powerful object recognition techniques provided by the SIFT algorithm to follow an arbitrary leader, with a simple PID based control scheme suitable for a small indoor robot.

3 Hardware

The robot used in this application was the ER1 from Evolution Robotics, shown in Figure 1. It has two independently driven wheels and power provided by an on-board battery pack.

Figure 2: *A trained image and the recognized image in a cluttered scene, with the red box indicating the position of the recognized object. Feature points are shown as yellow circles.*

It is a small, simple and low cost robot measuring approximately 80cm high and 40cm wide. The only sensor used in this work is an IRez Kritter USB camera with 640x480 resolution mounted at the top of the robot. Processing power was provided by a Dell D600 laptop running Fedora Core 3 Linux on a Pentium M 2.0Ghz CPU. Two Lucent Orinoco 802.11 PCMCIA wireless cards were used for communication with the robot laptop from a base computer.

4 Target Recognition Using SIFT

The follower system relies upon object recognition using the Scale Invariant Feature Transform (SIFT[18]) algorithm, built into the ViPR software libraries in the Evolution Robotics ERSP toolkit[19]. This technique extracts feature points from a training image and compares these feature points to those extracted from successive camera images. For a planar leader object, only one training image is necessary. For 3D objects, training images from different views makes the algorithm more robust.

The algorithm detects unique features in an image of an object by analyzing the texture of a small window of pixels. Up to 1,000 feature points are extracted from an image, each consisting of the feature's location and a texture description. A small portion of these features, filtered for uniqueness and robustness make up a model database for that object.

When attempting to recognize an object, it extracts similar types of features from a newly acquired image, associating them with with those in the trained model database. It compares these feature points using the texture descriptors. If there is a high number of feature matches between the acquired image and a trained model, a potential match is supposed. At this point, it attempts to match the acquired image to the trained model by applying an affine transform. If the difference between the transformed acquired image and the original trained model is low enough, it declares a match.

This method has a number of very desirable characteristics for real world applications. It is unaffected by changes in scale, rotation and translation. It also has some robustness to changes in lighting, and can be used on low cost, low resolution cameras. Finally, the algorithm will typically recognize objects with 50% to 90% occlusion. It specializes in planar, textured objects, but also works well with 3D objects having slightly curved

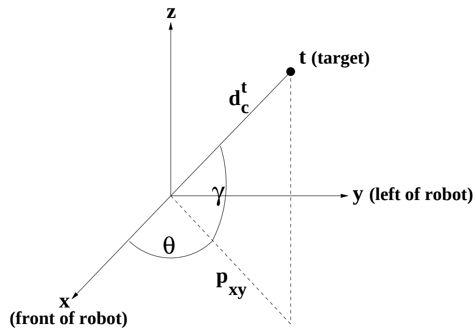

Figure 3: *Geometry of ER1 in image space.*

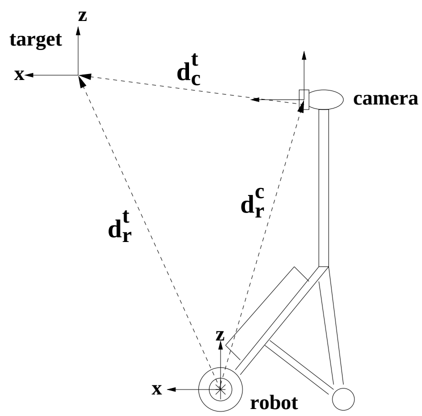

Figure 4: *Geometry of robot and target.*

components. A model image and the subsequent recognized image are shown in Figure 2.

5 Leader Tracking

The object recognition stage outputs a positive or negative recognition for each camera image. For positive recognition, it also supplies the model name, the pixel location of the object's center in y,z coordinates, and a distance estimate, $(t_y, t_z$ and $p)$.

This information can be used to find the leader's relative position. The camera used has a field of view of 46 degrees horizontally spread over 640 pixels, resulting in approximately 0.00125 radians/pixel, S. This constant is an approximation obtained from experiment, and does not take into account the unknown characteristics of the camera lens. The minimal effects of this simplification are reviewed in Section 7.1.2. Figure 3 shows the image geometry. If the number of pixels to the center of the image are C_y and C_z, the angles to the leader in the horizontal and the vertical planes,θ and γ are found as follows:

$$\theta \cong (C_y - t_y)S \tag{1}$$

$$\gamma \cong (C_z - t_z)S \tag{2}$$

From this, the distance from the camera to the target in the x,y plane (p_{xy}) can be found:

$$p_{xy} = p\cos(\gamma) \tag{3}$$

The vector $\mathbf{d}_c^t$ from the camera to the target is:

$$\mathbf{d}_c^t = \begin{bmatrix} p_{xy}\cos(\theta) \\ p_{xy}\sin(\theta) \\ p\sin(\gamma) \end{bmatrix} \tag{4}$$

Finally, the vector from the robot's wheels to the target, $\mathbf{d}_r^t$ is found using the static vector from the wheels to the camera, $\mathbf{d}_r^c$:

$$\mathbf{d}_r^t = \mathbf{d}_r^c + \mathbf{d}_c^t \tag{5}$$

This vector is used to control the vehicle's motion relative to the target.

6 Robot Control

As discussed in Section 2, there are a variety of methods for control in a leader/follower scenario. For simplicity's sake, basic PID control is used on heading and velocity in this application. The ER1's low level controller accepts velocity commands in terms of rotational and translational velocities in rad/sec and cm/sec (ω, v), so the heading and distance controls are decoupled and treated as separate control loops.

6.1 Heading Control

In order to pursue the leader directly, the heading controller attempts to fix the follower's heading directly at the leader by forcing the angle $\theta = 0$. The PID loop shown in Figure 5 changes the rotational velocity of the robot based upon the value of θ found in the previous step. $K_{p\omega}$, $K_{i\omega}$ and $K_{d\omega}$ represent the proportional, integral and derivative gains respectively for heading.

$$\omega = K_{p\omega}\theta + K_{d\omega}\frac{d\theta(t)}{dt} + K_{i\omega}\int \theta(t)dt \tag{6}$$

However, this is the ideal form of PID control, and does not represent a practical implementation in code. Therefore, it is approximated for each time step n with sampling time T_s:

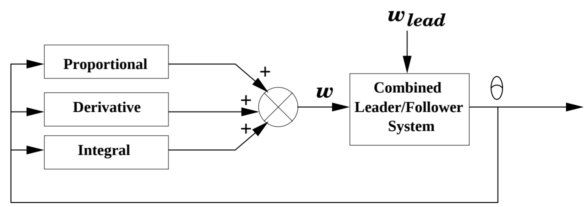

Figure 5: *PID loop for heading control.*

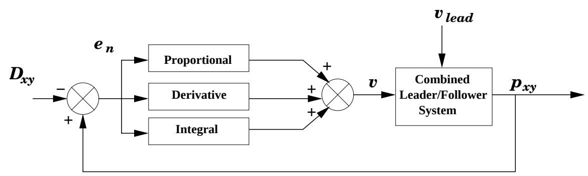

Figure 6: *PID loop for velocity control.*

$$\omega = K_{p\omega}\theta_n + K_{d\omega}\theta_{\Delta n} + K_{i\omega}\sum_{i=1}^{n}\theta_{\Omega i} \tag{7}$$

where

$$\theta_{\Delta n} = \left[\frac{\theta_n - \theta_{n-1}}{T_s}\right] \tag{8}$$

$$\theta_{\Omega i} = \left[\frac{\theta_i + \theta_{i-1}}{2}\right] T_s \tag{9}$$

6.2 Velocity Control

The velocity controller aims to keep the follower robot at a fixed distance in the x,y plane behind the leader robot while matching the leader's velocity, as shown in Figure 6. This means minimizing the error between the distance to the leader in the x,y plane, p_{xy}, and the following distance D_{xy} by adjusting the robot's translational velocity v.

$$e_n = p_{xy} - D_{xy} \tag{10}$$

$$v = K_{pv}e_n + K_{dv}e_{\Delta n} + K_{iv}\sum_{i=1}^{n} e_{\Omega i} \tag{11}$$

where

$$e_{\Delta n} = \left[\frac{e_n - e_{n-1}}{T_s}\right] \tag{12}$$

$$e_{\Omega i} = \left[\frac{e_i + e_{i-1}}{2}\right] T_s \tag{13}$$

Other than heading and velocity control, the system also has some other features. As a safety catch for the controller, if the system recognizes the object within a pre-defined safety distance, it sets ω and v to zero. Also, if no object is detected within a certain pre-set timeout period, the robot will set v to zero and ω to some constant C so that it slowly spins trying to reacquire the leader. Finally, it can also be trained on other leader signal objects to inform the robot that it should halt and perform no further recognition.

7 Results

Several tests were undertaken on the software system described, divided into two sections: Target Recognition and Robot Control.

7.1 Target Recognition

Using the hardware described earlier, these tests characterize the functionality of the SIFT object recognition software. The robot was kept static while various properties of the target were changed, such as distance, orientation, occlusion and position in the robot field of view. Each of the data points presented in the graphs 7-9 represent 100 target recognitions for each change in target.

The "Linux In A Nutshell" book pictured in Figure 2, which served as a leader target for these tests, only exemplifies a *typical* leader target. Other targets with more or less texture would cause results much different than those presented here. It also must be noted that this is a small target (15cm by 10cm). It is assumed that training a larger target would allow the image recognition system to work at longer distances, but reduces the practicality for the leader robot or human.

7.1.1 Distance to Target

The first experiment tested the effect of distance to target. The robot camera was first placed at a distance of 25cm from the target to train the image. The target was then moved in the camera's x-axis in increments of 50cm, with 100 recognition samples collected at each point. The results can be seen in Figure 7. It can be seen in the graph that there is a sharp dropoff in system performance at distances beyond 350cm. Furthermore, a slight offset is found in the distance to target in the 50 to 350cm range. It can be reasonably assumed that this is due to error in finding the distance from the focal position of the camera to the target when training the target object. A small error at this stage would create much larger errors in the recognition stage, when the target is much further away.

Another factor which affects the overall performance of the system is percent recognition. If the object is not recognized, no distance and position to the target is returned and the

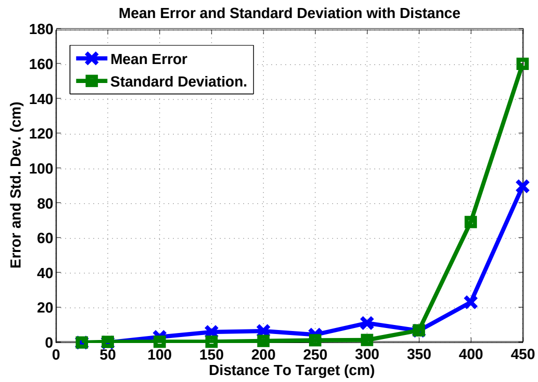

Figure 7: *The effect of distance to target on the mean and standard deviation of the distance measured.*

system cannot function. In these tests, the system recognized the target 100% of the time up to distances of 4 meters and 76% recognition at 4.5 meters. However, there was so much error in the distance measurement at these distances that the results would be unusable by the robot.

7.1.2 Target Position

This tests the calculation of target position in the robot's field of view, at a distance to target of 200cm. For this test, the object was moved in the robot's yz plane (i.e. left to right in the robot's field of view), and this displacement measured by the object recognition system. Starting with 0cm displacement, the mean error and standard deviation in distance measurement are presented in Figure 8. A displacement of 80cm represents the edge of the camera's field of view. From this graph it can be seen that there is a slight error in the position calculated, which increases with displacement. This error results from the approximation used to calculate the angle to target from its pixel location, in radians per pixel, which is not a completely accurate means. Despite this, the error is only an average of 2.5cm over the 80cm displacement, and did not noticeably affect system performance in following the leader object.

.

7.1.3 Target Orientation

Another factor which could potentially affect the system is the orientation of the target. Because of the mechanics of the SIFT algorithm, rotation in the robot's yz plane (about its x-axis) had no substantial effect on on any parameter as compared with the previous tests. However, as common sense would dictate, the angle about robot's y or z axes has substantial effect, due to the occlusion of details in the target. The system performed comparably with previous tests up to an angle of 40 degrees about the y or z axes, with an extremely sharp dropoff in both percent recognition and accuracy beyond this angle.

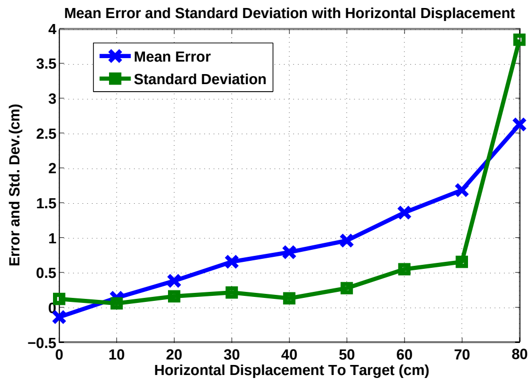

Figure 8: *The effect of horizontal displacement of target on the mean and standard deviation of the distance measured.*

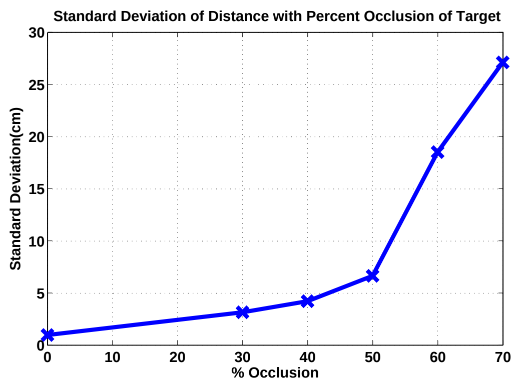

Figure 9: *The effect of target occlusion on the standard deviation of the distance measured.*

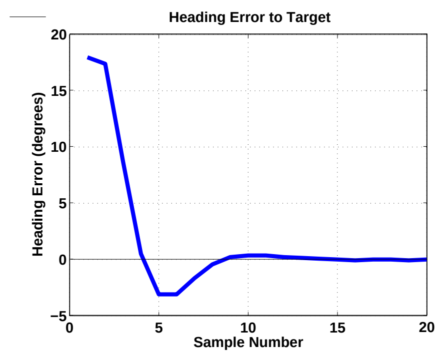

Figure 10: *The results of tuning the heading control PID loop.*

7.1.4 Target Occlusion

The final parameter of the object recognition system tested was the target occlusion. The distance to target was again set at a static 200cm for this test. The results of occluding the target by a given percentage are shown in Figure 9. It can be seen that the system performs very well up to an occlusion of about 50%. Once more, the percent recognition drops off quite sharply with this parameter. At 200cm, there was 100% recognition with 70% target occlusion, 11% recognition at 80% occlusion, and the system never recognized the target with 90% occlusion. Once more, it is important to bear in mind that the choice of target could greatly affect this result. A target with lots of detail spread all across it surface would be easier for the system to recognize with higher occlusion than one with large amounts of white space.

7.2 Robot Control

The second set of tests examined the tuning and performance of the robots rotational and translational velocity control.

7.2.1 Heading PID

The robot's translational velocity was disabled for this test, and the robot was oriented at a distance of 200cm in its x-axis and 80cm in its y-axis from the target. This presented a step change in orientation error to the heading PID controller. The angle sensed by the object recognition system versus sample number is shown in Figure 10. This system was intentionally underdamped for better performance in the leader/follower scenario to create more responsive behaviour when the leader's position is constantly changing. Once the leader begins turning, it is likely that he will continue turning.

7.2.2 Velocity PID

For this test, the robot was once more positioned 200cm from the static target object, with the distance to follow the leader set to 100cm. This represented a step change for the Velocity PID controller. Results are shown in Figure 11 as the robot moves itself to 100cm away. Once again this system was intentionally underdamped to provide stronger control with changing setpoints.

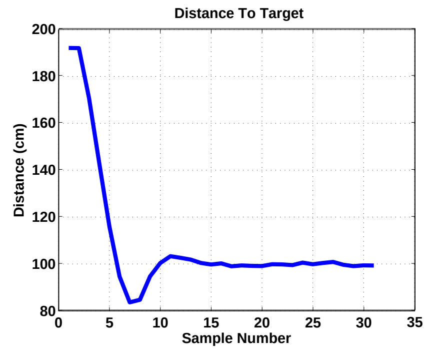

Figure 11: *The results of tuning the position control PID loop.*

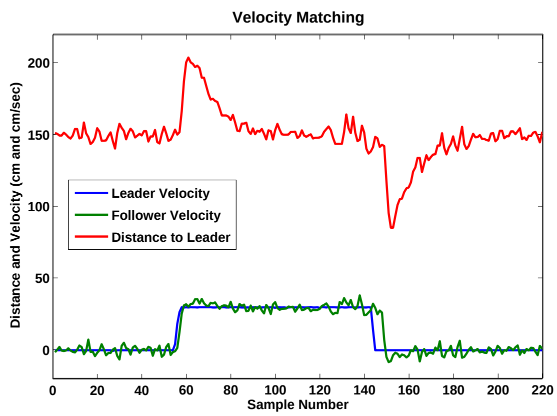

Figure 12: *Following distance and velocity with a leader robot moving at 30 cm/sec.*

With the robot showing good distance control to a static target, the system was tested with a moving leader robot. The follower was allowed to settle at its following distance of 150cm behind a static leader, before the leader accelerated to a constant velocity of 30 cm/sec. As is shown in Figure 12, the follower was able to match the leader's velocity, while mainting a fixed following distance fairly well. The follower's distance to leader(red) and velocity(green) are shown against sample number(0.5 seconds each).

7.3 System Performance

With all of the subsystems functioning well, the full leader/follower behaviour was tested in an office environment. A number of qualitative observations were made. The first is that the system had trouble with tight corners. The limited field of view of the camera kept it from seeing the leader if the turn was too sharp. Secondly, the heading PID controller pursued the leader directly, causing it to bump intervening corners if the leader was a long distance away.

The execution time for the system was found to be about 500msec on the hardware indicated. This meant that the PID loops were only being updated twice per second, hampering the speed control algorithm. It seemed to have difficulty estimating the leader's speed under acceleration and deceleration, although it would maintain the set following distance quite well under steady-state conditions. Additionally, the limited top speed of the platform meant it could not keep up to a human walking at full speed.

A third limitation of the system was the problem of false positive recognition, which would cause the robot to chase ghost targets if the actual leader was not in its field of view. Changing the positive recognition threshhold of the image recognition software would solve this problem.

8 Conclusion

Using an inexpensive, low resolution camera, and a simple robot platform, leader/follower behaviour has been created using the SIFT algorithm for object recognition, and PID control of rotational and translational velocity. The system was found to be effective recognizing and accurately positioning everyday objects up to a distance of 3.5 meters. Additionally, it was able to do so despite changes in orientation of the target object, and occlusion by intervening objects. Using this ability, it successful followed both human and robot leaders carrying a fiducial object in an office environment at moderate human walking speeds.

A number of limitations to this system were found. The 3.5 meter effective distance for the recognition system is adequate for a small robots operating indoors, but would not be adequate for larger outdoor platforms. Additionally, direct pursuit of the leader's current position is quite rudimentary and does not work well in complex environments. PID control loops were time consuming to properly tune, and the performance of the simple robot platform limits the applicability of the system as implemented to wider applications.

Despite these shortcomings, the benefits of using an object recognition technique for this application were immediately apparent. Firstly, the immunity to orientation and occlusion problems made the system easy to use. Secondly, although it was still beneficial to use a fiducial on a human leader rather than direct recognition, it was possible to use ad-hoc fiducials chosen at application time rather than during development. Object recognition also allows for the implementation of a wide variety of different behaviours based upon a set of different trained objects, opening the way for new avenues of human robot cooperation. And despite the image recognition software iterating at only 2Hz, the controllers were able to perform adequately for the task. With the application of increased processing power to the problem, robot control performance would improve.

9 Future Work

Given the simplicity of this system, a large number of improvements and extensions are available. In regards to the issue of object recognition, a number of solutions could provide improvements. Most easily, training on a larger target object from further away would immediately improve the ability to recognize the target at a farther distance and provide more accurate target localization. Secondly the implementation of higher resolution camera could also provide similar significant benefits. If this was still inadequate, a zoom lens controllable by the robot system would alleviate the issue further. Training the follower robot on multiple sides of a 3D object rather than on one side of a 2D object would also increase robustness.

In order to control path following performance beyond that enabled by direct PID pursuit, orientation cues using the bounding box from the object recognition software could be used. With this, more complex methods of control such as Bezier curves could be used. If it were desirable to track the complete path of a leader very exactly, global localization would be required. If this were in place, a pan/tilt unit could be added to the system to allow the vision system to track the leader's position while the robot followed it using the Pure Pursuit path tracking algorithm. The addition of more processing power to the problem would also allow more accurate tracking of the leader by reducing recognition iteration times.

Finally, the potential applications of this type of system need to be explored, such as a military convoying system. This will require adaptation to an Ackerman steered vehicle, and testing of the algorithm in less controlled outdoor lighting conditions. Further system development would also see the introduction of a variety of robot behaviours through the recognition of a variety of fiducial objects, leading to much easier cooperation between human and machine.

References

[1] Hogg, R., Rankin, A., McHenry, M., Helmick, D., Bergh, C., Roumeliotis, S., and Matthies, L. (2001), Sensors and algorithms for small robot leader/follower behavior, In *Proc. SPIE Vol. 4364, p. 72-85, Unmanned Ground Vehicle Technology III.*

[2] Smith, J. (2001), A Tracking System for a Seismic Surveying Mobile Robot, Master's thesis, University of Alberta.

[3] Ng, Teck Chew and Guzman, J. (2005), Autonomous Vehicle-Following Systems: A Virtual Trailer Link Model, In *Proceedings of IEEE/RSJ Conference on Intelligent Robots and Systems.*

[4] Hong, P., Sahli, H., Colon, E., and Baudoin, Y. (2001), Visual Servoing for Robot Navigation, In *Third International Conference on Climbing and Walking Robots.*

[5] Sattar, J., Giguere, P., Dudek, G., and Prahacs, C. (2005), A Visual Servoing System for and Aquatic Swimming Robot, In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems.*

[6] Nguyen, H., Kogut, G., Barua, R., and Burmeister, A. (2004), A Segway RMP-based Robotic Transport System, In *In Proceedings of SPIE Mobile Robots XVII.*

[7] Cervera, E., Martinet, P., and Renaud, P. (2004), Towards a reliable vision-based mobile robot formation control, In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems.*

[8] Parker, L., Kannan, B., Tang, F., and Bailey, M. (2004), Tightly-Coupled Navigation Assistance in Heterogenous, In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems.*

[9] Sukthankar, R. (1993), Racoon: A Real Time Autonomous Car Chaser Operating Optimally at Night, In *Proceedings of the IEEE Intelligent Vehicles and Systems.*

[10] Benhimane, S., Malis, E., Rives, P., and Azinheira, J. (2005), Vision-based control for car platooning using homography decomposition, In *Proceedings of the IEEE International Conference on Robotics and Automation.*

[11] Coulter, R. Craig (1992), Implementation of the Pure Pursuit Path Tracking Algorithm, (Tech Report CMU-RI-TR-92-01), Carnegie Mellon University.

[12] Wit, J., Crane, C., and Armstrong, D. (2004), Autonomous Ground Vehicle Path Tracking, *Journal of Robotic Systems*, 21(8), 439–449.

[13] Chiem, S. and Cervera, E. (2004), Vision-Based Robot Formations with Bezier Trajectories, In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems.*

[14] Naeem, W., Sutton, R., and Ahmad, S. M. (2003), Pure Pursuit Guidance and Model Predictive Control of an Autonomous Underwater Vehicle for Cable/Pipeline Tracking, In *World Maritime Technology Conference*.

[15] Das, A., Fierro, R., Kumar, R., Southall, B., Spletze, J., and Taylor, C. (2001), Real-Time Vision-Based Control of a Nonholonomic Mobile Robot, In *Proceedings of IEEE International Conference on Robotics and Automation*.

[16] Cowan, N., Shakernia, O., Vidal, R., and Sastry, S. (2003), Vision-Based Follow-the-Leader, In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*.

[17] Malis, E. and Benhimane, S. (2005), A unified approach to visual tracking and servoing, *Robotics and Autonomous Systems*, 52, 39–52.

[18] Lowe, D. (2004), Distinctive Image Features from Scale-Invariant Keypoints, *International Journal of Computer Vision*, 60(2), 91–110.

[19] Munich, M., Pirjanian, P., Bernardo, E., Goncalves, L., Karlsson, N., and Lowe, D. (2005), Break-Through Visual Pattern Recognition for Robotics and Automation, In *Proceedings of IEEE International Conference on Robotics and Automation*.

This page intentionally left blank.

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Centre sponsoring a contractor's report, or tasking agency, are entered in section 8.) Defence R&D Canada – Suffield PO Box 4000, Station Main, Medicine Hat, AB, Canada T1A 8K6	2. SECURITY CLASSIFICATION (overall security classification of the document including special warning terms if applicable). UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C,R or U) in parentheses after the title).

Leader/Follower Behaviour Using the SIFT Algorithm for Object Recognition

4. AUTHORS (last name, first name, middle initial)

Giesbrecht, J.

5. DATE OF PUBLICATION (month and year of publication of document) June 2006	6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc). 28	6b. NO. OF REFS (total cited in document) 19

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered).

Technical Memorandum

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include address).

Defence R&D Canada – Suffield
PO Box 4000, Station Main, Medicine Hat, AB, Canada T1A 8K6

9a. PROJECT NO. (the applicable research and development project number under which the document was written. Specify whether project).	9b. GRANT OR CONTRACT NO. (if appropriate, the applicable number under which the document was written).
10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique.) DRDC Suffield TM 2006-108	10b. OTHER DOCUMENT NOs. (Any other numbers which may be assigned this document either by the originator or by the sponsor.)

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

(X) Unlimited distribution
(   ) Defence departments and defence contractors; further distribution only as approved
(   ) Defence departments and Canadian defence contractors; further distribution only as approved
(   ) Government departments and agencies; further distribution only as approved
(   ) Defence departments; further distribution only as approved
(   ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution beyond the audience specified in (11) is possible, a wider announcement audience may be selected).

Unlimited

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

This paper presents an application of vision-based object recognition to create a leader/follower behaviour in mobile robots. A system is developed which makes use of the Scale Invariant Feature Transform (SIFT) agorithm to recognize a leader robot or human. The follower robot then uses PID control to track the leader's movements while maintaining a fixed following distance.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title).

leader/follower
robotics
navigation
object recognition