Socially Constrained Tracking in Crowded Environments Using Shoulder Pose Estimates

Alexander Virgona*, Alen Alempijevic*, and Teresa Vidal-Calleja*

*Centre for Autonomous Systems
University of Technology Sydney, 15 Broadway Ultimo, NSW 2007
Email: alexander.virgona@student.uts.edu.au

***Abstract*—Detecting and tracking people is a key requirement in the development of robotic technologies intended to operate in human environments. In crowded environments such as train stations this task is particularly challenging due the high numbers of targets and frequent occlusions. In this paper we present a framework for detecting and tracking humans in such crowded environments in terms of 2D pose (x, y, θ). The main contributions are a method for extracting pose from the most visible parts of the body in a crowd, the head and shoulders, and a tracker which leverages social constraints regarding peoples orientation, movement and proximity to one another, to improve robustness in this challenging environment. The framework is evaluated on two datasets: one captured in a lab environment with ground truth obtained using a motion capture system, and the other captured in a busy inner city train station. Pose errors are reported against the ground truth and the tracking results are then compared with a state-of-the-art person tracking framework.**

I. Introduction

Detecting and tracking human body pose is a key requirement for robots and intelligent systems that operate in human environments, enabling them to interact with people and make sense of human behaviour. A system with this capability could, for example, recognise when individuals are interacting with one another and model social connections.

The task of detecting and tracking the human pose is challenging, as human environments are typically dynamic and unstructured, and people come in a variety of shapes, sizes, and appearances. The difficulty of this task is further increased in crowded environments due to frequent visual occlusions, and the number of targets to be tracked. An inner-city train station is a prime example of such an environment. The framework presented in this paper is aimed towards developing an intelligent system capable of sensing the behaviour of commuters in an inner-city train station.

Person detection and person tracking are both mature research areas with researchers applying sensing modalities, such as laser range finders, colour cameras, and more recently depth cameras, to detect and estimate the pose of people in a variety of scenarios. Whilst recent work on full body pose estimation using depth cameras [1], [2], [3], [4] has shown impressive results, the density of people in crowded environments and the frequency of occlusions makes reliably observing the whole body very difficult. This difficulty has caused several authors [5], [6] to focus on the parts of the body that are most visible in crowded environments, i.e. the head and shoulders. Although the pose of the head and shoulders alone is less informative than a full body pose, it still provides rich information about people's behaviour in the target environment, such as whether a person intends to board a specific train.

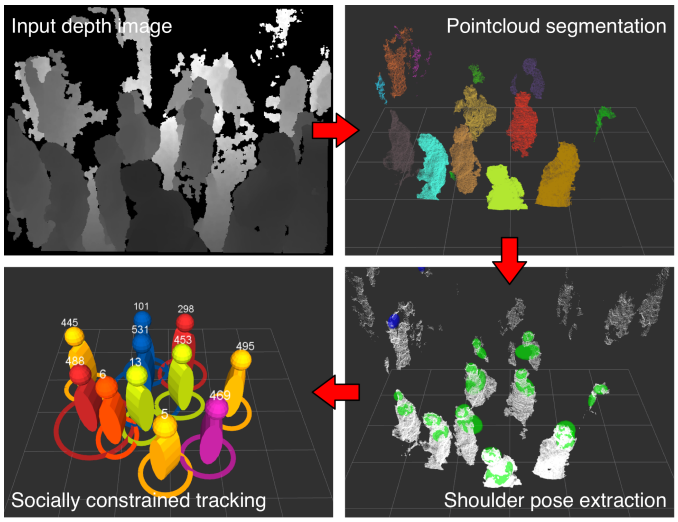


Fig. 1. Depth images (top-left) are converted to a pointclouds, aligned with the floor, and segmented into proposal pointclouds (top-right). The pose of each pointcloud is extracted by fitting ellipsoids (green) to the head and shoulders (bottom-right). Each extracted pose is input to a socially constrained tracker to produce a set of filtered poses with persistent tracking identifiers (bottom-left).

In this paper we present a framework for detecting and tracking the shoulder pose of multiple people in a crowded environment from a stream of depth images, in real-time. Shoulder pose is defined here as a position (x, y), and an angular orientation θ, about the vertical axis. The framework is organised into three main stages, as depicted in Figure 1. The *scene segmentation* stage receives depth images and outputs proposal pointclouds which potentially describe people. The *shoulder pose extraction* stage fits ellipsoids to the head and shoulder region of each proposal pointcloud to extract a shoulder position and orientation, discarding those that do not conform to shape and size constraints. Finally the *socially constrained tracking* stage leverages proxemics, the study of personal space in the context of non-verbal communication, to improve the robustness of tracking in crowded scenes.

The remainder of the paper is structured in order of importance rather than following the framework from start to finish. Section II provides an overview of relevant related work. This is followed by descriptions of the two major contributions: a socially constrained tracking algorithm in Section III; and a shoulder pose extraction algorithm in Section IV. In Section V, the scene segmentation front-end is described for completeness. Section VI contains the empirical evaluation of the framework on lab and real-world datasets. Finally, Section VII provides some conclusions on the work and intended future research directions.

II. Related Work

Tracking people in crowded environments requires robust person detection and data association capabilities. Significant work has been completed on tracking people using a monocular RGB camera, for instance [7], [8], and more recently exploiting RGB-D cameras [9]. Some powerful approaches to 3D human pose estimation have emerged recently [2], [9], however they often rely on significant visibility of each individual, which is not possible in dense environments.

In the context of public spaces, systems that use RGB cameras may raise privacy concerns given the degree of personal information inherent in this type of data. In the case of a system which tracks the movements of individuals, and which could be capable of linking people's behaviour with their identity, these concerns are likely to be amplified. It follows that the real-time tracking of individuals that does not capture superfluous personal information, is preferable to rail operators interested in optimising their operations through the analysis of commuter movement. As such, our work avoids the use of RGB data.

In [6], 3D depth sensors are used for person tracking but require specific placement combined with overlapping views to achieve complete coverage of large spaces, such as shopping centers. Though this work demonstrates the potential of 3D sensors for large-scale person tracking, the requirement for overlapping, overhead placement of sensors, limits the practicality of such an approach for our application.

In line with the above considerations, instead of leveraging additional information from RGB data such as [10] and [8], our prior work exploited the representation of a person solely in depth data through the head to shoulder region [11]. This region remains visible even in densely crowded environments with interactions, evident in [8] where crowding estimation in proximities of train stations is achieved using detection of people's heads. Descriptive information of the head to shoulder region was combined with tracking of the person's position using an Event Graph to achieve the data association when significant occlusions or interactions were present. However, the work in [11] did not leverage any knowledge of social norms that govern personal interactions.

III. Socially Constrained Tracking

This work aims at exploiting social constraints within a tracking framework. Given a stream of shoulder pose observations, the tracking algorithm presented leverages an understanding of proxemics, to constrain predictions and provide more robust tracking in crowded environments. A particle filter is an attractive framework for this task because its flexibility towards incorporating arbitrary prediction and observation models allows us to easily introduce such social constraints.

Shoulder poses are tracked using a particle filter per person, where each filter maintains a collection of particles $\mathbf{X}_t^p = \left[\mathbf{x}_t^1, \cdots, \mathbf{x}_t^N\right]$ and a corresponding set of weights $\mathbf{w}_t^p = \left[w_t^1, \cdots, w_t^N\right]$ representing the distribution over possible states of person p at time t. Each particle $\mathbf{x}_t^i = \left[x, y, \dot{x}, \dot{y}, \theta\right]^\intercal$ represents a possible state in terms of position x, y, velocity $\dot{x}, \dot{y}$ and orientation about the z-axis θ. The number of particles per filter N is selected as a trade-off between computational cost and better expression of the underlying distribution ($N = 500$ in our experiments). Although the pose extraction method described in Section IV-A is capable of extracting the 3D position of the shoulders, our tracker only operates in the the horizontal plane, ignoring height, as this is sufficient to maintain persistent tracks of people in our target environment.

At each iteration of the tracker, triggered either by a new frame of data or by time elapsed without new data, the tracker performs the following steps:

1) **Socially Constrained Prediction –** The state of each filter $\mathbf{X}_t^p$ is predicted based on its previous state $\mathbf{X}_{t-1}^p$, the assumed motion model and social constraints.
2) **Two-Tier Data Association –** Pose observations are associated to existing filters based on a two-tiered, nearest neighbours approach.
3) **Observation Update –** Particle weights are updated based on the likelihood of the associated pose observations, taking into account a measure of confidence in the orientation estimate.
4) **Shoulder Orientation Update –** Particle weights are updated based on correlation between shoulder orientation and velocity direction.
5) **Resampling –** Particles in each filter are systematically resampled to represent the weighted particle distributions as equivalent uniformly weighted particle distributions.
6) **Track Initiation and Deletion –** New tracks are created for unassociated observations and tracks are deleted based on covariance or missed observations.

A. Socially Constrained Prediction

At each time step, the position x, y and velocity $\dot{x}, \dot{y}$ of each particle are propagated according to a constant white noise acceleration model [12], and the orientation θ treated independently

$$\mathbf{x}_t^i = \mathbf{F}\mathbf{x}_{t-1}^i + \nu_\mathbf{t} \qquad \nu_\mathbf{t} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{Q}\right)$$

with state transition matrix $\mathbf{F}$ and covariance $\mathbf{Q}$ of process noise $\nu_\mathbf{t}$ as follows:

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & t & 0 & 0 \\ 0 & 1 & 0 & t & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{Q} = \begin{bmatrix} \frac{q}{3}t^3 & 0 & \frac{q}{2}t^2 & 0 & 0 \\ 0 & \frac{q}{3}t^3 & 0 & \frac{q}{2}t^2 & 0 \\ \frac{q}{2}t^2 & 0 & qt & 0 & 0 \\ 0 & \frac{q}{2}t^2 & 0 & qt & 0 \\ 0 & 0 & 0 & 0 & \sigma_\theta^2 \end{bmatrix}.$$

The parameter q is the process noise intensity, and is selected based on expected variation in acceleration of tracking targets

in the desired application. Similarly the angular variance σ_θ^2 must be selected based on expected variations in people's orientation.

In crowded environments, the close proximity of people to one another can cause tracks to erroneously change targets. This problem occurs when the separation between two or more targets is small compared to the observation error and is exacerbated in cases where the targets have similar velocities. To improve the robustness of our tracker to crowding, we implement simple social constraints based on the study of proxemics [13] which describes people's inclination to maintain comfortable interpersonal distances from one another, even in crowded situations.

This social constraint is enforced in the prediction step by deleting and redrawing any particles drawn which violate these interpersonal constraints. First the prediction step is performed without social constraints to update the mean hypothesis of each filter. Based on the predicted states we then check for *invalid* particles in any filter, i.e. those within a fixed radius of any other filter mean position. *Invalid* particles are redrawn until *valid*, or until the number of redraw attempts reaches an upper limit (set to 3 in our experiments). Redrawing invalid particles maintains the desired expressiveness of the filter by maintaining closer to the nominal number of particles. Limiting the number of redraw attempts on the other hand ensures that the algorithm does not become impractically slow in extreme crowding.

B. Two-Tier Data Association

On each tracking iteration the data association step attempts to match each observation with its respective filter so that the state of these filters can be updated. Following this association process, any unmatched filters could be considered candidates for deletion, while unmatched observations may be candidates for the creation of new filters.

Naively we might choose to initiate a new filter for every such candidate, however in a crowded environment new filters created due to false positive observations are likely to "steal" legitimate observations from existing filters in subsequent data association steps and erroneously modulate the prediction step with regard to the social constraints described in Section III-A.

To avoid this issue we have a two tiered approach to data association. When a new filter is created it is labeled *invalid* and an associated *validation counter* is initialised to zero. On every iteration of the tracker the validity counter of unassociated filters is decremented, while the validity count of associated filters incremented. If the validity count reaches the negatively-valued, lower validity threshold, the filter is deleted for lack of observations. If the validity count reaches the positively-valued, upper validity threshold, the filter is permanently promoted to *valid*. By this mechanism if a filter is associated in more than 50% of frames it will progress towards *valid* status, while if associated in fewer than 50% of frames it will progress towards deletion.

Given these labels, the data association is performed in two stages. First the set of *valid* filters are each matched with the nearest observation that is statistically consistent with the distribution of positions represented by its particles, with 95% confidence according to the Chi-squared test. This matching is done in a greedy fashion whereby the nearest consistent pair is matched at each iteration and removed from further consideration until there are no consistent pairs remaining. This same process is then followed for the remaining unassociated observations and the set of *invalid* filters. In this way, confirmed *valid* tracks are given precedence over newly created tracks in the data association, and are therefore less likely to be adversely affected by false positive observations.

C. Observation Update

Following data association, each filter is updated to incorporate information from the associated observation into its distribution of possible states. This update is achieved by multiplying each particle's weight by the likelihood of its state given the associated observation, or equivalently, by the probability of the observation $\mathbf{y}_t^j$ given the state $\mathbf{x_t^i}$, and then normalising by the sum of the resulting weights

$$w_t^i = \frac{p(\mathbf{y}_t^j \mid \mathbf{x_t^i}) w_{t-1}^i}{\sum_{i=1}^N p(\mathbf{y}_t^j \mid \mathbf{x_t^i}) w_{t-1}^i} \,.$$

The likelihood function is the product of two terms: one concerning the 2D position L_{xy} and the other concerning the orientation L_θ

$$p(\mathbf{y}_t^j \mid \mathbf{x_t^i}) = L_{xy} L_\theta \,.$$

The term L_{xy} is the probability of observing the Euclidean position error δ_{xy} assuming a Gaussian sensor model with zero mean and variance σ_{xy}^2

$$L_{xy} = p(\delta_{xy} \mid 0, \sigma_{xy}^2) \,.$$

The shoulder poses extracted by our algorithm have some ambiguity in their orientation, between forwards and backwards, discussed further in Section IV-A. To deal with this ambiguity, the term L_θ is a sum of two components: one relating to the angular error of the detected orientation δ_θ, and the other relating to the angular error of the opposite orientation $\delta_{\theta+\pi}$

$$L_\theta = \beta p(\delta_\theta \mid 0, \gamma\sigma_\theta^2) + (1-\beta) p(\delta_{\theta+\pi} \mid 0, \gamma\sigma_\theta^2) \,.$$

The balance between components is controlled by the ambiguity ratio β which describes the proportion of pose observations expected to have the correct facing direction. Based on empirical evaluation of our pose extraction algorithm we use $\beta = 0.7$ in our tracking experiments.

Additionally the shape of the ellipsoid fit to the shoulders in the *shoulder pose extraction* stage (Section IV) gives an indication of the quality of the extracted orientation. If the ellipsoid fit is spherical, the extracted orientation is arbitrary and therefore uninformative, however if the ellipsoid is narrow it is likely to provide a more reliable orientation measurement. To reflect this a variable noise sensor model is used to calculate L_θ.

An orientation confidence measure γ is computed based on the eccentricity of the shoulder ellipsoid ϵ and used to scale the variance σ_θ^2 of the orientation observation model. The eccentricity is defined as the ratio of the shortest radius of the ellipse over the longest and can therefore have values in the interval $(0, 1]$. The orientation confidence is given by:

$$\gamma = \max\left(\frac{1-\epsilon_0}{1-\epsilon}, 1\right)$$

hence $\gamma \in [1, \infty)$ but is capped in our implementation to a suitable maximum value to avoid numerical issues.

D. Shoulder Orientation Update

To enforce the social constraint that people tend to align their shoulders with their walking direction, a pseudo-observation update is applied on each iteration of the tracker. This update is similar to the observation update described in Section III-C in that the weights of particles are updated based on a likelihood function $L_{\theta v}$, however it differs in that it is not based on any actual observation and is applied to all filters regardless of whether they have any associated observations. The likelihood is computed as

$$L_{\theta v} = p(\delta_{\theta v} \mid 0, \sigma^2_{\theta v}),$$

$$\delta_{\theta v} = \begin{cases} \theta - \operatorname{atan2}(\dot{y}, \dot{x}) & \text{if } v > v_0 \\ 0 & \text{otherwise} \end{cases},$$

where $v = \sqrt{\dot{x}^2 + \dot{y}^2}$. For particles where $v > v_0$ this has the effect of lowering their weight when their velocity direction is not aligned with their shoulder orientation.

This pseudo-observation has the effect of correlating shoulder orientation and walking direction in the particle distribution of each filter. This allows the walking direction of the person to refine the estimated orientation of each person when they are moving with sufficient velocity particularly with regard to the ambiguity of shoulder orientation estimates between forwards and backwards facing directions. Additionally when people transition from stationary to moving, which is often a challenge for tracking systems with a single motion model, the estimated orientation allows the tracker to better predict the persons new velocity.

IV. Shoulder Pose Extraction

The main input to the tracker is a set of shoulder pose observations. To extract the pose of each person, we aim to fit a model of the visible surface of the head and shoulders to pointcloud data of the upper body. The surface model should: (1) be similar enough to the shape of human head and shoulders to provide a good fit, (2) allow extraction of a stable shoulder position and orientation, (3) be flexible enough to encompass the variety of shapes and sizes within the population, and (4) be robust to relative motion between the head and shoulders. With these requirements in mind a pair of ellipsoids was chosen as a suitable surface model: one fitted to the head, and one fitted to the shoulders, as shown in Figure 2.

A. Extracting Shoulder Pose Using Ellipsoids

Each of the pointclouds output by the *scene segmentation* stage (Section V) is provided as input to the pose extraction stage which extracts a shoulder pose comprised of a 3D position and angle of orientation about the vertical axis. In order to fit ellipsoids specifically to the head and shoulders, candidate points must be selected from the pointcloud which are likely to represent these parts of the body.

This task requires us to make some assumptions about the size and shape of the head and shoulders of a person, and is made challenging by the wide range of sizes and shapes within the population. To guide these assumptions we have used statistical data taken from a 2012 Anthropometric Survey Of U.S. Army Personnel [14] to set physical selection criteria where needed. The surveyed personnel consisted of both genders and a broad range of occupations, not only those on the front line. We also fit the ellipsoids sequentially to leverage parameters of the head ellipsoid in selecting candidate points for the shoulder fit, hence adapting our physical criteria to the individual and reducing the sensitivity of the method to the chosen parameters.

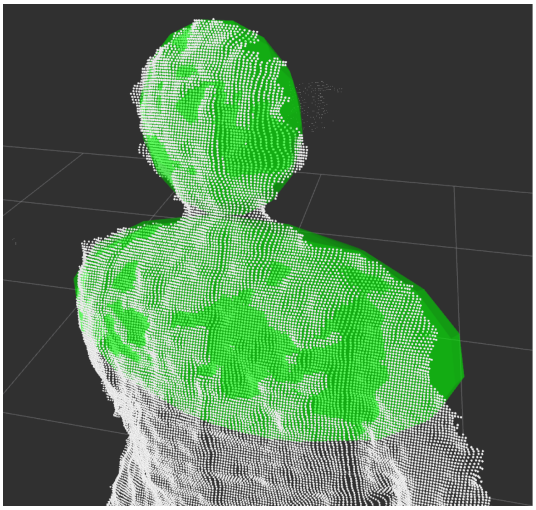


Fig. 2. A pair of ellipsoids representing the head and shoulders (green) fit to a pointcloud of a person(white).

First the head ellipsoid is fit to a vertical window of points with fixed height extending downward from the top of the pointcloud. A window size of 21cm is used based on the 10th percentile measurement from the top of head to the cervicale [14] to capture most of the points on the head while minimising the chance of including the neck or shoulders. The shoulder ellipsoid is similarly fitted to a fixed vertical window of points, extending 21cm downward (90th percentile neck to scye length [14]) from the centre of the head ellipsoid. However, to ensure that the shoulder ellipsoid fits the breadth of the shoulders rather than the neck area, a dilated copy of the head ellipsoid is used to remove the neck and collar region from the points to be fit. This ensures that the fit is dominated by the shoulder tips, improving the quality of orientation estimates obtained.

Once the head and shoulder ellipsoids have been fitted they are used to extract a shoulder pose consisting of a 3D position and angle of orientation about the vertical axis. The horizontal components of the pose are taken directly from the centre of the shoulder ellipsoid as this position is more stable than that of the head. However the vertical component of the shoulder ellipsoid is less stable due to its high dependance on the vertical window used to select points for the fit. For this reason the vertical component of the pose is based on the top surface of the shoulder ellipsoid as it is more indicative of the true height of the persons shoulders, it is calculated by the intersection between a vertical line passing through the ellipsoid centre and the upper surface of the ellipsoid.

Finally the orientation of the shoulders is obtained by projecting the major axis of the shoulder ellipsoid into the horizontal plane and taking the angle of the resulting line. This angle is rotated 90°to obtain the facing direction of the person rather than the line of their shoulders, however the forwards direction is ambiguous based on the axis of the shoulders alone. To resolve this ambiguity we make the assumption that the head is forward of the shoulders. The horizontal location of the head ellipsoid centre relative to the shoulder ellipsoid major axis is used to determine the forwards facing direction and set the orientation angle accordingly. While this method for disambiguating the facing direction works 70% of the time, the possibility of obtaining the opposite direction is also explicitly handled in our tracking algorithm.

B. Ellipsoid fitting

In the crowded scenarios targeted by this work, the number of people in the field-of-view (FOV) of the sensor at any time can be upwards 20. With 2 ellipsoids to be fitted per person and 30 frames of depth data per second this could mean the fitting of as many as 1200 ellipsoids per second. In order to process all data in real-time it was therefore a priority to use an efficient method for ellipsoid fitting.

The ellipse fitting method used, proposed by Li et al. [15], finds the least squares fit of a quadric surface of the form

$$ax^2+by^2+cz^2+2fyz+2gxz+2hxy+2px+2qy+2rz+d=0$$

to a set of 3D points subject to the constraint $4J - I^2 > 0$ where:

$$I = a + b + c\,,$$

$$J = ab + bc + ac - f^2 - g^2 - h^2\,.$$

Li et al. [15] show that this constraint is sufficient to guarantee that the quadric surface fit is an ellipsoid, and the problem can be efficiently solved by formulating it as an eigensystem.

V. Scene Segmentation

Prior to extracting shoulder poses as described in Section IV a scene pre-processing stage is required. This stage converts each frame of depth data into an upright, foreground pointcloud and subsequently segments it into spatially separated clusters. The foreground pointcloud contains only points which are not part of the static environment and is transformed such that the $z = 0$ plane is aligned with the ground plane of the scene. The benefits of this are: (i) the chance of false positive person detections is reduced by removing the static environment from consideration, (ii) the amount of computation required in subsequent stages is reduced by drastically lowering the number of pixels processed, and (iii) downstream clustering and pose estimation algorithms are simplified thanks to alignment of the pointcloud to the floor. The scene processing stage consists of background subtraction, ground-plane alignment and spatial segmentation.

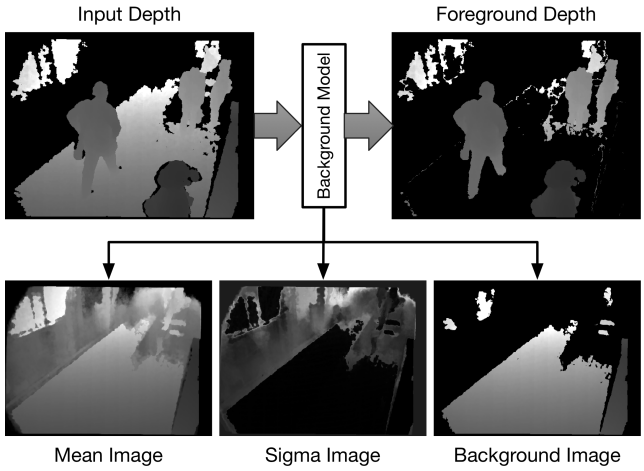


Fig. 3. Each frame of input depth data is compared to the current background image in order to mask out background pixels and output the foreground depth image. At regular intervals the input depth image is used to update the background model.

A. Background Subtraction

Background subtraction segments parts of the depth image potentially describing people from those representing the static environment. A model of the static background is learned incrementally from the depth data and used to mask out pixels of each depth image consistent with the model, leaving only those considered to describe the foreground as illustrated in Figure 3.

The background model is learned based on the ideas presented in [16], in which the expected value of the background at each pixel in a colour or greyscale scene is modelled as a mixture of Gaussians. In contrast to [16] we are able to greatly simplify the approach to use a single Gaussian per pixel, owing to the inherent relevance of depth information to the task of background modelling.

B. Ground-Plane Alignment

After allowing an initial burn-in time for the background image to be established it is projected into a pointcloud representation and a plane is fit using Random Sample Consensus (RANSAC) [17]. It is assumed here that the dominant plane represents the floor. In all subsequent frames the foreground depth image is projected into a pointcloud and transformed using the established floor plane such that the $z = 0$ plane is aligned with it.

C. Spatial Segmentation

Pointclouds are segmented into human sized clusters based on the following assumptions:

1) People stand with the length of their body perpendicular to the floor.
2) The tallest point on a persons body is their head.
3) Peoples heads are spatially separated from one another.

The segmentation algorithm sorts the pointcloud in descending height order, then iterates through each point p_i, comparing the horizontal distance d_{ij} between each point p_i and each cluster C_j to a fixed separation distance threshold

Fig. 4. *Left:* Our sensor platform (top) was mounted on a fixed pole aimed at the centre of the room and the movement of all participants was tracked using infrared marker cards (bottom). *Right*: A sample depth image taken from the *Passing* sequence.

d_0. If $d_{ij} \leq d_0$ the point is added to the nearest cluster and the mean of the cluster is updated, otherwise a new cluster is created containing only p_i.

After clustering there are often cases where a person is split into multiple clusters due to points at a person's horizontal extremities, such as their shoulders for which $d_{ij} > d_0$. To deal with this occurrence a final cluster joining step checks the distance between cluster means and joins those with a distance less than d_0 in a greedy fashion. Finally any clusters with a small number of points or representing a small surface area are removed.

VI. Empirical Evaluation

In order to evaluate the performance of our framework, we conducted experiments on data collected both in lab environments and in the wild. In this section we explain our data collection, present results and provide discussion of these experiments. The parameters used in our experiments were determined empirically but could be optimised if desired.

A. Sensor placement

The choice of sensor placement for data collection was primarily determined by the requirements of the urban rail environment. The sensors were placed high enough to avoid the risk of injury to commuters, meanwhile the upper height limit was determined either by the ceiling height or practicality of mounting procedure. Once the height was determined the sensor angle was chosen to ensure the floor was visible for ground plane alignment as described in Section V-B.

B. Lab dataset with ground truth

In order to quantify the precision and accuracy of our approach, a dataset was captured consisting of 9 depth image sequences of people moving in different ways through the depth sensor FOV, with accompanying ground truth measured using an optical motion capture system. The dataset was captured in the UTS Data Arena, a circular cinema room, with an Optitrack motion capture system comprising of high frame rate cameras with infrared illumination. Each participant had a rigid infra-red marker card attached to their back using a velcro strap (pictured in Figure 4), used to accurately track the position and rotation of their upper body. A brief description of the different depth sequences is provided below.

TABLE I
Mean Absolute Errors of Pose Extraction Against Ground Truth From an Optical Tracking System

Dataset	Horizontal (cm)	Vertical (cm)	Orientation (°)	Forwards (%)
Wandering 1	8.37	3.34	13.99	70.59
Wandering 2	7.36	3.10	10.40	72.20
Wandering 3	9.40	3.63	14.29	84.74
Wandering 4	9.02	3.44	12.37	82.12
Alighting 1	7.82	5.37	10.00	77.59
Alighting 2	8.97	3.95	9.10	71.37
Alighting 3	16.80	5.93	18.55	74.74
Walkthrough	8.37	3.99	12.03	77.88
Passing	12.93	5.67	14.85	80.26

Wandering 1/2/3/4 – Participants casually moving and stopping within the FOV of the depth camera (3/4/8/8 people, 120/123/47/98 seconds).

Alighting 1/2/3 – Participants simulating situations where 4 train passengers wait to board a service while 2 passengers alight. (6 people, 24/21/16 seconds).

Walkthrough – 4 participants stand still while 4 others repeatedly cross the FOV weaving between stationary participants. (8 people, 142 seconds).

Passing – All participants repeatedly crossing the FOV weaving past one another (8 people, 131 seconds) (pictured in Figure 4).

C. Pose extraction precision

In evaluating the shoulder pose estimation algorithm presented, depth image sequences from the lab dataset were processed using our framework and the results of pose extraction, prior to tracking, were compared with the pose ground truth obtained from the motion capture system. Table I summarises the results of this comparison with in terms of precision, horizontally, vertically and in orientation angle.

To account for the arbitrary offset between infra-red markers attached to participants and the centre-of-shoulder position extracted by our algorithm, a single 3D offset has been applied to the ground truth data in the local frame of each marker card based on the mean 3D position error. The results therefore do not capture any positional bias in the extracted poses but do capture the consistency of the extracted poses which is more important in the target scenario. The starting orientation of the marker cards is also arbitrary and a similar offset has been applied to each card orientation prior to error computation. Orientation errors are also wrapped between $\pm\frac{\pi}{2}$ to better characterise errors in the face of ambiguity between the forwards and backwards direction. For clarity the percentage of extracted poses which correctly estimated the forwards direction (and hence did not require wrapping) are also given.

D. Tracking comparison

In order to evaluate our tracker we used person detection and pose extraction from our framework and passed them into both our tracker and a state-of-the-art person tracker [18]. The results of this comparison are presented in Table II in terms of the CLEAR-MOT metrics [19], a system of metrics devised to enable intuitive and fair benchmarking of multiple object tracking systems. The metrics are *multiple object track*

TABLE II
COMPARISON BETWEEN OUR SOCIALLY CONSTRAINED TRACKER AND [18] ON THE CLEAR-MOT [19] PERFORMANCE METRICS

Dataset	MOTP (cm)		MOTA (%)	
	Ours	[18]	Ours	[18]
Wandering 1	**70.51**	71.93	**98.90**	97.77
Wandering 2	**69.15**	70.90	**98.77**	97.39
Wandering 3	**66.99**	68.04	92.60	**94.24**
Wandering 4	**69.07**	70.38	90.67	**91.86**
Alighting 1	**68.47**	68.63	**59.04**	52.39
Alighting 2	**68.83**	70.76	**56.37**	46.65
Alighting 3	63.39	**61.96**	**53.94**	46.74
Walkthrough	65.78	**64.32**	**63.63**	57.43
Passing	68.24	**64.29**	**35.72**	26.45

precision (MOTP), a measure of the average position error in real units, and *multiple object tracking accuracy* (MOTA), the percentage of accurate tracking outputs.

Our tracker performed similarly well to the tracker from Linder et al. [18] in the *Wandering* sequences, which is unsurprising as both trackers are provided with the same pose detections and use very similar motion models. Interestingly our tracker performed better in terms of MOTA for the *Alighting*, *Walkthrough* and *Passing* sequences all of which involve movement of people through a densely crowded area in close proximity to others. This improvement can be attributed to the addition of social constraints in track prediction which significantly narrow the spread of particle states in crowded scenarios by avoiding predictions in close proximity to others. Note that MOTA scores of both trackers are low due to time participants spent outside the depth sensor FOV but still visible to the optical tracking system. However the comparison between the trackers remains fair.

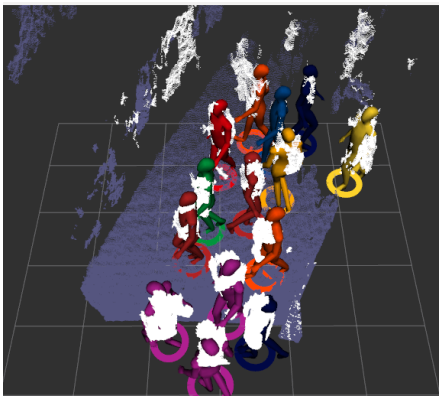
Poorer MOTP scores are likely due to the pose detections rather than either tracker. Two major sources of error exist which have not been accounted for in these results: (1) the arbitrary offset between the rigid marker placed on the back of subjects and their shoulder-centre, (2) significant scale errors in the depth values reported by our depth camera. The first of these is simply the result of our ground truth data collection method and cannot be eliminated, the second could be addressed in future by calibrating for depth scaling using one of several published methods [20], [21].

E. Performance on a Crowded Train Platform

To evaluate the success of our approach in the intended scenario we captured depth image sequences at a busy inner-city train station (Sydney's Town Hall Station) using purpose built sensing platforms, pictured in Figure 5a. Over the course of three days, depth images were recorded at 30Hz in crowded train platform areas. Figure 5b shows an example frame from the depth data recorded.

The performance of the system on this challenging real world dataset can be assessed qualitatively in the accompanying video showing a 3D visualisation of the tracking results. Figure 5c shows a still frame from this visualisation. Additionally, in order to provide a quantitative evaluation of the system's performance, we compare the number of active tracks in each frame of tracking data output by the system, against a manual person count taken directly from the depth

(a) (b)



(c)

Fig. 5. Our framework tracking people on a busy train platform at Town Hall Station, Sydney. (a) A depth image from the Town Hall dataset. (b) Visualisation of tracking output for the same frame. The background pointcloud (blue), foreground pointcloud (white) and coloured human avatars (courtesy of the Spencer rviz plugin [18]) show the tracking result.

image sequence. While more detailed quantitive results would be desirable, it is infeasible to collect accurate ground truth without marking participants, which is not possible in this public scenario.

We show results in Figure 6 on a difficult 30 minute sequence taken from the morning commuter rush hour. Despite crowds of up to 26 people in the sensors FOV the total number of people reported by the system is accurate to within 2 people most of the time with some bias towards overestimation. This tendency to overestimate total numbers of people can be explained by the lingering of tracks after people leave the sensors FOV. While the manual count will immediately decrement, our tracker maintain its hypothesis of the persons location until the position covariance reaches an upper threshold. The highest errors in the total person count occur in periods of sharp increase or decrease of the ground truth person count. This is attributed to the tracker's tendency to lag behind the true count, not only in deletion of tracks as mentioned above but also in the initiation of new tracks due to the requirement for multiple consistent measurements before confirming tracks as *valid*, discussed in Section III.

VII. CONCLUSION

To address the challenges of estimating and tracking human pose in crowded environments, we have presented a framework of components including a novel method for shoulder pose

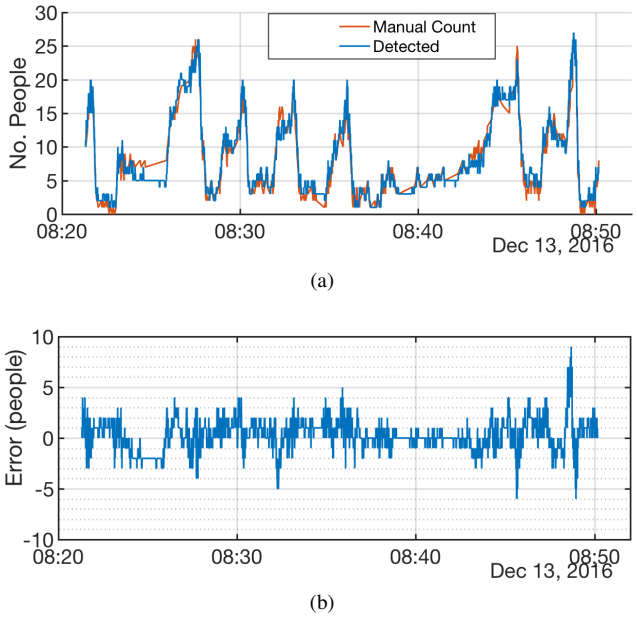


Fig. 6. (a) Total person count detected (blue) and manual count (red) on a sequence of depth images during morning peak at Sydney Town Hall Station. (b) Error in person count compared with manual count

extraction and a tracking algorithm which exploits social constraints to improve track prediction. We have evaluated our approach on a dataset with accurate ground truth demonstrating the precision of our pose extraction technique. We have also shown favourable results in terms of tracking accuracy when compared with recent work in person tracking, particularly in densely crowded scenarios where our socially constrained tracker improves track prediction.

One limitation of this work is poor orientation estimation at long range (>4m) due to insufficient pointcloud density and coarse depth resolution causing bias in orientation estimates. Another limitation is the need for fixed thresholds on the head and shoulder ellipsoid parameters for classifying proposed pointclouds as human, which require manual tuning.

Future work will aim to empirically characterise errors in pose extraction and their correlation with observable factors such as range and observation angle, with a view to incorporate these relationships into the observation model used by the tracking algorithm. Additionally the aim will be to use machine learning based classification of ellipsoid parameters to replace manually set thresholds in task of classifying pointcloud proposals as human. Beyond this binary classification task we will also investigate the potential for the ellipsoid parameters to be used in individualising people and potentially performing re-identification across sensors. Additionally future work could seek to leverage data collected in real world scenarios and our robust tracking framework to learn more complex motion models able to overcome the limitations of a simple motion model in predicting complex social behaviours.

ACKNOWLEDGMENT

The material presented is derived from work under RMCRC Project 3.1.2 supported by the Rail Manufacturing CRC (RMCRC) and Downer EDI Rail Pty Ltd. The authors would also like to thank The Centre for Autonomous Systems, The UTS Transport Research Centre and Sydney Trains for their support.

REFERENCES

[1] A. Dib and F. Charpillet, "Pose estimation for a partially observable human body from RGB-D cameras," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, no. SEPTEMBER 2015. IEEE, sep 2015, pp. 4915–4922.

[2] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, and R. Moore, "Real-time Human Pose Recognition in Parts from Single Depth Images," *Commun. ACM*, vol. 56, no. 1, pp. 116–124, 2013.

[3] A. Baak, M. Muller, G. Bharaj, H. P. Seidel, and C. Theobalt, "A data-driven approach for real-time full body pose reconstruction from a depth camera," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1092–1099, 2011.

[4] C. Plagemann, V. Ganapathi, D. Koller, and S. Thrun, "Real-time Identification and Localization of Body parts from depth images," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3108–3113, 2010.

[5] N. Kirchner, A. Alempijevic, and A. Virgona, "Head-to-shoulder signature for person recognition," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2012, pp. 1226–1231.

[6] D. Brscic, T. Kanda, T. Ikeda, and T. Miyashita, "Person Tracking in Large Public Spaces Using 3-D Range Sensors," *IEEE Transactions on Human-Machine Systems*, vol. 43, no. 6, pp. 522–534, nov 2013.

[7] S. W. Joo and R. Chellappa, "A multiple-hypothesis approach for multiobject visual tracking," *IEEE Transactions on Image Processing*, vol. 16, no. 11, pp. 2849–2854, 2007.

[8] I. Ali and M. N. Dailey, "Multiple human tracking in high-density crowds," *Image and Vision Computing*, vol. 30, no. 12, pp. 966–977, 2012.

[9] M. Munaro, A. Basso, A. Fossati, L. Van Gool, and E. Menegatti, "3D reconstruction of freely moving persons for re-identification with a depth sensor," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 4512–4519, 2014.

[10] P. Morton, B. Douillard, and J. Underwood, "Multi-sensor identity tracking with event graphs," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, may 2013, pp. 4742–4748.

[11] N. Kirchner, A. Alempijevic, A. Virgona, X. Dai, P. G. Pl, and R. K. Venkat, "A robust people detection , tracking , and counting system," in *Australasian Conference on Robotics and Automation*, 2014, pp. 2–4.

[12] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York, USA: John Wiley & Sons, Inc., 2001.

[13] E. T. Hall, *The hidden dimension.* Doubleday & Co, 1966.

[14] C. C. Gordon, C. L. Blackwell, B. Bradtmiller, J. L. Parham, P. Barrientos, S. P. Paquette, B. D. Corner, J. M. Carson, J. C. Venezia, B. M. Rockwell, M. Mucher, and S. Kristensen, "2012 Anthropometric Survey Of U.S. Army Personnel: Methods And Summary Statistics," Tech. Rep., 2012.

[15] Qingde Li and J. Griffiths, "Least squares ellipsoid specific fitting," in *Geometric Modeling and Processing, 2004. Proceedings*, vol. 2004. IEEE, 2004, pp. 335–340.

[16] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," *Proceedings 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Cat No PR00149*, vol. 2, no. c, pp. 246–252, 1999.

[17] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[18] T. Linder, S. Breuers, B. Leibe, and K. O. Arras, "On multi-modal people tracking from mobile platforms in very crowded and dynamic environments," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, may 2016, pp. 5512–5519.

[19] K. Bernardin and R. Stiefelhagen, "Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics," *EURASIP Journal on Image and Video Processing*, vol. 2008, pp. 1–10, 2008.

[20] A. Teichman, S. Miller, and S. Thrun, "Unsupervised Intrinsic Calibration of Depth Sensors via SLAM," in *Robotics: Science and Systems IX*. Robotics: Science and Systems Foundation, jun 2013.

[21] M. Di Cicco, L. Iocchi, and G. Grisetti, "Non-Parametric Calibration for Depth Sensors," *Robotics and Autonomous Systems*, vol. 74, pp. 309–317, 2015.