Covert Behaviour Detection in a Collaborative Surveillance System

Punarjay Chakravarty David Rawlinson Ray Jarvis
National ICT Australia and Monash University
punarjay.chakravarty@nicta.com.au,david.rawlinson@nicta.com.au,ray.jarvis@eng.monash.edu.au

Abstract

Marzouqi and Jarvis' covert path transform [Marzouqi and Jarvis, 2003] allows a robot to plan a path that minimizes robot visibility, in an effort to avoid detection. This paper uses the same transform to measure the overtness of human movements, in an attempt to identify deliberately covert and suspicious human behaviour.

In a series of trials we detect and track both anonymous individuals and known targets (sentries) using multiple cameras. Prior to tracking the system is calibrated automatically by simultaneous observations of a mobile robot in both image and ground planes. Subsequently, people are tracked on the ground plane and the proposed covertness metrics indicate the extent to which their trajectories are covert or overt with respect to the environment and with respect to patrolling sentries. Both these metrics show differences between normal and covert human trajectories.

1 Introduction and Related Work

The algorithms for covert behaviour detection described in this paper are part of a larger collaborative surveillance system developed during the first author's PhD candidature [Chakravarty, 2011a]. The system comprises a mobile robot and a network of fixed surveillance cameras. The robot is capable of detecting and tracking people within the field of view (FOV) of its on-board panoramic camera as it patrols its environment [Chakravarty and Jarvis, 2008]. It is also capable of autonomously calibrating all the external cameras to the ground plane of the environment it is operating in [Chakravarty, 2011b]. When an intruder is out of range of the robot's on-board sensing, but is detected in one of the external surveillance cameras, the system is able to guide the robot to the ground plane position of the detected intruder using the calibration information for that camera [Chakravarty and Jarvis, 2009].

Once the robot has calibrated all fixed cameras it is possible to track other targets on the ground plane. Ground plane tracking enables analysis of target trajectories, and classification as covert or overt.

Our method of determining the covert/overtness of human paths was inspired by the Dark Path algorithm [Marzouqi and Jarvis, 2003], previously used in the simulation of robot war games. Given a map of the environment, the Dark Path is the most covert route between start and goal locations. The Dark Path is a set of connected, minimally observable non-obstacle points, produced by transforming an obstacle map into a general visibility map. Areas that are observable from many locations are more "visible".

Existing work has focused on use of the covert path transform to allow a robot agent to avoid detection. The proposal here, is to concentrate on analysis of human trajectories, rather than scrutinize the characteristics of individuals' appearance, gait, size or shape. Also proposed, is analysis of targets' positions with respect to each other: how do the targets interact? In our work, the overtness of a human target is given by computing two functions at each point on his or her trajectory:

1. Observability with respect to other targets (human or robot sentries).
2. Observability with respect to all non-obstacle points in the environment.

Both metrics of observability are measured in terms of line-of-sight, which we achieve by ray-tracing between locations.

[Hu *et al.*, 2004a] provide a survey of the different methods of analyzing the behaviour of tracked people. In the literature, the determination of covert or overtness of peoples' trajectories is less comprehensively researched than recognition of individuals' characteristics but some pertinent work on classifying behaviour based on trajectories is described below. A common feature of these works is that model parameters are derived from

training data, rather than application of a prior model as in our case. In consequence, these works tend to detect statistical anomalies rather than specific characteristics of interest.

[Hu *et al.*, 2004b] train a neural network using normal trajectories and are subsequently able to use the network to detect trajectories that deviate from the learnt normal ones. This works well in situations where a large body of human traffic behaves consistently, as is often the case in public transport stations. However, it cannot be applied in low traffic secure environments, where there is insufficient ordinary behaviour for learning.

In other work [Brand *et al.*, 1997; Natarajan and Nevatia, 2007], computer-synthesized agents are used to simulate the trajectories of people meeting and interacting with each other. These agents are used to train Coupled Hidden Markov Models (HMMs) and Coupled Hidden Semi Markov Models, mathematical modelling frameworks for classifying behaviours like "Follow", "Approach, Talk and Continue Separately", "Approach, Talk and Continue Together", etc.

[Nguyen *et al.*, 2005] also use track trajectories obtained from an overhead camera to model a person's activities in a room. Track segments between landmarks (door, chair, TV, etc.) in a room are modelled as primitive behaviours and sequences of these are strung together to model complex behaviours such as Short Meal, Normal Meal, etc. Primitive and complex behaviours are mapped into a hierarchical HMM, whose parameters are trained using 45 training sequences. 43 test sequence behaviours are perfectly identified by their algorithm.

HMMs are a convenient framework for describing sequences of discrete movements. However, in trajectory analysis and many other real-world problems it can be difficult to decompose a stream of continuous measurements into discrete states. These problems are compounded by weakly or partially ordered sequences with many permutations. For example, activity 3 could sometimes follow activity 1, but other times follow activity 2. In addition, the set of states and partial ordering might not be known a priori.

[Bui *et al.*, 2008] have tried to address these problems in trajectory analysis. They carried out experiments in which students carried GPS devices around campus for a few days. A density-based clustering algorithm was first used to cluster GPS locations according to time spent at each location. Sequences of GPS locations were associated with 7 clusters, identified by location (Bank, Watson Theatre, etc.) and tagged with activities (Banking, Lecture 1, etc.). An example of partial ordering of a student's activities is: "Lecture 2 follows Group Meeting 1 but could also follow Lecture 1; Group Meeting 1 follows Lecture 1, but could also follow Breakfast." They introduce the Hidden Permutation Model (HPM) as a means of learning and recognizing high-level routines which consist of permutations of "hidden" partially ordered activities. They demonstrated better recall and precision for recognition of activities, compared to other baseline methods.

Statistical modelling of activities or behaviour has the drawback that observations have to be hand-labelled for the models to be learnt, which can be a time consuming and laborious process. There is also the danger that an overly complicated model is over-fitted to noisy data and will perform poorly on unseen datasets.

[Dee and Hogg, 2004; Dee and Hogg, 2005] describe a system that does not use statistical models to determine the most commonly taken routes, and is able to interpret trajectories not previously observed during training. This system can also deal with moving non-target objects such as cars.

Their first system [Dee and Hogg, 2004] determines the goal-directedness of a person by using the direction of the person's motion, along with a model of possible goal locations for the person at each point on his/her trajectory, given the location of obstacles in the scene. Obstacles are marked off manually in the image plane, and like our work, ray-tracing is used to find out a polygonal area of visibility in an arc of one radian on either side of the direction of motion of the tracked entity. Obstacle vertices falling within this polygonal area are marked as goals. In subsequent work [Dee and Hogg, 2005], they outline an algorithm that uses a modified Hausdorff measure to compare points on the trajectory followed by a person with points on trajectories to known goal sites generated by the system.

Our work does not require statistical modelling of observed historical track trajectories, to identify outliers. Instead it classifies behaviour based on functions of trajectories' relative visibility, and visibility with respect to the environment. Our primary contributions are:

1. Metrics for classification of peoples' behaviour as covert or overt, based on track trajectory relative to other persons and trajectory with respect to the environment.

2. Description of a practical framework in which these metrics can be computed, using typical surveillance cameras.

The remainder of the paper is organised as follows: Section 2 describes the autonomous calibration of external cameras to the ground plane using a mobile robot. Section 3 introduces the two visibility metrics we use to determine the covertness of track trajectories. Section 4 discusses the experiments conducted; their results are presented in Section 5. Concluding remarks are given in Section 6.

2 Autonomous Camera Calibration and Target Tracking

The system described in this paper comprises two low-resolution surveillance video cameras, a mobile robot, and observation of people in the environment visible to the cameras. The cameras are used to track people and the robot. The robot has two roles. First, it enables automatic computation of the relationship between every camera and a common coordinate system on the ground plane. Second, the robot can function as an autonomous sentry. The trajectories of people observed by any or all cameras are then analysed in an attempt to detect covert behaviour.

This section describes camera-calibration and target-tracking components that enable people to be tracked on the ground plane. Subsections and their content are as follows:

1. The localization of the robot on the ground plane, in subsection 2.1.
2. The detection and tracking of targets in the image plane, in subsection 2.2.
3. The autonomous calibration of each camera image plane, to the ground plane, using homography, in sub-section 2.3.

For a more detailed explanation of the algorithms discussed in this section, the reader is referred to [Chakravarty and Jarvis, 2009].

2.1 Particle Filtering for Mobile Robot Localization

The mobile robot is equipped with a Hokuyo scanning laser range finder and is able to localize itself on the ground plane. Localization is achieved using a 2-D map of obstacles on the plane of the robot's Hokuyo sensor, and a particle-filter localization algorithm [Chakravarty and Jarvis, 2009]. The map is provided a priori, but there are well known methods for autonomously creating occupancy maps of this type [Thrun *et al.*, 2005]. Each particle (x, y, θ) represents a possible pose of the holonomic robot on the map. Laser range-finder readings predicted by each particle are compared to actual readings from the sensor. Particles (poses) that agree with observed laser measurements are given higher weights than particles that don't match. Particles are also moved between cycles of the algorithm using odometry data the robot's wheel encoders. Particles are resampled according to their weights so that particles with higher weights survive and particles with lower weights die out. Over time, particles cluster around the actual pose of the robot.

2.2 Target Detection and Tracking

Mixture of Gaussians (MoG) [Stauffer and Grimson, 2000] background subtraction is used to detect targets in the image plane. The background is modelled at each pixel using a mixture of weighted Gaussian probability density functions in the 3 colour channels R, G and B, each considered independent of the other. During an initialization phase, the first 100 frames are stored in memory, and 3 clusters are found for each pixel in the R-G-B space using a k-means clustering process. The means and standard deviations of each cluster are used to initialize the Gaussians, whose weights are initialized to be low.

Thereafter, the pixel intensity for every new image, is checked against the Gaussians modelling the background at that pixel. If it is within 2.5 standard deviations of the mean of any of the currently modelled Gaussians in all 3 colour channels, then the pixel is considered part of the background and the corresponding Gaussian weighted higher. If this is not the case, then the pixel is considered foreground and a new Gaussian is introduced into the background model, centred around this intensity value, with a low initial weight.

Large connected-components of foreground pixels are associated to form targets. Small components are assumed to be noise and removed. Targets are tracked by a bank of kalman filters. A new filter is initialized on each target, provided it is well separated from currently tracked targets. Targets are designated as robot/human. The robot is identified by the colour of its red base.

Once the robot enters the image plane of a camera, the system begins the process of calibrating the image plane to the ground plane. The robot begins a visual servoing procedure [Chakravarty and Jarvis, 2009] that makes it orbit the centre of the image and keeps it within the camera FOV until the calibration process is completed. Simultaneous localization of the robot in the image and ground planes allows the system to collect a set of corresponding points, allowing computation of a homography between image and ground planes. This process is discussed below.

2.3 Camera Calibration using Homography

Homography is used for calibration of the image planes of the fixed cameras to a common ground plane. Let the points (collected during robot movement) in the ground plane be represented using the homogeneous vector $\mathbf{X}= (X, Y, Z)$, and their counterparts in the image plane by $\mathbf{x}= (x, y, 1)$. The coordinates x, y of each target in the image plane are the centre of the base of the bounding-box of each foreground component. Under perspective projection, the points are related by:

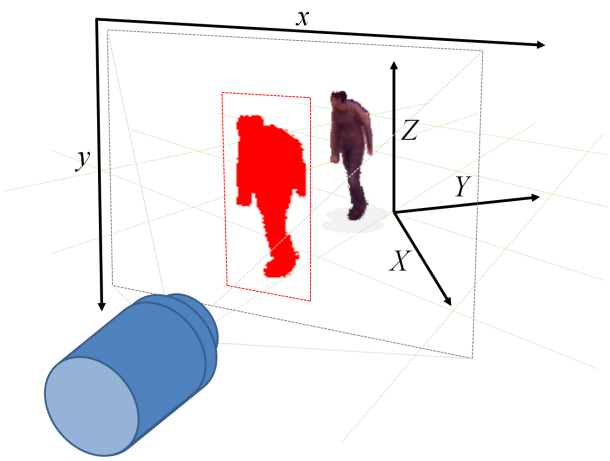

Figure 1: Homography yields a matrix **H** which transforms points x, y in the camera image plane to ground plane coordinates X, Y, Z. Subsequently it is possible to compute the ground-plane position of any moving object, allowing tracking of human targets.

$$\mathbf{X} = \mathbf{H}\mathbf{x} \tag{1}$$

Written in vector notation, we have:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{2}$$

Equation 2 can be re-arranged to give the homogeneous equation system for n points $\mathbf{Ah} = \mathbf{0}$.

To fit the equation system into single column formatting, **A** is split into $\mathbf{A} = [\mathbf{A_1}|\mathbf{A_2}]$, where :

$$\mathbf{A_1} = \begin{bmatrix} x_i & y_i & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_i & y_i & 1 \\ & & \vdots & & & \end{bmatrix}_{2nx6} \tag{3}$$

$$\mathbf{A_2} = \begin{bmatrix} -x_i X_i & -y_i X_i & -X_i \\ -x_i Y_i & -y_i Y_i & -Y_i \\ & \vdots & \end{bmatrix}_{2nx3} \tag{4}$$

and

$$\mathbf{h} = \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \\ h_{33} \end{bmatrix} \tag{5}$$

The elements of the transformation vector **h** are obtained from the eigen vector corresponding to the least eigen value of $\mathbf{A}^T\mathbf{A}$. For each pair of points on the image plane and the ground plane that the robot traverses, we fill up two rows of the matrix A, which becomes a $2n$x9 matrix, where n is the number of points. The Singular Value Decomposition (SVD) of **A**:

$$\mathbf{A} = \mathbf{UDV}^T \tag{6}$$

results in the eigenvalues being arranged in decreasing order along the diagonal of the matrix **D** and the corresponding eigenvectors along the columns of the matrix **V**. The values of the eigenvector associated with the smallest eigenvalue make up the values of the vector **h**.

The set of corresponding points collected during the robot's movement in front of the camera are subject to errors due to transient localization errors in image or ground planes. RANSAC (RANdom SAmpling Consensus) [Fischler and Bolles, 1981] is employed to filter out these erroneous points. RANSAC is normally used when a mathematical model needs to be fitted to a set of points. In this case, the model is the homography matrix **H** (equation 1), whose 9 parameters need to be estimated. The method randomly samples a set of points, calculates the model from these points, and then uses the same model to find the set of inlier points (those points which agree to the calculated model within a pre-defined threshold). A refined model is then calculated from all the inlier points. This process is repeated for either a pre-determined number of iterations or until the reduction in error between successive iterations falls below a threshold.

Once the homography is calculated for a camera, the ground plane location corresponding to any arbitrary point in the image plane can now be found using equation 1. The tracking algorithms discussed in the previous sub-section can now find ground plane positions for targets detected on the image plane and track them on the ground plane (Figure 2).

Tracking on the ground plane gives the trajectories of people moving through an environment in a convenient common format, even during transitions between different camera views. In experiments conducted for

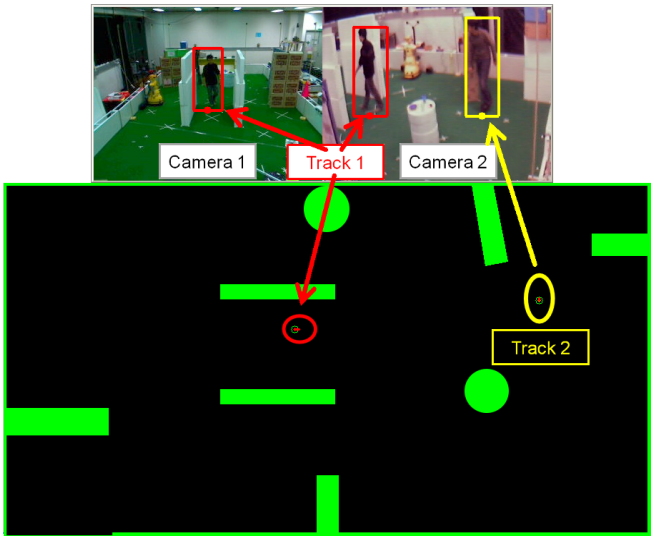

Figure 2: Cameras' homography matrices are found by the mobile robot during a calibration phase. Afterwards, people moving within view of any cameras are tracked relative to a common ground plane system shared by all cameras. A map of obstacles in the environment is provided a priori or generated by the robot during exploration and calibration. In this figure, obstacles are shown in green and free space is black. Two targets are being tracked; one target is currently simultaneously visible in both cameras.

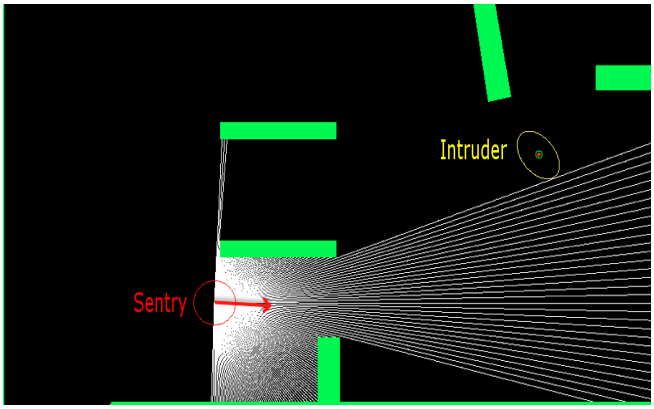

Figure 3: Visibility map of the sentry: ray-tracing is used to determine the field of view of each sentry and thereby the visibility of intruders with respect to sentries.

this paper, we explored two metrics for the analysis of these trajectories. These are discussed in the following section.

3 Covert Behaviour Detection Metrics

Our emphasis is on human targets' trajectories rather than characteristics of their appearance. The latter have been widely studied but are difficult to exploit given typical low resolution overhead cameras that are most commonly used in surveillance systems. Ordinary low quality cameras and typical camera positioning are employed in our experiments. Two metrics are used to determine the overtness of a tracked target at each point on its trajectory. These are described in the following sections.

3.1 Visibility With Respect to Sentry

For each sentry, a visibility map (Figure 3) is used to measure overtness of targets' positions. The map represents the extent of a sentry's field of vision, given his/her position in the environment. Ray-tracing is used to generate the visibility map of the sentry. Rays are extended in angular increments around the sentry's current position until an obstacle is hit; viewing angles are limited to 90 degrees on either side of the sentry's direction of motion. We assume that the sentry is looking the direction it is moving. We also assume that the first confirmed track is the sentry. Many appearance features could be used to identify sentries - particularly robot ones - but for simplicity, in this study we always allow the sentry to enter the environment first. Additional tracked targets are assumed to not be sentries. Generation of the sentry's visibility map is formalized in Algorithm 1.

When non-sentry targets are detected, the visibility map is sampled in a window around the targets to ascertain whether or not they are in the sentry's FOV at that point in time. The number of pixels in this window around the intruder's position that are "hit" by the traced rays is given as the visibility count. This metric is determined at each point of the intruder's trajectory and the visibility measure of the intruder with respect to the sentry at any time is a moving average of this visibility count.

3.2 Visibility With Respect to Environment

A generalized visibility map (first used by [Marzouqi and Jarvis, 2003] for covert path planning for a mobile robot) is constructed to measure the visibility of the tracked suspect with regard to his environment, regardless of the position of the sentry. This reflects the fact that the intruder may not know the trajectories of sentries and instead may simply try to avoid observation. Rays are traced until they hit obstacles in all directions around each unoccupied cell in a map of the environment. The

Algorithm 1 Visibility with respect to Sentry

Definition :
$VisSentry(M, x_s, y_s, \theta_s, d)$
Inputs :
Obstacle map M, a matrix of size $w \times h$ with obstacles given value 1 and pixels representing empty space 0.
x position of sentry in the map x_s,
y position of sentry in the map y_s,
Bearing of sentry in map θ_s
Incremental distance in pixels for ray tracing d
Output :
Completed visibility map with respect to sentry V_s, a matrix of equal dimension to M, representing the sentry's field of view in the environment
Procedure :
Initialize every element in V_s to zero.
$x' = x_s$,$y' = y_s$
for $\theta = \theta_s : \theta_s \pm \pi/2$ **do**
  while $M[x', y']$ is not an obstacle **do**
    $V_s[x', y'] = 1$
    $x' = x' + d \cdot cos(\theta)$
    $y' = y' + d \cdot sin(\theta)$
  end while
end for

Algorithm 2 Generalized Visibility

Definition :
$VisGeneral(M, d)$
Inputs :
Obstacle map M, a matrix of size $w \times h$ with obstacles given value 1 and pixels representing empty space 0.
Incremental distance in pixels for ray tracing d
Output :
Completed generalized visibility map V_g
Procedure :
Initialize every element in V_g to zero.
for x=0:w **do**
  for y=0:h **do**
    if $M[x, y]$ is not an obstacle **then**
      for $\theta = 0 : 2\pi$ **do**
        $x' = x + d \cdot cos(\theta)$
        $y' = y + d \cdot sin(\theta)$
        while $M[x', y']$ is not an obstacle **do**
          $V_g[x, y] = V_g[x, y] + 1$
          $x' = x' + d \cdot cos(\theta)$
          $y' = y' + d \cdot sin(\theta)$
        end while
      end for
    end if
  end for
end for

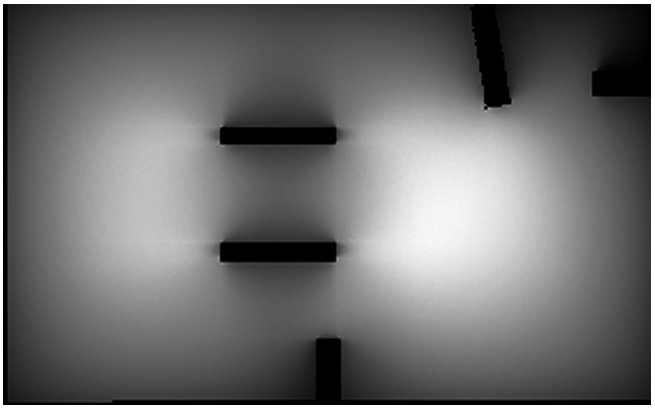

Figure 4: Generalized visibility map of the environment: Darker areas are less visible than lighter areas. Black areas are obstacles.

number of cells that are visible from the current cell are recorded for each cell in the environment. The generation of the generalized visibility map is given in Algorithm 2. This is a time-consuming method (with $O(N^2)$ complexity, where N is the number of free pixels in the environment), but can be done offline and only needs to be computed once. For each position in the environment, the corresponding value in this map (shown in Figure 4) gives a generalized visibility value. Generalized visibility is measured at each point on each intruder's trajectory and is averaged over time, giving the intruder's visibility with respect to the environment for the entire trajectory.

Both these measures are used to determine covertness of intruders in experiments detailed in the next section.

4 Experiments

The objective of the experiment reported in this section is to determine whether there are useful, measurable differences between deliberately covert and normal human behaviour. In these trials the sentries are humans, but they could easily be robots following predefined patrol routes. The metrics used for covert behaviour detection have already been described in Section 3. This experiment also demonstrates effective multi-target tracking between multiple camera views, in a low-traffic environment.

An arena was set up with obstacles offering hiding places. All parts of the arena were visible from at least one of two cameras mounted overhead. The map of the arena and an image to ground plane mapping for each camera were provided to the system. Actors were recruited to play the roles of sentry and intruder. In total 16 trials were conducted (each trial being the traversal of an intruder from one point in the arena to another, while a sentry patrols). In each case the sentries

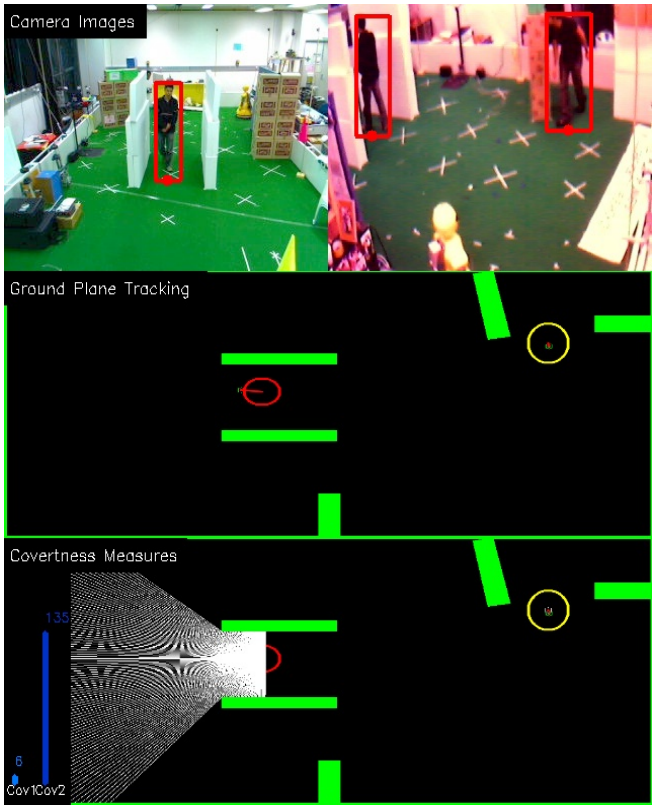

Figure 5: A snapshot of target tracking on the ground plane: Red rectangles indicate the bounding boxes of targets detected in the image planes of the 2 cameras with overlapping fields of view. Red and yellow ellipsoids encircling their ground plane positions show the probability distribution of their tracks (sentry and intruder respectively) on the ground plane. Rays are traced from the sentry's position outwards in a 180 degree arc around his direction of motion to estimate his field of view. Blue bars on the bottom left labelled cov1 (intruder's visibility with respect to sentry) and cov2 (intruder's visibility with respect to environment) show the averaged visibility/covertness measures up to that point in the trajectory of the intruder. The smaller the bars, the more visible their trajectories.

were asked to patrol the environment at a relatively constant speed in all the trials. The intruders were asked to pursue a covert path in 8 trials, and a normal (non-suspicious)/overt path in the other 8. The instructions to the actors were given exactly as detailed below. The cameras were obvious but the actors were not explicitly made aware of them.

Instructions for the covert path actors were as follows: "Move from the START to the GOAL without being seen by the guard. You can crouch and hide behind obstacles. Walk/crawl slowly: You must not run. Instead, plan your route to avoid observation by the guard. You can take as long as you like. The guard will repeat his patrol route continuously. Try to minimise any view of you the guard has, but don't worry if you're seen temporarily. Just keep going."

Instructions for the overt path actors: "Move from the START to the GOAL. Walk slowly: you must not run. You can take as long as you like. You may encounter another subject during your movement. You can choose any route you like."

A screen snapshot of target tracking during the experiment is displayed in Figure 5. Video streams from the 2 cameras show intruder and sentry positions, and the multi-target tracking system described in Section 2 allows ground-plane positions to be shown also. Uncertainty ellipses for each target's filter are displayed.

5 Results

The two measures of covertness of a person walking through the environment (detailed in Section 3) are:

1. Visibility with respect to patrolling sentry.
2. Visibility with respect to environment.

Figure 6 displays the average visibility of the intruder with respect to the sentry over 8 trials. Covert runs are shown in blue, and overt ones in red.

Figure 7 shows the average visibility of the intruder with respect to the environment over 8 trials. Again, blue and red bars are used to show covert and overt trajectories respectively.

It can be seen that covert (blue) trajectories have consistently lower visibility values than overt (red) trajectories, both with respect to the sentry (metric 1) and with respect to the environment (metric 2). There is a marked difference between overt and covert trajectories with respect to sentry visibility. However, with respect to the environment the difference is less distinct. Nevertheless the generalized visibility with respect to the environment was less for every covert trial than any overt trial. Using a single-tailed T-test and assuming the two sets have unequal variance, the p-value obtained for metric 1 is 5×10^{-5} and the p-value for metric 2 is 1.71×10^{-6}. These low p-values lead us to conclude that these results

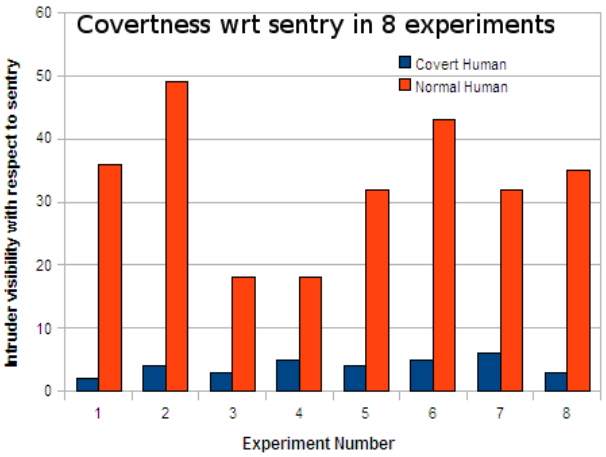

Figure 6: Visibility of intruder with respect to sentry (averaged over trajectory lengths) for 8 covert (blue) and 8 overt (red) trials in our experiment.

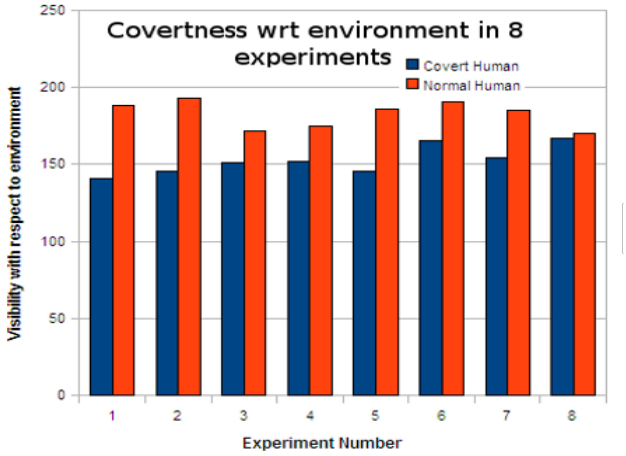

Figure 7: Visibility of intruder with respect to environment (averaged over trajectory lengths) for 8 covert (blue) and 8 overt (red) trials.

reflect changes in behaviour caused by differing intentions of the actors between experimental conditions.

A video clip showing two covert and two overt runs is available at

`http://www.youtube.com/watch?v=pgn88_azriE`

6 Conclusions and Future Work

We have shown that using a network of typical fixed security cameras in a low-traffic indoor environment we can distinguish the behaviour of actors having differing (presumed) intentions. This result implies another conclusion, namely that persons intent on evasion choose to move in a different way to "normal" individuals. However, without a much larger and more strictly controlled experiment in real environments, it is difficult to draw firm conclusions about human behaviour in general.

This paper contributes evidence that the intent to avoid observation changes behaviour in a way that can be autonomously and reliably detected by typical security equipment. While many technologies can help to identify potentially threatening or unauthorised persons, few are able to exploit the low-resolution fixed camera networks that are prolific today. With such data it is hard to model facial appearance, expression, limb movements or other factors that might arouse suspicion. But it is relatively easy to model trajectories.

Although between covert and overt trials differences in general visibility were less marked than differences in visibility with respect to sentries, general visibility is potentially equally or more useful. Mobile sentries - either robot or human - are expensive and therefore coverage is sparse. It is very useful to be able to monitor behaviour in areas not regularly or constantly patrolled by sentries.

A useful extension to this work could be dynamic sentry path planning. Any unknown persons who happened to avoid observation by sentries could cause patrols to be rerouted to intercept. Interception urgency could depend on the covertness of observed behaviour. Robot sentries could be routed to make innocent evasion highly unlikely, and therefore highly suspicious.

The authors also hope to examine human trajectories for other criteria that highlight suspicious persons. For example, acceleration or other moments of motion may be significant. These criteria may not be obvious: Although the authors expected that prolonged proximity to obstacles would be a good indicator of "hiding", ad-hoc observations during our experiment contradicted this. People tend to move close to obstacles for efficiency and convenience, as well as covertness.

This work may have a number of direct applications in low-traffic secure environments, such as public places out of hours, and private business premises. Many false alarms could be eliminated and security improved by

causing the system to autonomously instruct human or robot guards to intercept individuals who have, by chance or design, thus far avoided inspection.

References

[Chakravarty, 2011a] P. Chakravarty *Surveillance & Intervention: Collaboration Between a Robot & Fixed Cameras.* PhD Thesis, Monash University, 2011.

[Chakravarty and Jarvis, 2008] P. Chakravarty and R. Jarvis. People Tracking from a Moving Panoramic Camera. In *Australasian Conference on Robotics and Automation (ACRA)*, 2008.

[Chakravarty, 2011b] P. Chakravarty *Surveillance & Intervention: Collaboration Between a Robot & Fixed Cameras,* chapter Autonomous Calibration of Cameras to Ground, pages 126–136. PhD Thesis, Monash University, 2011.

[Chakravarty and Jarvis, 2009] P. Chakravarty and R. Jarvis. External Cameras & A Mobile Robot: A Collaborative Surveillance System. In *Australasian Conference on Robotics and Automation (ACRA)*, 2009.

[Thrun *et al.*, 2005] S. Thrun, W. Burgard and Dieter Fox. Mapping, *Probabilistic Robotics, pages 279–485*, MIT Press, 2005.

[Marzouqi and Jarvis, 2003] M. Marzouqi and R. Jarvis. Covert Path Planning for Autonomous Robot Navigation in Known Environments. In *Australasian Conference on Robotics and Automation (ACRA)*, 2003.

[Hu *et al.*, 2004a] W. Hu, T. Tan, L. Wang and S.J. Maybank. A Survey on Visual Surveillance of Object Motion and Behaviors. In *IEEE Transactions on Systems, Man and Cybernetics, pages 334–352, Vol. 34*, 2004.

[Hu *et al.*, 2004b] W. Hu, D. Xie, T. Tan and S.J. Maybank. Learning Activity Patterns using Fuzzy Self-organizing Neural Network. *IEEE Transactions on Systems, Man and Cybernetics, pages 1618–1626, Vol. 34*, 2004.

[Brand *et al.*, 1997] M. Brand, N. Oliver and A. Pentland. Coupled Hidden Markov Models for Complex Action Recognition. In *Computer Vision and Pattern Recognition (CVPR), pages 994–999*, 1997.

[Natarajan and Nevatia, 2007] P. Natarajan and R. Nevatia. Coupled Hidden Semi Markov Models for Activity Recognition. In *IEEE Workshop on Motion and Video Computing, Vol. 0*, 2007.

[Nguyen *et al.*, 2005] N.T, Nguyen, D.Q. Phung, S. Venkatesh and H. Bui. Learning and Detecting Activities from Movement Trajectories using the Hierarchical Hidden Markov Model. In *Computer Vision and Pattern Recognition (CVPR), pages 955–960, Vol. 2*, 2005.

[Bui *et al.*, 2008] H. Bui, D. Phung, S. Venkatesh and H. Phan. The Hidden Permutation Model and Location-Based Activity Recognition. In *23rd National Conference on Artificial Intelligence (AAAI), pages 1345–1350, Vol. 3*, 2008.

[Dee and Hogg, 2004] H.M. Dee and D.C. Hogg. Detecting Inexplicable Behaviour. In *British Machine Vision Conference (BMVC), pages 477–486*, 2004.

[Dee and Hogg, 2005] H.M. Dee and D.C. Hogg. Navigational Strategies and Surveillance. In *IEEE International Workshop on Visual Surveillance, pages 73–81*, 2005.

[Stauffer and Grimson, 2000] C. Stauffer and W. Grimson Learning Patterns of Activity using Real-time Tracking. In *IEEE Transactions on Pattern Recognition and Machine Intelligence (TPAMI), pages 747–757, Volume 22*, 2000.

[Fischler and Bolles, 1981] M. Fischler and R. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. In *Communications of the ACM, pages 381–395, Vol. 24*, 1981.