2010 Seventh IEEE International Conference on Advanced Video and Signal Based Surveillance

Tracking People with a 360-Degree Lidar

John Shackleton and Brian VanVoorst
Raytheon – BBN Technologies
Saint Louis Park MN 55416
john.shackleton@bbn.com

Joel Hesch
University of Minnesota
Minneapolis, MN 55455
joel@umn.edu

Abstract

Advances in lidar technology, in particular 360-degree lidar sensors, create new opportunities to augment and improve traditional surveillance systems. This paper describes an initial challenge to use a single stationary 360-degree lidar sensor to detect and track people moving throughout a scene in real-time. The depicted approach focuses on overcoming three primary challenges inherent in any lidar tracker: classification and matching errors between multiple human targets, segmentation errors between humans and fixed objects in the scene, and segmentation errors between targets that are very close together.

1. Introduction

Until a few years ago, lidar devices were limited to either line scanning sensors that swept a beam across a scene, taking measurements along a single plane, or flash lidars that only face in a single direction. These sensors typical supported applications such as aerial remote sensing and precise 3D mapping of static scenes [1, 2]. While some systems use these devices to track motion, in practice they do not possess the field-of-view necessary to recognize moving objects in real-time without the aid of additional sensors [3, 4]. New lidar systems have emerged that utilize multiple coordinated emitter-detector pairs rotating 360-degrees multiple times a second. The result is a dense point cloud of precise 3D data representing a wider field-of-view of its surroundings, capable of monitoring an entire room without additional sensors or manual intervention. By processing a sequence of point clouds from successive rotations of the sensor, a computer is capable of capturing a dynamic scene of moving objects, thereby initiating an area of research that has not yet received much attention i.e., tracking objects inside dense 3D real-time point clouds.

For the work described in this paper, the lidar device employed was the Velodyne HDL-64 with a practical range of 120 meters. Operating at a 10Hz rotation rate, this device is capable of collecting 1.3 million 3D points a second, or approximately 130 thousand points each frame, depending on the content of the scene (see Figure 1). Such density is two orders of magnitude higher than traditional line scanning lidars and thus presents new technical obstacles and opportunities over previous lidar systems. Lidar has additional advantages as well over other sensor modalities. Unlike radar, lidar is very precise in its measurements. Unlike traditional video cameras, lidar is not affected by lighting conditions. Moreover, the range accuracy of the device is +/- 2 cm, which makes it an attractive technology for surveillance applications.

Tracking and activity recognition are not new topics, but performing these tasks effectively with such a large data stream in real-time is a new challenge. Our goal is to develop algorithms to process this large mass of data efficiently: segmenting the scene into targets of interest, classifying the segmented point cloud into those segments that are human, correlating tracks with established targets, and accurately tracking the targets from frame to frame. Of particular concern is target identification and correlation that minimizes false positives. Fortunately, we can draw from decades of prior work in the areas of traditional image processing, radar tracking, and static point cloud processing.

For our initial attempt at tracking in dense lidar point clouds, this paper concentrates on minimizing errors as much as possible. Real-time constraints have been relegated to a secondary priority, to be addressed once feasibility is demonstrated. Because of its relationship to practical security applications, our experiments focus on tracking people in an indoor space. While our approach has been developed for the detection of humans, it can be easily adapted to monitor vehicles and other non-human subjects as well. We have limited ourselves, for now, to a single stationary lidar.

2. Technical Approach

Tracking moving people with video cameras is a well-known problem that shares many of the characteristics of 3D lidar data processing. What differentiates dense lidar point clouds from camera images is the way in which the raw data is segmented and classified to find the objects of interest. Analogous to the pixels of 2D cameras, 3D lidar

978-0-7695-4264-5/10 $26.00 © 2010 IEEE
DOI 10.1109/AVSS.2010.52

IEEE computer society

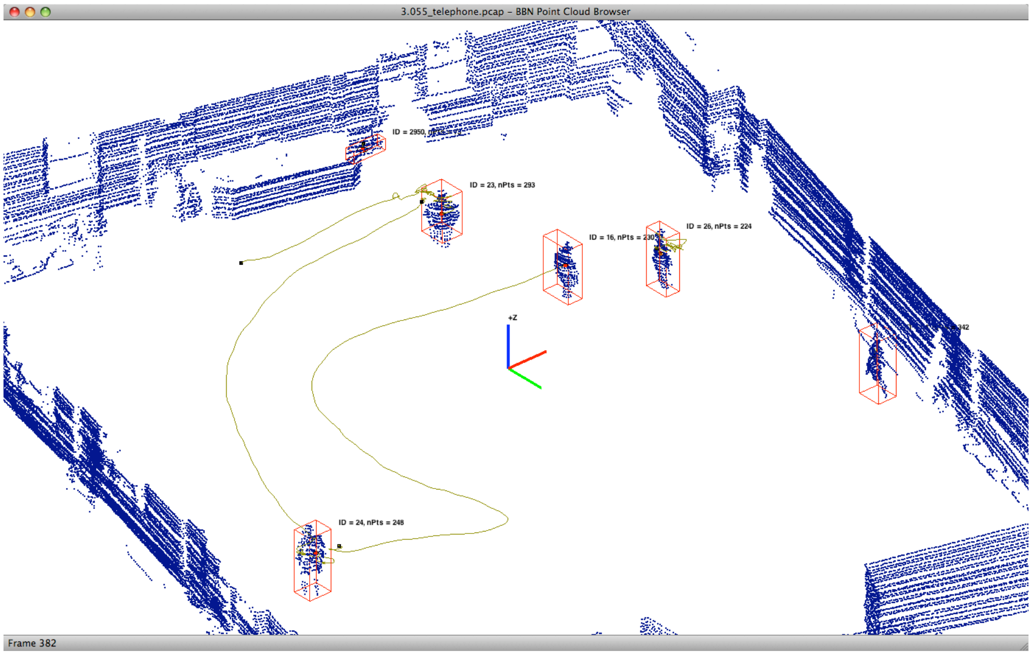

Figure 1: A sample frame of 360-degree lidar data is shown with human targets detected. Frame capture takes 1/10th of a second and contains over 120,000 3D points. The system must segment the scene into discrete objects, classify the objects, identify candidate targets, correlate candidate targets with prior targets, and track known targets.

segmentation and classification requires grouping and separating points (or voxels) of the point cloud into objects that are irrelevant, matched with existing targets, or identified as new targets. With traditional video, segmentation of pixels is accomplished via regions of color, textures, or illumination values based on edges that form boundaries. Segmentation in lidar data is based only on the physical 3D structure and separation of the objects. As with cameras, lidar segmentation must handle arbitrary occlusions and clusters of multiple targets in close proximity. Human targets that are near each other create an acute segmentation problem. A lidar alone cannot rely on texture or color information to disambiguate tracks.

In the process of developing our people tracking approach, three failure modes became the most apparent: classification and matching errors between multiple human targets, segmentation errors between humans and fixed objects in the scene, and segmentation errors between targets that are very close together. We will discuss each of these in greater detail. Figure 2 illustrates the basic architecture.

2.1. Classification and Matching Humans

Our segmentation and classification approach takes advantage of two characteristics of lidar data. First, the human target, or almost any specific class of target, has a unique signature formed by the curvature of its surface. Second, since a new point cloud is generated at 10Hz, target information accumulated from a previous frame can facilitate and validate target detection in subsequent frames.

2.1.1 Surface Matching

A wealth of surface matching techniques already exists and any number of them can be applied to real-time point cloud processing [5]. Our implementation settles upon a

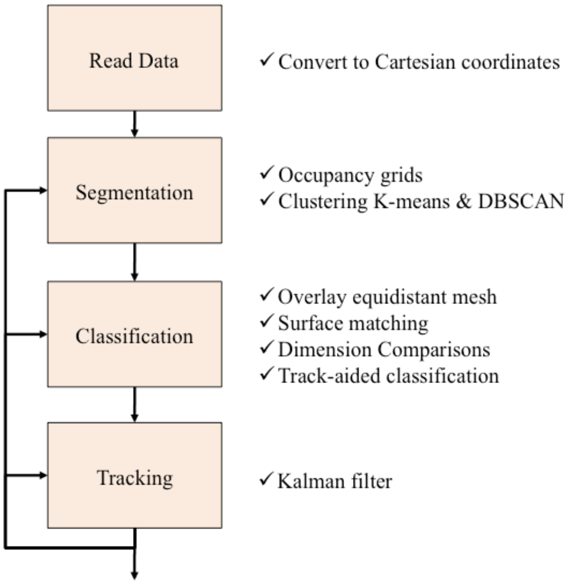
Read Data
✓Convert to Cartesian coordinates
Segmentation
✓Occupancy grids
✓Clustering K-means & DBSCAN
Classification
✓Overlay equidistant mesh
✓Surface matching
✓Dimension Comparisons
✓Track-aided classification
Tracking
✓Kalman filter

Figure 2: The basic architecture of the people tracker employs segmentation, classification, and tracking. The results of the tracking component are fed back into segmentation and classification.

surface matching technique called *spin images*, an approach that collects a set of contour descriptions local to each vertex on an object's surface [6]. Spin images are a good fit for 3D real-time point clouds because they handle occlusions very well, since they reduce the surface representation to a set of localized patches. Spin images are also relatively efficient, especially compared to other surface matching techniques developed for applications with static scenes that are not real-time constrained.

We utilize surface matching for lidar tracking as follows. A group of points is initially segmented based on spatial separation using 3D occupancy grids over the scene. Point groups are considered candidate human targets based on simple constraints of dimensions and orientation. For potential human targets, a spin image map is constructed. As consecutive frames are generated, spin image maps of new potential targets are compared to the spin image maps of previously identified targets, using the correlation formula described in [6]. When a new track has a correlation score that compares well enough with a prior target, then a match is declared. When a match is detected the spin image map of the target is updated with the latest surface descriptions of the track, and the target is tracked for another frame.

Tracks that do not match an existing target, and meet the dimensional constraints of a human, are identified as new targets. We could also use spin images to derive a set of templates representing various human shapes, which in turn can be used to classify new targets. For this initial work, however, we defer template-matching classification and use simple dimensional characteristics to identify new targets.

2.1.2 Tracking-Aided Classification

It would be inefficient to perform spin image comparisons against all existing targets in a scene, potentially hundreds of targets. Consequently, we employ an extended Kalman filter (EKF) to estimate the position of a target from one frame to the next [7]. Surface matching is then limited to the comparison of potential targets of the current frame against the spin image signatures near known tracks. The spin image correlation score for each comparison is thus refined to include the distance of the current track position from the estimated position of the established target, as shown in Eq. (1).

$$correlation = \sum_{n} C - e^{d*k} - 1 \qquad (1)$$

The value $\Sigma_n C$ is the accumulation of the initial scores resulting from n spin image map comparisons, each score is a normalized value between -1 and 1 with a "good" value above +0.5 (we are using n=20 spin image comparisons per matching attempt); d is the Mahalanobis distance between the current track and the known target calculated by the Kalman filter (through trial-and-error we have determined that a value less than 10 is required for matching tracks); and k is a small constant scaling factor on the order of 0.1. As a result, a potential match is scored lower if it is too far from the expected target position, creating much added robustness in the case where multiple, similarly shaped targets occupy the same area of a scene.

2.1.3 Uneven Point Density

Spin images were designed originally to recognize occluded static objects in controlled settings. A key assumption is that adjacent 3D points on an object's surface are approximately equidistant from each other. A 360-degree lidar device, however, cannot make this guarantee. The emitters on the Velodyne lidar are spread across a 26.8-degree vertical field-of-view, and fire every 0.09 degrees of rotation when spinning. The space between points grows farther apart as the target moves away from the sensor. Human subjects in our data sets typically consist of 100 to 1000 points, in contrast to high-resolution lidar applications that consist of tens of thousands of points per object [8]. To maintain proper robustness our approach cannot afford to eliminate points arbitrarily.

Instead, we compensate by interpolating a mesh of equidistant segments over the raw points of the potential target. The length of the segments is a function of the linear distance between the object and the lidar sensor. The technique we use to build the mesh is described in [9].

2.2. Segmentation and the Background Scene

A high number of tracking errors and false-positives with 360-degree lidars are related to the interaction of the people and the background scene. For example, people who stand next to the wall are often mistaken for the wall. Another key observation is that people cast *lidar shadows*, a transient, moving patch devoid of points in the scene created when a moving object passes in front of a fixed object (Figure 3). When working with a large number of

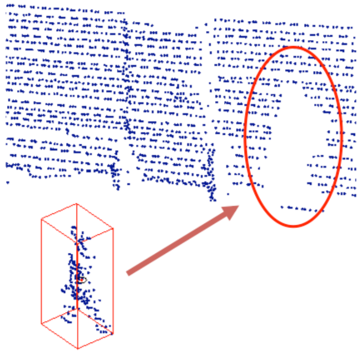

Figure 3: Lidar shadows create transient holes against static objects in the scene. Partially overlapping shadows moving in tandem create false positives during the people tracking process.

people in the room we found that moving shadows can intersect to create the illusion of moving patches on the wall that are misinterpreted as a moving object.

Both of these problems are addressed by eliminating from each frame the persistent static objects that are known to be non-human, such as the ground, walls, and furniture. In 2D image processing, this technique is known as background subtraction.

We perform 3D background subtraction by populating a separate occupancy grid with static objects. The cells of the occupancy grid form our background mask. Each 3D cell also includes a history buffer that records its recent occupancy, represented as a string of bits (eight bits in our implementation). Initially all cells are marked as empty and all bits are set to zero. Periodically we sample a frame of lidar data, and for each voxel in the frame we calculate which cell of the background mask it occupies. For each occupied cell, we set the highest order bit. Before sampling another frame, the background mask is aged, so that the lowest order bits of each cell's mask are shifted one place.

To determine which cells represent the static background, each cell in the scene's background mask is examined. A cell is marked as static background if a majority of its bits are set. A persistently empty space will have all bits in its cell mask unset. After sampling a minimum number of periodic frames, the learned background mask is applied continuously as a filter for future frames. A point in subsequent frames is discarded if it occupies a background cell. In practice, this approach reduces the lidar input data by as much as 90% (Figure 4).

This approach yields other benefits as well. First, it greatly simplifies segmentation, since most of points that are not marked as background are usually objects of interest. Segmenting people standing near walls is much simpler when the walls are removed. Second, transient artifacts such as lidar shadows are automatically removed without further processing. Following this procedure, as much as 85% of the false positives are removed before processing the frame.

2.3. Segmenting Close Groups of People

Some of the most challenging scenes for analyzing include human targets crowded together in a huddle, walking in a tight line, and standing shoulder to shoulder in a lineup. To segment these situations properly, special consideration must be given to human targets that are

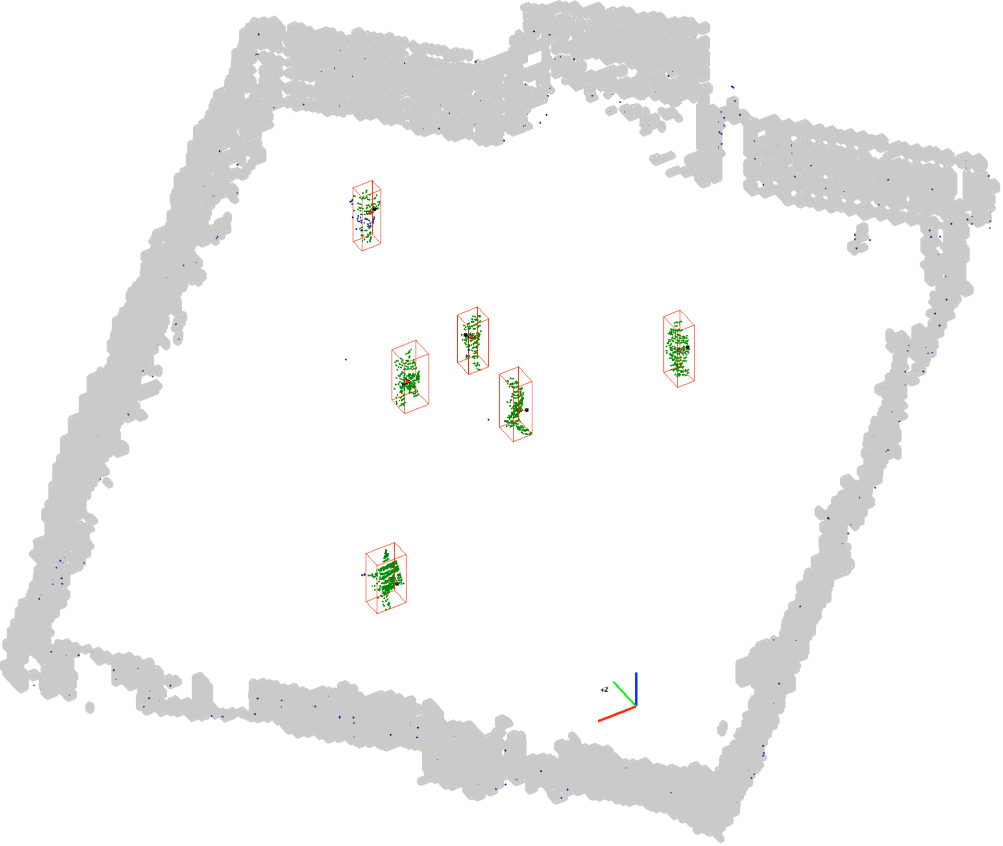

Figure 4: Background subtraction removes points that are masked with an occupancy grid. In the display above, the walls highlighted by grey boxes are filtered out via time-sensitive occupancy grids that are sampled over previous frames.

within a few centimeters of each other.

While surface matching can disambiguate patches of human targets, reliable surface matching comparisons within real-time lidar data cannot take place until after the human targets are first properly segmented into separate objects. For example, if multiple tracks in close proximity are incorrectly segmented into a single object, such as two people shaking hands (Figure 5), the resulting surface description does not contain sufficient detail to identify the separate tracks within the same. Moreover, sampling of correlation points of a multi-track object will likely include an unordered mixture of points from each of the tracks, and therefore will not capture a surface description that separates the multiple tracks properly. Therefore, false negatives during tracking will occur.

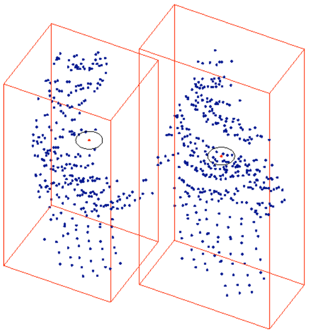

Figure 5: Two human targets shaking hands creates a challenge for segmentation. K-means and DBSCAN clustering techniques perform fine-grained segmentation when multiple tracks are close together.

Thus, a segmented object should contain at most one track. Three kinds of events can lead to an object with more than one track: 1) separate established targets can come together in close proximity, 2) new tracks can enter a scene together in close proximity, and 3) a hybrid event, such as a new track that was previously occluded first appears within a close group of existing tracks. Clustering is a common technique to segment 3D points in space with many algorithms and variations to choose from [10]. Our solution applies two common clustering algorithms, K-means clustering [11], [12] and DBSCAN clustering [13] to objects of a certain minimum size that may contain more than one track. For both clustering algorithms, the 3D points of the larger object are clustered in the x-y plane, i.e., the top-down view.

For the first case, K-means is used when known targets come together into a single larger object. The algorithm is seeded with the number of expected clusters (tracks) and an initial guess for the centroid point of each cluster, which are the estimated x-y midpoints generated by the Kalman filter. The K-means clustering iterates recursively over the points in the object until convergence is reached and all the points are assigned to a new cluster subdivided from the original object. The new (smaller) objects are then treated as any other potential track in the system, ready for classification. Clusters that do not have enough points for surface matching are discarded.

The other two scenarios are less predictable, because we have less confidence as to the number of tracks bundled into the larger object. Thus, we switch to DBSCAN clustering when the number of human tracks within an object is unknown. We tried an approach that exclusively used DBSCAN, but it was not reliable enough. For example, if two people are shoulder to shoulder, the DBSCAN approach often clusters the two tracks into a single object, while K-means is able to identify the separate tracks when their existence is established in prior frames. Certainly many other clustering algorithms are suitable for segmenting the point cloud, each perhaps optimal in specific situations.

2.4. Kalman Filter Details

As stated earlier, an extended Kalman filter estimates the frame-by-frame position of each person in the sensed environment, and we offer additional details here. The state of the ith person is

$$\mathbf{x}_i = \begin{bmatrix} \mathbf{p} & \dot{\mathbf{p}} & \theta \end{bmatrix}^T \qquad (2)$$

where $\mathbf{p}$ denotes the 3D position of the person with respect to the tracking frame of reference, $\dot{\mathbf{p}}$ denotes the 3D velocity of the person, and θ denotes the 1D heading direction of the person. The state propagation equation is based on the constant-velocity tracking model [7], which captures two characteristics of typical human motion: a person follows non-holonomic trajectories, and a person does not accelerate or decelerate quickly.

After each lidar rotation, the human target measurements that are matched to existing tracks are processed by the EKF. The measurement function is

$$\mathbf{z} = \mathbf{p} + \boldsymbol{\eta} \qquad (3)$$

where $\mathbf{p}$ is the measured 3D position of the person and $\boldsymbol{\eta}$ is the 3 x 1 measurement noise, distributed as zero-mean white Gaussian with covariance $\mathbf{R}$. We employ the standard EKF equations for updating the state.

Whenever a new person is observed, it is assigned a new EKF track. The uncertainty in the initial position estimate for the person only depends on the measurement covariance. The initial velocity of the person is set to zero, since it cannot be estimated from a single measurement. However, the covariance of the velocity is initially large to account for the uncertainty in the first estimate. Likewise, the initial heading direction of the person is set to zero, with a large uncertainty.

3. Experimental Results

To test the effectiveness of our overall approach, we collected over 60 different trials of people moving about in a controlled setting of a spacious indoor room. The trials ranged from the most basic situations, such as a person walking without obstruction across the room, to the more complex, such as six people performing a variety of activities (walking, running, sitting, standing) in close proximity with various obstacles in the room. The details of our experiments are as follows, highlighting each of our three main capabilities enabled and disabled in isolation: surface matching, background subtraction, and clustering.

3.1. Surface Matching Evaluation

At the outset of this endeavor, our assumption was that surface matching would be required to maintain tracking in the more challenging conditions, namely when tracks run the risk of getting confused with one another. This occurs most often when one track partially occludes another track or non-human objects occlude moving tracks. In our implementation, as it is currently configured, the impact of surface matching is minimal, and only affects the most pathological cases. Our experimentation shows that point clustering segmentation in combination with background subtraction and a well-tuned Kalman filter has been more crucial to improved performance using the data sets we have collected.

Table 1 illustrates this point. Three trials are identified. In the first trial, one person walks behind three partitions of varying degrees of occlusion. In the second trial, two people meet behind the same partitions before returning to their starting positions. In the third trial, two people walk back and forth around the obstacles, intentionally trying to hide and confuse the tracker.

Table 1: *Comparing False Negatives with and without Surface Matching.*

Trial	Frames	Frames with False Negatives WITHOUT Surface Matching	Frames with False Negatives WITH Surface Matching
1 person	185	5	5
2 people	186	10	8
Hiding	696	30	15

As the data shows, only in the most challenging trial did surface matching significantly reduce the number of frames with false negatives. Moreover, at no point in these trials did the Kalman filter fail to track the targets as they passed behind the static obstacles. This is largely due to robust segmentation and the ability of the 360-degree lidar to uncover reliable spatial relationships between objects in a scene.

The results do reveal that surface matching may have an important role in larger scale crowd scenes with unconstrained settings.

3.2. Background Subtraction Evaluation

A key benefit of background subtraction is the removal of false positive tracks. Table 2 describes the results of two sample trials. In the first sample, six people walk about randomly in the open space without coming into close contact. In the second trial, six people walk about in open space, coming into closer contact with each other and the walls. We counted the number of false positives targets in the frames after the background subtraction had been established (the first 89 frames).

As the results show, background subtraction eliminated the presence of all false positives in the scene due to lidar shadows or the interaction of people and fixed objects.

Table 2: *Comparing False Positives with and without Background Subtraction.*

Trial	Frames	False Positives WITHOUT Background Subtraction	False Positives WITH Background Subtraction
A	561	37	0
B	697	39	0

3.3. Clustering Evaluation

To illustrate the benefits of a combined K-means and DBSCAN clustering approach, consider three of the trials in which multiple people come into close contact with each other. In the first trial, two people simply cross back and forth in open space without occlusions. In the second, five people begin at different positions along the walls of the open room, about 15 meters apart, and then come together in a close huddle in the middle of the room for about 15 seconds, before dispersing into different directions. For the third trial, five people begin in a tight line formation in the middle of the room, remain still for about three seconds, play follow-the-leader for about 20 seconds, winding around in the open space, and then disperse into different directions for about seven seconds.

Table 3 shows the results. With the clustering enabled, four to fives times fewer frames display false negatives. The benefits are more pronounced when you examine the details of the false negatives. For example, in the follow-the-leader line trial when clustering is enabled, 86% of the frames with false negatives involve a single track getting lost for just a few frames, caused by one of the people in line getting partially obscured by the others. The remaining 14% of the false negatives involve two tracks getting temporarily lost. In contrast, when clustering is disabled for the same dataset, only 11% of the false negative frames can be considered a momentary loss of one or two tracks, while 29% of the frames lose three of the tracks, and the remaining 59% of the false negative frames are unable to segment any of the tracks correctly.

Table 3: *Comparing False Negatives with and without Clustering.*

Trial	Frames	Frames with False Negatives WITHOUT Clustering	Frames with False Negatives WITH Clustering
Simple	626	20	5
Huddle	249	93	16
Line	295	228	50

4. Conclusions

The approach described herein demonstrates how a 360-degree lidar may be implemented to track people in real-time. A few final observations are worth noting. First, the integration of position tracking estimator (Kalman filter), point clustering (K-means and DBSCAN) and surface matching (spin images) proves to be a powerful combination that offers improved results for boundary conditions than either of the three techniques alone. Second, background subtraction is a straightforward approach to eliminate the majority of false positives in a typical scene, and greatly reduces the complexity required by segmentation and classification. Finally, although its benefits have not yet been fully realized, real-time surface matching is feasible and shows potential as a technique to classify and identify targets in more unconstrained environments.

Acknowledgements

We would like to give special thanks to Velodyne Lidar Inc. for the use of their sensor.

5. References

[1] Lefsky, M., Cohen, W., Parker, G., Harding, D. "Lidar Remote Sensing for Ecosystem Studies," *Bioscience*. Vol. 52, No. 1. pp. 19-30, January 2002.

[2] Topol, A., Jenkin, M., Gryz, J. et al. "Generating Semantic Information from 3D Scans of Crime Scenes". Computer and Robot Vision (CRV '08), May 2008.

[3] Albus, J., Hong, T., and Chang, T. "Segmentation and Classification of Human Forms using LADAR Data". 35th Applied Imagery and Pattern Recognition Workshop (AIPR'06), Washington DC, 2006.

[4] Wenzl, K., Ruser, H., and Kargel, C. "Configuration of a Sparse Network of LIDAR Sensors to Identify Security-Relevant Behavior or People". Proc. of SPIE Vol. 7480, 748007, 2009.

[5] Campbell, R. and Flynn, P. "A Survey of Free-Form Object Representation and Recognition Techniques". Computer Vision and Image Understanding 81: pp166-210, 2001.

[6] Johnson, A. "Spin-Images: A Representation for 3-D Surface Matching". PhD Thesis, Carnegie Mellon University, CMU-RI-TR-97-47, August 1997.

[7] Ramachandra, K.V. *Kalman Filtering Techniques for Radar Tracking*. CRC, 2000.

[8] The Stanford 3D Scanning Repository. http://graphics.stanford.edu/data/3Dscanrep/

[9] Hoppe, H. "Surface Reconstruction from Unorganized Points". PhD Thesis, Dept. of Computer Science and Engineering, University of Washington, June 1994.

[10] Xu, R., Wunsch, D. "Survey of Clustering Algorithms". IEEE Transactions on Neural Networks, Vol. 16, No. 3. pp 645-678. May 2005.

[11] Lloyd, S. P. "Least squares quantization in PCM". IEEE Transactions on Information Theory 28: pp 129–137, 1982.

[12] MacQueen, J.B. "Some Methods for Classification and Analysis of Multivariate Observations". *Proc. of 5th Berkeley Symposium on Mathematical Statistics and Probability*. Berkeley, University of California Press, Vol. 1, pp 281-297, 1967.

[13] Ester, M., Kriegel H., Sander, J., and Xu, X. "A density-based algorithm for discovering clusters in large spatial databases with noise". Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96). AAAI Press. pp. 226–231, 1996.