TrackSense: Infrastructure Free Precise Indoor Positioning using Projected Patterns

Moritz Köhler[1], Shwetak N. Patel[2], Jay W. Summet[2], Erich P. Stuntebeck[2], Gregory D. Abowd[2]

[1] Institute for Pervasive Computing, Department of Computer Science
ETH Zurich, 8092 Zurich, Switzerland
koehler@inf.ethz.ch
[2] College of Computing & GVU Center
Georgia Institute of Technology
801 Atlantic Drive, Atlanta GA 30332-0280 USA
{shwetak, summetj, eps, abowd}@cc.gatech.edu

Abstract. While commercial solutions for precise indoor positioning exist, they are costly and require installation of additional infrastructure, which limits opportunities for widespread adoption. Inspired by robotics techniques of Simultaneous Localization and Mapping (SLAM) and computer vision approaches using structured light patterns, we propose a self-contained solution to precise indoor positioning that requires no additional environmental infrastructure. Evaluation of our prototype, called TrackSense, indicates that such a system can deliver up to 4 cm accuracy with 3 cm precision in rooms up to five meters squared, as well as 2 degree accuracy and 1 degree precision on orientation. We explain the design and performance characteristics of our prototype and demonstrate a feasible miniaturization that supports applications that require a single device localizing itself in a space. We also discuss extensions to locate multiple devices and limitations of this approach.

1 Introduction and Motivation

We introduce a solution to indoor localization, TrackSense, that requires no additional infrastructure in the environment and provides 3D positioning and orientation data that performs well against existing research and commercial solutions. Although we have seen great progress toward the goal of indoor localization, almost all of the solutions that offer precise (few centimeter) indoor localization have been limited to techniques that require the introduction of new infrastructure to the physical space (*e.g.* cameras or beacons). These solutions are often costly and typically require time-consuming installations, and it is not easy to move the instrumentation from one space to another. Although existing commercial positioning systems are adequate for prototyping user experiences, their ultimate success relies on a localization approach that is inexpensive and easily deployed.

TrackSense is appropriate for situations where the localized device has a clear view of the walls and ceilings. By centralizing all computation to a single, small device, we reduce cost and substantially increase the number of places the localized device can be used.

In addition to the inherent technical challenges, there are several motivating applications in which a single computational device benefits from precise location. Patel *et al*. demonstrate a see-through augmented reality, handheld device capable of performing precise at-a-distance interaction [20]. Their iCam device provided simple authoring and retrieval of digital content attached to physical objects, as well as manipulation of digital content in an augmented reality game. The iCam relied on a commercial ultra-wideband positioning system for localizing the handheld. Cao and Balakrishnan demonstrated the use of a handheld projector for viewing and interacting with multiple dynamically defined information spaces projected in the physical space [5]. Their application used a commercial camera-based motion capture system to determine the pose and position of the projector. In addition to these research prototypes, many examples of augmented reality rely on precise tracking of an object (such as an individual's head) and these applications would be improved by any solution that would speed deployment in multiple spaces.

TrackSense determines its distance and orientation to fixed large planes in a space (*i.e.*, walls and ceilings) and uses that information to calculate its 3D position and pose in the room. Inspired by robotics localization and camera-projector research, our solution uses a camera to locate and track a grid pattern projected onto surfaces in the camera's field of view. This solution is more accurate and reliable than standard computer vision feature extraction techniques, because the exact feature (the grid pattern) is known and ever-present in the camera's view. It also provides a useful complement to traditional stereo vision, which does not perform well on plain surfaces. In addition, our technique provides information regarding its pose that is not available with standard ultrasonic or laser range finding solutions. Combining our solution with less precise room-level positioning systems we can provide localization within an entire world coordinate frame. Our current prototype is bulky and only demonstrates localization of a subsection of any given room. However, we also describe a miniaturized system that can be extended to an entire room.

2 Related Work

Indoor location technologies have been a long-studied topic in pervasive and ubiquitous computing. Hightower and Borriello provide an overview of the various location technologies and techniques [14]. The two basic approaches are to build the entire infrastructure from the ground up (*e.g.*, Ultra-wideband [35], ActiveBadge [37], Cricket [25], Vicon [36], NorthStar [21] and Active Bat[1]) or to leverage existing infrastructure that can yield localization, either through triangulation or fingerprinting (*e.g.*, 802.11 work such as RADAR [2] and Place Lab[15], GSM Cell Towers [22], Bluetooth [19], and powerlines [24]). Typically, solutions that offer precise indoor localization of a few centimeters use the first approach of installing new environmental infrastructure that is both expensive and hard to move, thus limiting location-based applications to a few highly specialized environments. Although researchers are exploring ways to leverage existing public infrastructure, the solutions are currently limited to resolutions of a few meters (room-level).

The robotics community has a long history of exploring ways to localize autonomous robots without having to install custom infrastructure or gather *a priori* topological knowledge of the environment. Researchers have extensively studied the use of highly precise laser or ultrasonic range finders to automatically construct a feature map of the environment and then later consult it for localization [9, 18, 34]. This class of techniques is called Simultaneous Localization and Mapping (SLAM). A visual variant of SLAM, visual SLAM (vSLAM), builds a map entirely using vision [7, 30]. SLAM solutions typically employ various statistical and probabilistic models for localization. In addition, SLAM is a recursive process that evolves over time to improve accuracy and address changes in the environment. Although inspired by robotics, our solution does not rely on a statistical model or the construction of a complete map of the environment.

Vision-based techniques extract features from the physical environment, such as detecting planar surfaces for 3D model extraction [3, 6, 8, 17, 27, 33]. One limitation of purely vision-based techniques is the requirement of easily discernible and static features in the environment. Features many not always be available, such as on single-colored or plain walls. Additionally, lighting conditions may change the way features appear at different times. In our solution, the features (projected grid) are placed artificially in the environment to ease feature extraction. Our solution works best on the plain surfaces on which other computer vision approaches such as stereo vision fail. However, stereo vision techniques would be complimentary when the device is used in more textured spaces.

Our approach with TrackSense is similar to previous vision work using structured light to extract physical feature information from an object, which use projected coded patterns of light at an object to extract the 3D features of that object [3, 29]. Other research has used the detection of structured light on a planar surface to automate projector calibration [31, 32]. These solutions temporally encode different structured light patterns; we focus on a static pattern produced by a laser to ensure a small, low-cost solid-state solution.

Finally, augmented reality researchers have explored using fiducials (such as barcodes or 2-dimensional glyphs) to determine distance and pose to labeled objects and surfaces [12, 28]. However, large glyphs are needed for long distances and a number of them must be placed in the environment to cover a large space. In addition, glyphs are not always aesthetically pleasing unless they are blended with the décor of the environment. Our approach can be made invisible to the user by using infrared lasers.

3 System and Implementation Details

TrackSense projects a grid pattern into the environment to locate planes (walls) and intersections (corners). By detecting three orthogonal planes (two walls and a ceiling or floor), the system can recover its position and orientation with respect to that corner. By using a

3-axis accelerometer and magnetometer (compass), the unit can determine which corner of a room it is looking at, and hence, its position and orientation with respect to the room's coordinate frame. Practically, several TrackSense units (2-5) angled in different directions can cooperatively identify three planes within their combined views. This section discusses the implementation of a single TrackSense unit and how it obtains distance and orientation measurements from one or more planes it observes in the environment.

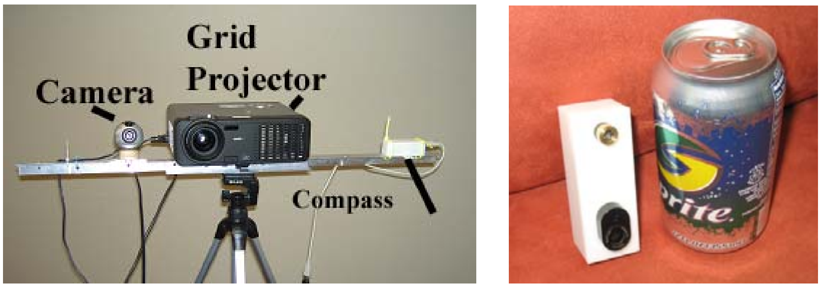

Figure 1: Left: Operating TrackSense prototype and its components. Right: A miniaturized design prototype with laser diode, camera, and 400 MHz GumStix computer.

3.1 Hardware

TrackSense has a grid projector and a camera (see Figure 1). We used a 2000 Lumen DLP projector, which simulates a grid projecting laser diode. We used a projector to allow for easy prototyping of different sizes and shapes of projected grids. In an actual engineered solution, the relatively large desktop projector would be replaced with a single laser diode and grid diffraction lens, possibly projecting infrared light to make the system's operation imperceptible. For our prototype camera, we used a Logitech QuickCam Pro 4000 USB webcam with VGA (640x480) resolution. Our prototype system also had a custom-built magnetometer for a cost of $50 USD with a resolution of about 2°. In an actual system, several TrackSense units would share a 3-axis accelerometer and magnetometer. For prototyping, these components were connected to a desktop PC running our software, which was written in C++ using Intel's OpenCV and the VXL computer vision libraries.

Projected Grid & Camera Calibration.
As with all stereo vision devices, for a TrackSense unit to operate correctly, the grid projector and camera must be calibrated, both with respect to each other and with respect to a ground truth or world coordinate system. By using point correspondences between the grid projector and the camera using a known calibration rig, we can find the Fundamental matrix, F that encodes the relationship between the grid projector and the camera. The

Fundamental matrix is defined by the equation $x'^T Fx = 0$ for any pair of matching points $x \leftrightarrow x'$ in two images. In other words, if two points x and x' correspond, the equation described above evaluates to 0. Note that mathematically, the grid projector is assumed to be a second camera, with grid line intersections at specific points in the virtual grid "image." We use F to help determine point correspondences as described in Section 3.2.3. By using known world coordinate points on our calibration rig, we can also calculate the projective matrices, P and P`, between the world coordinate system and the camera and virtual grid "camera" that is used to determine the location of detected points with respect to the TrackSense unit. We computed this standard multiple-view calibration with a custom rig using the calibration routines from OpenCV [13]. In actual operation, this calibration would be performed a single time at the time of manufacture.

3.2 System Operation

As the grid is projected onto and reflected from objects in the environment such as the ceiling and walls, the camera detects the lines using a custom edge detection algorithm, Using these detected lines, we can find the location of each grid intersection point. Because we are integrating data from hundreds of pixels for each line, we can develop a mathematical model of the line that is more accurate than any single pixel. Hence, we can measure locations of intersection points with sub-pixel accuracy. By triangulation (using the same math as standard stereo vision), we can find the distance and orientation to each point relative to the camera. Using multiple points, we can detect planes and corners where multiple planes meet. From this, we can recover the orientation and position of the TrackSense unit with respect to the corner. If we use a 3-axis accelerometer and magnetometer to determine which corner we are observing, we can locate the TrackSense unit within the room. Furthermore, if we already can identify the room using a less accurate positioning system, such as GSM fingerprinting [22], the within room position translates to an accurate world position. In this section, we describe elements of this procedure in more detail.

3.2.1 Line Detection

To determine where the grid intersection points occur in the camera image, our system must first detect the projected lines from a potentially noisy image. Using a standard Canny edge detection algorithm would detect the grid, but it would also detect many other lines in the image (*i.e.,* edges of windows, picture frames, desks, pencils, *etc*). One way to enhance the detection of the projected grid would be to take pairs of images, one with the grid turned off, and one with it turned on, and then subtract them to obtain the location of only the grid. However, this reduces the frame rate of the system by one-half, requires precise synchronization between the grid projector and camera, and assumes that the system is not in motion between subsequent frames. Custom hardware operating at extremely high frame rates where these assumptions may be valid could make use of this subtraction technique to greatly simplify the line detection algorithm.

However, our prototype uses a web cam with limited frame rate and no synchronization to the grid projector. We also wanted our system to operate while in motion and be able to provide a new orientation and position with every camera image. To enable this, we developed an enhanced edge-finding algorithm that detects projected grid lines while ignoring many environmental lines. Figure 2 (center) shows results obtained using a standard implementation of the Canny edge detection algorithm [4]. The Canny algorithm looks for gradients in the image, detecting lines for both low-to-high and high-to-low transitions. Naturally occurring lines in the environment (*i.e.*, from a corner or edge) typically only have one of these gradients, either increasing or decreasing, as the edge typically separates objects of different reflectance levels. However, projected lines are typically brighter than the objects they fall upon, leading to both a rising and falling gradient on either side of the line. As shown in Figure 2:

1. For our single luminous projected line, two edges are detected: One for the increasing gradient and one for the decreasing gradient.
2. Each edge of the black square (upper left) results in exactly one detected edge.

We can obtain better results by modifying a gradient-based edge detection algorithm so that a positive gradient followed by a negative gradient of similar magnitude is used to detect a line. This leads to a zero-crossing edge detection algorithm that uses the 1st derivative instead of the 2nd derivative. Figure 2 shows the result of such an algorithm. This helps reduce both the ambiguity problem and the false positives.

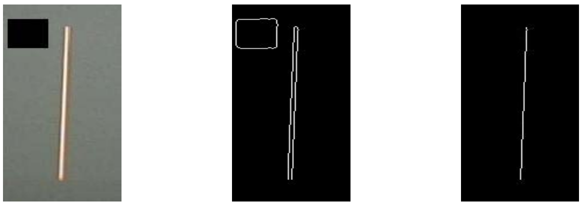

Figure 2: Left: A line projected onto a wall and a black square representing an object in the environment. Middle: Results obtained applying Canny edge detection to the image on the left. Right: Result obtained applying the gradient based edge detection algorithm we developed.

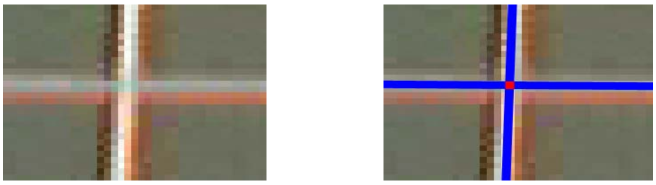

Figure 3: Left: Zoomed in image of two intersecting lines. Right: The same lines from the left superimposed with lines detected by our line finding algorithm.

3.2.2 Intersection Points

Using the edges detected from the projected grid, a Hough transform can determine the parameters of each line [11]. By mathematically determining the intersection point of each pair of lines, we obtain the position of these points with sub-pixel accuracy. Figure 3 shows an intersection of two projected lines in our camera view, and an overlay of the detected lines (blue) and intersection point (center red dot). Our prototype used a grid of 9 vertical lines and 7 horizontal lines, which corresponded to our 4:3 aspect ratio camera, giving a maximum of 63 detectable points.

3.2.3 Point correspondences and 3D reconstruction

An important step of reconstruction is the correct identification of point correspondences between two views. To determine the orientation and distance to any point in the environment, a stereo rig must identify where that point appears in both camera views. Traditional stereo vision algorithms [11] rely on distinctive textures in the pair of images to determine which points from the left camera image corresponds to a particular point in the right camera image. However, we are unable to use a similar method for two reasons. First, our system needs to be able to work on plain walls without features, which lack the texture that traditional stereo vision algorithms rely upon. Second, we are using a projector as a virtual “camera”. The advantage of the projector is that our system will work on walls without texture by projecting its own features, but the disadvantage is that the grid is regular and each intersection point looks very much like all the others.[1]

Given a grid intersection point in the projector “view”, the correct corresponding point in the camera view must to be found. In order to reduce the search space for this point the epipolar constraint from the Fundamental matrix is applied [13]. Figure 4 shows epipolar lines for our prototype in which the camera and grid projector were mounted horizontally.

To determine point correspondences we use a cost function that is the sum of the squared distance to the epipolar line and the position of that point in the previous image (timestep t-1). Using these cost functions, the Hungarian algorithm [16, 20] minimizes the total cost and produces the best match in correspondences between the grid intersections and the detected line intersections in the camera image. In some cases, intersection points may be missing from the camera image. For example, lying on a dark or textured object or falling outside the view of the camera would prevent detection by the edge and line finding algorithms. To prevent errors in these cases, the cost for questionable points are set to infinite cost (allowing the Hungarian algorithm to skip that point) if the distance to the epipolar line was greater than a pre-set threshold.

Once point correspondences are known, the projective matrices, P and P` obtained in the initial system calibration are used to calculate the 3D position of the point. Linear

[1] As our prototype uses a data-projector to simulate a laser-grid, we could have used textured patterns (as used in 3D reconstruction using structured light [3, 29]). To work with very small and inexpensive laser diodes, we limited the output of the projector to a uniform static image. With custom diffraction lenses on a laser diode it would be possible to produce a laser pattern that, while static, would not be regular, allowing for easier calculation of point correspondences.

triangulation is used to obtain the desired 3D position of a point. More details of the linear triangulation approach, along with methods for improving its accuracy can be found in [13].

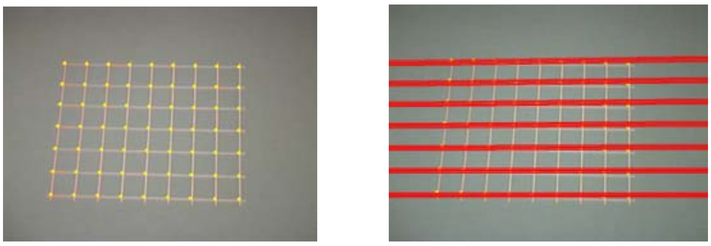

Figure 4: Left: A projected grid. The yellow dots show points of intersecting lines. Right: Left image superimposed with epipolar lines.

3.2.4 Identifying Planes

A TrackSense unit models walls, ceilings or floors as large planes. After triangulation, we have data points that may represent points on the surface of planes, or may be noise, either from random objects in the environment or measurement errors. In order to develop a robust algorithm that can detect planes correctly, several issues arise.

- The exact number of planes in each frame is unknown. Because each TrackSense unit has a finite field of view and operating range, we do not expect to detect more than a maximum of 3 planes of usable size (a corner of two walls and a ceiling or floor).
- It is not known which points lie on the surface of the same plane and form a group.
- A significant amount of points represent noise and have to be classified as outliers so they do not affect the correct computation of a plane.

Our approach uses the RANSAC (RANdom Sample Consensus) algorithm [10]. First, three points which specify a plane are randomly selected, and every remaining point is tested to see if it is close to the candidate plane. We have found that a threshold of 3 cm includes most valid points while eliminating most outliers. After selecting many possible random planes, the one with the largest group of supporting points is chosen. The valid points are then used to compute a least mean square solution for the actual position and orientation of the plane, resulting in better accuracy than any of our single point measurements. Planes without enough supporting points are discarded. The algorithm terminates with failure to detect a plane. Otherwise the previous steps are repeated in order to find the next plane. We have found that with a 7 by 9 grid (63 total points) a threshold of 18 points generally indicates that a valid wall, floor, or ceiling plane has been found. Figure 5 shows a point cloud representing points on the surface of two walls of a corner (left), and the planes that have been fitted to the points using the approach described above (right). Once the four parameters characterizing each plane have been determined, the distance from the TrackSense unit to the plane can be determined geometrically.

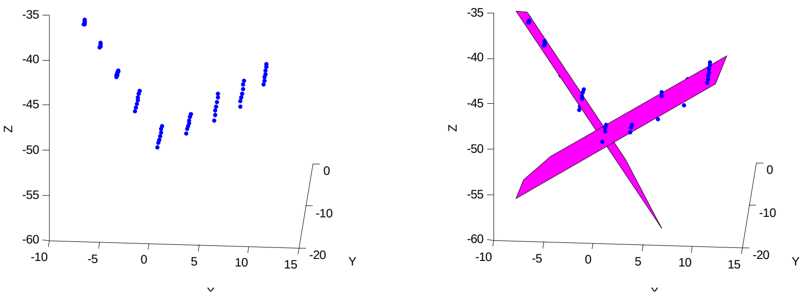

Figure 5: Left: Point cloud from a two wall corner. Right: Point cloud plus the fitted planes.

3.2.5 Determining Position and Orientation

Depending on the number of walls, we discuss the different strategies for determining the position and orientation of the device.

One Wall: A TrackSense unit that can observe a wall can determine its orientation with respect to that wall, and distance from the wall. With a single wall in view it can act as an enhanced ultrasonic tape measure, being able to calculate the direct distance from the unit to the closest point on the wall. Unlike an ultrasonic tape measure, the TrackSense unit does *not* have to be pointed directly at the closest point on the wall to make this measurement, as it also knows the orientation of the wall.

Two Walls: By observing two orthogonal walls, a TrackSense unit is able to determine its (X,Y) position with respect to the corner. If it makes the assumption that the two walls are vertical (at 90 degrees to the ground plane) the TrackSense unit can determine its own orientation with respect to the ground plane. Note that the two walls do NOT have to form a 90 degree angle with each other, only with the ground plane.

By making use of a magnetometer a TrackSense unit can determine its global bearing, and determine which corner of the room it is observing, which leads to a global (X,Y) position within the room. Also note that if the room has a different angle at each corner (as opposed to the standard 90 degree corner) the TrackSense unit can measure the angle between each set of two walls and use that to “fingerprint” the corner that it is looking at. Although two walls do not provide a Z (or height) measurement, if the unit is held at a consistent height, or mounted on a mobile base the Z component may be stable. While an ultrasonic transducer could be used pointing towards the ground or ceiling to provide an estimate of Z, we recommend the use of a second TrackSense unit for redundancy.

Three Walls / Corner: By observing three orthogonal planes (such as the intersection of two walls and the ceiling or floor) a TrackSense unit can determine its full 6 degree of freedom position and orientation with respect to the corner. If it can identify the corner, it can also obtain its global position and orientation within the room. Note that our prototype only has enough resolution and field of view to accurately detect two planes simultaneously. Hence, the analysis in Section 4.2.2 assumes a constant Z value (the unit sat on a wheeled platform). We expect in actual operation, 2-5 TrackSense units would operate on the same rigid body. Even if three TrackSense units could each only detect a single unique plane, the combination of distance and orientation data from each TrackSense unit would be equivalent to a single "super TrackSense" observing the three planes directly.

3.2.6 Adding a Magnetometer & Accelerometers

The previous analysis assumes that the TrackSense unit is somewhat vertical. If it is held upside down, its yaw, pitch, and roll measurements will be incorrect by 180 degrees, and if held sideways it will mistake the floor and ceiling for walls and visa versa. If the TrackSense unit will be held generally level (within 30 degrees), the use of accelerometer data is not strictly necessary for satisfactory operation. However, most solid state magnetometers integrate a 3 axis accelerometer (and 2 orthogonal magnetometers) to ensure that the compass bearing is accurate even if the unit is tilted. By using data from a 3 axis accelerometers we can enhance the TrackSense unit in two ways:

- By detecting the 1G acceleration of gravity and magnetic north the unit can operate in any orientation and provide correct yaw, pitch, and roll data (Excepting zero-gravity environments).
- The data from inexpensive accelerometers (with fast update rates, but moderate drift) can be used to provide updated position and orientation data between camera frames or while the cameras do not have a view of enough planes to obtain full 6 DOF data.

4 Performance Evaluation

We conducted four experiments to determine the accuracy and precision of the system's position and orientation measurements. The first two involved measuring the distance and orientation with respect to a single wall or plane. The third measured the ability of a single TrackSense unit to measure the angle between two walls or planes. The fourth experiment used the distance and orientation to the intersection of two walls to measure the (2D) location of the TrackSense unit in a room. These tests allow us to report on the overall positioning and orientation accuracy of our prototype and predict the accuracy and precision of a system using multiple TrackSense units to position and orient itself within a room. For accuracy, we report the difference between the system's determination of its location and the measured ground truth. For precision, we show the smallest discernable position unit by observing the variations of the system's reported position at specific locations.

4.1 Distance and Orientation to a Single Wall

Figure 6 (left) shows the single wall experimental setup. At each test point 300 consecutive samples were taken. In the two experiments, we varied the prototype's distance and angle to the wall while keeping lighting constant at normal office illumination levels.

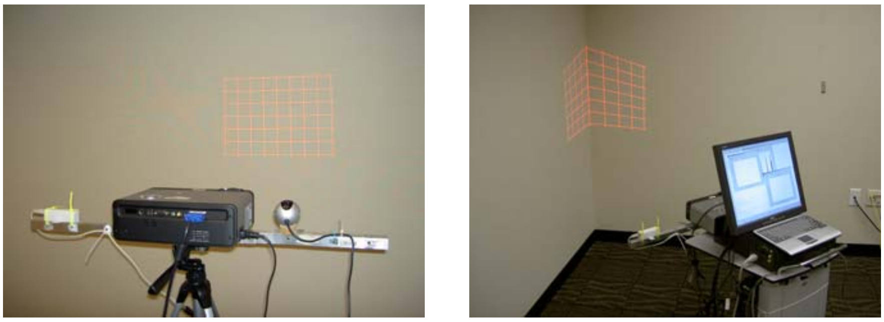

Figure 6: Left: Experiment set-up for one wall experiment with projected grid. Right: Two wall experiment setup with projected grid. In both experiments, the apparatus was at a fixed height.

4.1.1 Distance

In this experiment, the apparatus was pointed straight towards one wall and the perpendicular distance was measured as the ground truth. We took measurements at nine points ranging from 75cm to 325cm from the wall. The TrackSense prototype has a minimum range slightly under 75cm (due to geometric constraints), and the optimal working range extends to 275cm, although less accurate results can be obtained up to 350cm.

The accuracy of the system is shown in Figure 7. The straight blue line represents the result of a least mean square linear regression for the sample points. The distance between the measured and the actual data closely follows a linear function (*y=ax+b)*. This systematic error comes from the fact the lines from the projector increase in thickness as the apparatus is moved farther back, thus shifting the detected lines farther to the left. Using a true laser grid would mitigate this problem substantially, although there would still be a slight systematic error. However, because the system error is linear, a correction factor can be applied at the factory to improve the overall accuracy. This correction factor is a simple offset value learned through experimentation. When we applied the linear correction factor, our corrected accuracy was between 3-4 cm.

Within the working range, the precision is on average 2 cm (see Figure 7). The results indicate that the standard deviation increases quadratically with increasing distance from the wall. As the area imaged by our camera increases with the square of the distance, this curve corresponds to the expected reduction in sensor resolution with respect to wall area.

Precision decreases drastically near 3 meters. Several factors cause the system performance to begin degrading at this point. As the distance from the TrackSense unit to the wall is increased, the intensity and size of the projected lines in the camera image is reduced until the edge detection algorithm can no longer successfully identify all lines. With fewer lines, fewer points are detected, and more incorrect point correspondences are made, leading to more outliers for the RANSAC algorithm. Increasing the resolution of the camera and using a brighter laser grid projector would increase the effective working range.

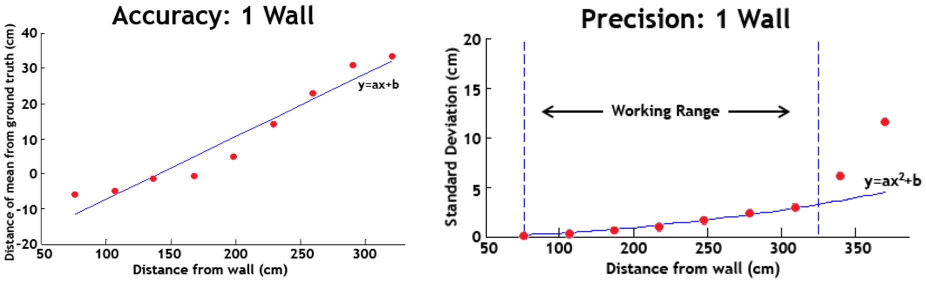

Figure 7: Accuracy (left) and precision (right) of distance facing a single wall.

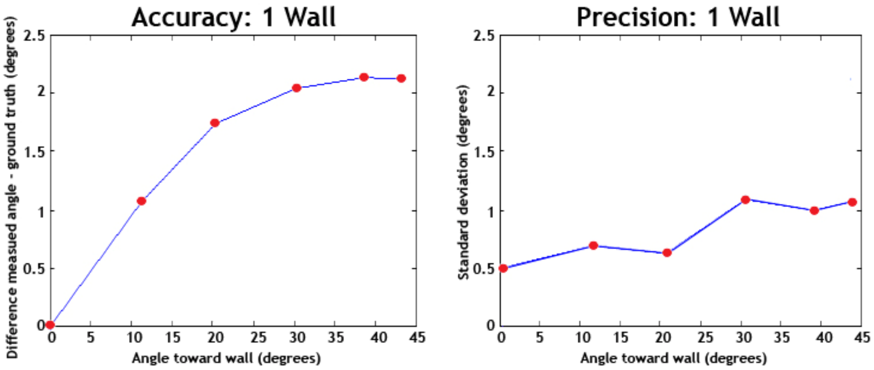

Figure 8: Accuracy (left) and precision (right) for angle measurement towards one wall.

4.1.2 Orientation

We also characterized the system's ability to determine its orientation with respect to a single wall or plane. We measured the angle between the prototype and a wall at six different angles: 0,10,20,30,40 & 45 degrees. At each position, 300 samples were collected to calculate the device's precision. The TrackSense prototype was located 120 cm from the wall, and swiveled from 0 degrees (directly facing the wall) to 40 degrees in 10 degree increments, and a final measurement was taken at 45 degrees. Beyond the 45 degree angle we expect the TrackSense unit would have a less acute angle to an adjacent wall.

Over the tested 0 – 45° range, the TrackSense prototype has an accuracy of 2° or better and has a precision of 1° or better (see Figure 8). We attribute the slight decrease in precision and accuracy as the angle increases to the smaller surface area that is visible to the camera as the incident angle is increased. We did not measure angles beyond 45 degrees because in standard operation we expect the TrackSense unit to have a view to an adjacent wall with a less-acute angle.

4.2 Two Wall Experiments

We investigated using our prototype to measure the angle between two walls, as well as using the distance and orientation from a known corner to measure the (X,Y) location of our prototype within a room (see Figure 6). All measurements were taken with the grid centered horizontally in the corner and the TrackSense unit directly facing the corner.

4.2.1 Angle between two walls

The ability to calculate the angle between intersecting planes is important in recognizing unique corners (*e.g.*, an odd shaped room where each corner has a different angle). For the two wall angle experiment, a movable surface against the corner of a room approximated a second wall, and measurements were taken at three different angles at 90°, 67.5° and 45° as measured with a protractor. The accuracy of angle measurements is shown numerically in Table 1. TrackSense provides accuracy of better than 2°. The accuracy degrades as the angle gets narrower. As the angle between the walls decrease, the angle from each wall to the TrackSense unit increases causing an increase in error similar to that seen in Section 4.1.2. The precision of the system remains relatively constant (around 1°) despite the angle of the walls.

Table 1: Accuracy of angle measurements between two walls

Ground truth	Measured angle (Mean)	Difference (Error)	Standard Deviation
90.00°	90.08°	0.08°	1.70
67.50°	69.09°	1.59°	1.83
45.00°	43.06°	1.94°	1.25

4.2.2 Location Within a Room

By observing a corner, our prototype measures the distance to two walls and can produce an (X,Y) location within a room. Using a fixed height, and keeping the TrackSense grid projector and camera pointing towards a known corner, measurements were taken from a total of 25 positions equally spaced on a grid covering an area of approximately 2.0m x 3.0m. We took 300 samples at each location with standard office lighting conditions.

Figure 9 shows the raw data results of the two wall position experiment. Throughout the experiment, we pointed the apparatus toward the lower left corner of the room (the corner at the origin of the coordinate frame). This raw data has an average accuracy of only 29.0cm. When we apply the linear correction factor discussed in Section 4.1.1, the average accuracy of all 25 data points is increased to 17.3cm. If we look at only the 9 points closest to the corner (grid size of 1.2m x 1.2m), our corrected accuracy is 9.53cm.

To calculate the precision, we first computed the variances of the detected *X* and *Y* values. *var(X)* and *var(Y)* are the squared mean distances of *X* and *Y* from the mean. Because the data is in a Cartesian coordinate frame, the Euclidean distance can be applied, and the average distance of all samples from their mean is

$$std(X,Y) = \sqrt{var(X) + var(Y)} .$$

Within a working range of 2m x 3m the precision of the system is at most 15.8cm (approximately 3 cm – 4cm in each direction) for 90% of the readings (see Figure 9). The device performs the best when it is close to at least one wall when pointed at a corner. The reason for the less accurate results when compared to the single wall experiment is that fewer points were being used to define each plane, as the grid pattern is distributed across two different walls. An accuracy of 10 to 17cm with 15cm of precision is still significantly better than other indoor location systems that do not require the deployment of infrastructure. By using multiple TrackSense units, each plane in the room would be illuminated by more feature points. This would increase the total accuracy and precision, approaching the performance of the single-wall experiments.

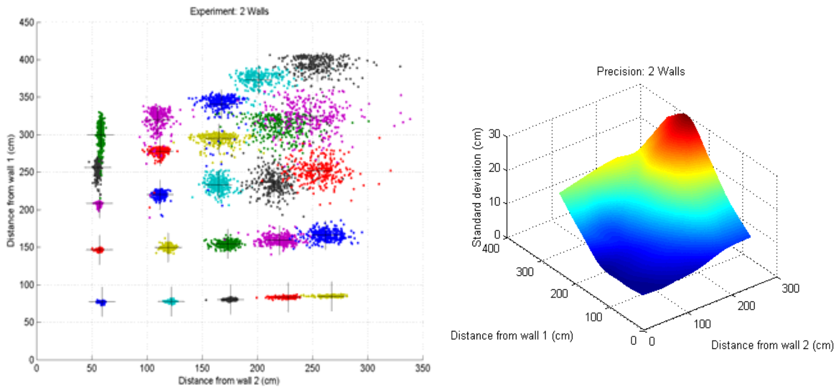

Figure 9: Two wall experiment. Left: Each dot represents a single data sample. Dots of the same color belong to the same position data set and black crosshairs show mean values. Right: Up front view of interpolated standard deviation of the two wall readings**.**

5 Miniaturization and Addressing Limitations

In this section, we discuss miniaturizing this system and present two realistic prototypes: a handheld device and a head-mounted unit. We also discuss the limitations of the current prototype and show how we would extend the system to cover an entire room. Finally, we consider some limitations that are beyond simple engineering considerations.

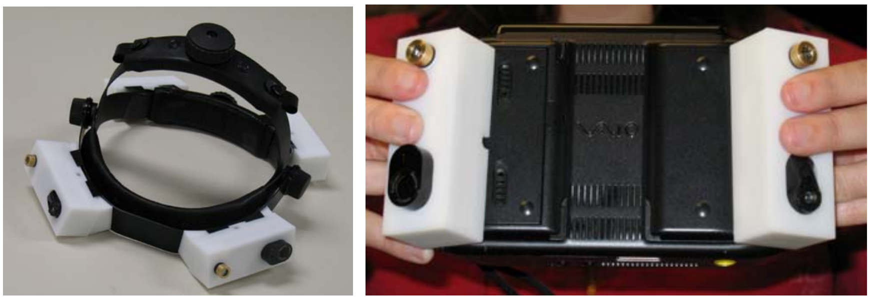

Figure 10: Headset and handheld with active components on miniaturized TrackSense units.

5.1 Miniaturization and Improvements

Our system is composed of relatively simple parts. We chose to use a projector in our prototype because it gave us the flexibility to experiment with various patterns quickly. Because the projected pattern is static, we can replace the projector with an infrared grid laser diode. The infrared diode would eliminate the visible patterns and increase the range because it is brighter. We would replace the camera with a smaller, black-and-white camera with an infrared pass filter and could place multiple laser/camera units on the localized device as a result of the miniaturization (see Figures 1 and 10). Theoretically, a single TrackSense unit with a wide enough field of view could always have at least one corner in its field of view, permanently maintaining a position and orientation fix. Practically, we expect several TrackSense units angled in different directions to cooperatively identify three planes within their combined views, interpolate the location of the (possibly non-observed) corner, and determine their location and orientation regardless of their platform's motion. For the handheld unit (see Figure 10), we placed two TrackSense units facing forwards and angled 45 degrees away from center. This configuration ensures that two walls are detected at any given time for proper localization within the entire room. A third camera facing up or down enables full 3D positioning. Another strategy is to slightly angle the two front facing units up to capture the wall and ceiling corners, eliminating the need for the third. However, this solution would also limit how much the user can tilt the handheld forward.

On a head-mounted device (see Figure 10), we can place four units looking at 90 degree intervals. The units could be angled slightly upwards for full 3D positioning or a fifth could be added pointing vertically. The advantage of the head mounted unit is that more units can be installed facing in opposite directions, which would result in better precision in a larger room. As we saw in the results, the farther the device is from a wall, the lower the precision and accuracy. Since the head mounted device has a full view in all directions, the system can select the closest walls to offer the best results.

5.2 Limitations

There are still several limitations of our approach worthy of further examination. The current solution only supports one device in a room at a time. This might be acceptable for some applications, but not for multiplayer games or collaborative applications. One solution is to synchronize the devices and have them alternately flash their patterns. Since the devices know their position within the room, the devices can turn off certain parts of the grid to avoid interfering with another device.

An important limitation to our approach is the need for walls in the space. The wall has to be free of major obstructions and large windows. In our experience, posters and other flat objects do not cause major problems, and our implementation can detect outliers. However, many raised objects on the wall cause the system to incorrectly identify the plane. The current technique also assumes a flat surface with little to no curvature, limiting the types of rooms that are appropriate. However, our intent with this device is to enable applications where a user would already want to interact with multiple large surfaces that are relatively plain in the first place [5]. If the intent is to extend our solution to more complex spaces we can incorporate stereo vision techniques that work well in cluttered environments and use both approaches in a complementary fashion.

Some dark wall colors cause problems for detecting the grid. Very bright lighting conditions (*i.e.*, near a window during the day) can make the projected lines too faint. However, most artificial lighting from standard fluorescent and incandescent lights does not cause major problems with detection. Another concern is very tall ceilings, in which case the units would have to be oriented to detect walls and floor corners. To obtain three dimensions, we must select a unit to face downwards. The detector camera resolution also limits how far a unit can be from a wall, thus limiting the working range in a room. Our experimental prototype was limited to a room approximately 5m x 5m in size. A higher resolution camera and the use of a laser grid would improve those limits.

6 Conclusion

We presented TrackSense, a localization system that requires no additional infrastructure in the environment and provides 3D positioning and orientation data that compares favorably against existing solutions. Inspired by robotics localization and camera-projector calibration techniques, our solution uses a camera to locate and track a grid pattern projected onto surfaces in the camera's field of view to determine its distance and orientation to multiple fixed large planes in a space (*i.e.*, walls and ceilings). A system of TrackSense units can obtain up to 4 cm accuracy with 3 cm precision in rooms up to 5 square meters, as well as 2 degree accuracy and 1 degree precision on orientation. The relatively simple hardware used in its implementation makes miniaturization possible.

TrackSense provides localization within a room, but combining it with a room-level localization system, such as WiFi or GSM fingerprinting, can provide localization within a global coordinate frame. In addition, our solution provides a useful complement to traditional stereo vision techniques, which do not perform well on plain surfaces. The addition of another camera would provide localization within both a cluttered and uncluttered environment, thus extending the capabilities of the device further.

References

1. Active Bat. The BAT Ultrasonic Location System. http://www.uk.research.att.com/bat/. 2006.
2. Bahl, P. and Padmanabhan, V. RADAR: An In-Building RF-Based User Location and Tracking System. In the proc of *IEEE Infocom 2000*. Los Alamitos. pp. 775-784. 2000.
3. Bouguet, J.Y., and Perona, P. 3D photography on your desk. ICCV'98, pp. 43–50, 1998.
4. Canny, J. A Computational Approach to Edge Detection. In IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 8, No. 6, Nov. 1986.
5. Cao, X. and Balakrishnan R. Interacting with dynamically defined information spaces using a handheld projector and a pen. In the proc. of UIST 2006. pp. 225-234. 2006.
6. Cobzas, D. and Sturm, P. 3D SSD Tracking with Estimated 3D Planes, In the proc. of *2nd Canadian Conference on Computer and Robot Vision (CRV) 2005*, pp. 129-134. 2005.
7. Davison, A., Cid, Y., Kita, N. Real-Time 3D SLAM with Wide-Angle Vision. In proc. of IFAC Symposium on Intelligent Autonomous Vehicles. 2004.
8. Dellaert, F., and Tariq, S. A Multi-Camera Pose Tracker for Assisting the Visually Impaired In 1st IEEE Workshop on Computer Vision Applications for the Visually Impaired. 2005.
9. Dissanayake, M. W. M. G., Newman, P., Clark, S., Durrant-Whyte, H. F., and Csorba, M. A solution to the simultaneous localization and map building (slam) problem. IEEE Transactions on Robotics and Automation, 17, 3 (June 2001), pp. 229-241.
10. Fischler, M. and Bolles, R. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. In Communications of the ACM. Volume 24, Number 6, pp. 381-395. 1981.
11. Forsyth, D.A., Ponce, J. Computer Vision: A Modern Approach. Prentice Hall, U.S. 2002.
12. Harle, R.K., A. Hopper. Cluster Tagging: Robust Fiducial Tracking for Smart Environments. 2nd International Workshop on Location- and Context-Awareness 2006. pp. 14-29. May 2006.
13. Hartley, R., Zisserman, A. Multiple View Geometry in Computer Vision. Cambridge Press. 2003.
14. Hightower, J. and Borriello, G. A Survey and Taxonomy of Location Systems for Ubiquitous Computing, University of Washington Tech Report CSC-01-08-03. 2001.

15. LaMarca, A., Chawathe, Y., Consolvo, S., Hightower, J., Smith, I., Scott, I., Sohn, T., Howard, J., Hughes, J., Potter, F., Tabert, J., Powledge, R., Borriello, G., and Schilit, B. Place Lab: Device Positioning Using Radio Beacons in the Wild. In the proc. of *Pervasive 2005*, pp.. 116-133. 2005.
16. Kuhn, H.W. The Hungarian Method for the assignment problem. Naval Research Logistic Quarterly, Issue 2. pp. 83-97. 1955.
17. Lourakis, M.A., Argyros, A.A. Vision-Based Camera Motion Recovery for Augmented Reality. In the proc of the Computer Graphics International (CGI 2004). pp. 569-576. 2004.
18. Liu, Y. and Thrun, S. Results for outdoor-SLAM using sparse extended information filters. In Proc of the IEEE International Conference on Robotics and Automation. pp. 1227-1233. 2003.
19. Madhavapeddy, A. and Tse, T. Study of Bluetooth Propagation Using Accurate Indoor Location Mapping. *UbiComp 2005.* Tokyo, Japan. pp. 105-122. September 2005.
20. Munkres, J. Algorithms for the Assignment and Transportation Problems. Journal of the Society of Industrial and Applied Mathematics, 5(1). pp. 32-38. 1957.
21. NorthStar. Evolution Robitcs. http://www.evolution.com/products/northstar/. 2007.
22. Otsason, V., Varshavsky, A., LaMarca A., and de Lara, E. Accurate GSM Indoor Localization. In the proc. of *UbiComp 2005*. Tokyo, Japan. pp. 141-158. September 2005.
23. Patel, S.N., Rekimoto, J., Abowd, G.D. iCam: Precise at-a-Distance Interaction in the Physical Environment. In the proc. of Pervasive 2006, pp. 272-287, May 2006.
24. Patel, S.N., Truong, K.N., and Abowd, G.D. PowerLine Positioning: A Practical Sub-Room-Level Indoor Location System for Domestic Use. In the proc. of *Ubicomp 2006*. pp. 441-458. 2006.
25. Priyantha, N. B., Chakraborty, A., and Balakrishnan, H. The Cricket Location-Support System. In the proc. of *Mobicom 2000*. Boston, MA. August 2000.
26. Rekimoto J. and Ayatsuka Y. CyberCode: Designing Augmented Reality Environments with Visual Tags. In the proc. of *DARE 2000*. Elsinore, Denmark. pp. 1 – 10. 2000.
27. Rachmielowski, A., Cobzas, D., Jagersand, M. Robust SSD tracking with incremental 3D structure estimation. In the proc. of *Canadian Conference on Computer and Robot Vision (CRV)*. 2006.
28. Rekimoto, J and Katashi, N. The World through the Computer: Computer Augmented Interaction with Real World Environments. In the proc. of *UIST 1995*. Pittsburgh, PA. pp. 29-36. 1995.
29. Scharstein, D and Szeliski, R. High-accuracy stereo depth maps using structured light. In CVPR 2003, volume 1, pp. 195-202, Madison, WI, June 2003.
30. Se, S., Lowe, D., Little, J. Vision-Based Mobile Robot Localization and Mapping Using Scale-Invariant Features. In the Proc of ICRA 2001. 2001.
31. Sukthankar, R., Stockton, R., Mullin, M. Smarter Presentations: Exploiting Homography in Camera-Projector Systems. In the proc. of ICCV 2001.
32. Sukthankar, R., Stockton, R., Mullin, M. Automatic Keystone Correction for Camera-assisted Presentation Interfaces. In the proc. of International Conference on Multimodal Interfaces. 2000.
33. Tariq, S. and Dellaert, F. A Multi-Camera 6-DOF Pose Tracker. In IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR). pp. 296-297. 2004.
34. Thrun, S., Burgard, W., and Fox, D. Probabilistic Robotics. MIT Press, Cambridge, MA, 2005.
35. Ubisense. http://www.ubisense.net. 2006.
36. Vicon MX. http://www.vicon.com/products/systems.html. 2006.
37. Want, R., Hopper, A., Falcao, V., and Gibbons, J. The active badge location system. *ACM Transactions on Information Systems*. Volume 10. pp. 91-102. January 1992.