2011 American Control Conference
on O'Farrell Street, San Francisco, CA, USA
June 29 - July 01, 2011

Localization Through Fusion of Discrete and Continuous Epipolar Geometry with Wheel and IMU Odometry

Jinglin Shen, David Tick and Nicholas Gans

Abstract— **This paper describes a novel sensor fusion implementation to improve the accuracy of robot localization by combining multiple visual odometry approaches with wheel and IMU odometry. Discrete and continuous Homography Matrices are used to recover position, orientation, and velocity from image sequences of tracked feature points. An Inertial Measurement Unit (IMU) and wheel encoders also measure linear and angular velocity of mobile robot. A Kalman filter fuses the measurements from the visual and inertial measurement systems. Time varying matrices in the Kalman filter allow each sensor to receive higher or lower weight in situations where each is more or less accurate. Experiments are performed with a camera and a IMU (Wiimote controller) mounted on a mobile robot.**

I. INTRODUCTION

Building mobile robotic systems that are capable of real-time autonomous navigation is a complex, multi-faceted problem. One of the primary aspects of this problem is the task of localization. Localization, or pose-estimation, refers to the task of estimating the position, orientation and velocity (both angular and linear) of the robot over time. There are many established ways to approach the task of localization including wheel odometry [1], inertial sensors [1], GPS [1], sonar [2], and IR/laser-based range finding sensors [3].

There have been significant results in localization techniques based solely on vision data. Visual odometry is a method of localization that uses one or more cameras to constantly capture images or video frames taken of a scene [4]. The frames are analyzed sequentially using various computer vision techniques. The analysis of these frames estimates the angular and linear velocities of the camera between each time step. These velocities are integrated over time to estimate how the camera has moved.

Some vision-based localization techniques are designed to calculate the pose of the camera relative to a well known reference object that appears in each frame [5], [6]. These techniques are usually dependent on accurate prior knowledge regarding the geometric properties of the scene or a reference object. In many situations, there exists little accurate prior knowledge regarding a scene and the objects therein. In such cases, two frames can be compared with one another based on a set of feature points which exists in both frames. Once a set of feature points is identified in both frames, a mapping of the feature points from one frame onto the next must exist. This mapping encapsulates the rotation and translation of the camera that occurs between the taking of each picture. In practice, the Essential matrix or Euclidean Homography Matrix is used to estimate a camera's pose in terms of rotational and translational transformations [7], [8].

Over the years, numerous control systems have been designed that utilize Essential or Homography Matrices in vision-based robotic tasks [9]–[17]. Of particular interest, is the application of the Euclidean Homography Matrix as a means of estimating velocity [18]. The Euclidean Homography Matrix can be used in both a continuous and a discrete form [8]. However, it seems that there has not been as much work done regarding applications of continuous Homography Matrix. If one does a discrete homography-based estimation of the camera position and orientation (i.e. pose), then one can also integrate the continuous estimate of the velocity at each time step to extract the pose as well. Discounting the effects of noise, *the pose estimate obtained from integrating the continuous homographic estimate of angular velocity must agree with the discretely estimated position of the camera*. This notion was initially explored in [19], and verified with simulations.

A pose estimate produced in this manner can be further enhanced by combining visual odometry with traditional wheel odometry and other sensor devices. This can be achieved through various signal processing techniques known as sensor fusion. The Kalman filter is particularly useful for fusing signals from multiple sensors and removing errors in localization that occur due to many factors such as sensor noise, quantization, flawed process modeling, and sensor bias or drift [1], [20].

Vision-based Kalman filter approaches have been used for years in robot localization and mapping problems. Soatto et al. give two Kalman filter approaches for motion estimation based on the Essential matrix [21]. Some other methods use image features as inputs to the Kalman filter with pose and/or velocity as output. These methods are more computationally efficient and avoid the constraints inherent to the pose reconstruction methods [22], [23]. Zhang et al. use a trinocular stereo system mounted on a mobile robot to build a local map of the environment while the robot explores it [24]. A Kalman filter is used to merge matched line segments from the trinocular stereo system. In [25], a multi-sensor, on-the-fly, localization system is proposed to fuse estimations from visual sensors with other types of sensors, such as laser range finders. Furthermore, Gutmann developed a vision-based Markov-Kalman method for robots observing known landmarks, which combines the accuracy

This work was not supported by any organization

D. Tick is with the Department of Computer Science, University of Texas at Dallas, Dallas, TX, USA dqt081000@utdallas.edu

J. Shen and N. Gans are with the Department of Electrical Engineering, University of Texas at Dallas, Dallas, TX, USA sxj096220@utdallas.edu,ngans@utdallas.edu

of Kalman filter and robustness of Markov method [26].

In this paper, we propose a novel localization technique involving vision-based estimates of pose and velocity from continuous and discrete Homography matrices, an IMU, and wheel encoders. The visual odometry method uses a single camera rigidly mounted on the robot. The change in camera pose is estimated by using the discrete form of the Euclidean Homography Matrix, while the continuous form of the Euclidean Homography Matrix is utilized to estimate the velocity of the camera over time. The IMU measures angular velocity, and the wheel encoders of the mobile robot measure linear and angular velocity. The system utilizes a Kalman filter to fuse all the estimates and remove error which accrues over time due to integration, noise, and quantization [20]. Furthermore, a low-cost IMU is used, specifically a Nintendo Wiimote. The results obtained, despite the poor performance of the IMU, show the strength of the system and the possibility of low-cost alternatives for localization. Some sensors are more accurate in certain regimes of operation. The proposed system alters the covariance matrices used in the Kalman filter based on the system state. This allows sensors that are currently more accurate to receive a higher weighting in the sensor fusion. This results in a more accurate estimate and allows for improved estimation of bias on certain sensors.

Section II will clarify the terminology used in this paper as well as formally define and model the problem. Section III will explain the proposed sensor fusion approach. We present experiment results in section IV that illustrate the effectiveness of the proposed system. Finally, in section V we will espouse the conclusions reached from this work.

II. BACKGROUND

Different conventions exist for the notation and terminology used in the vision-based estimation and robot control. For example, the vision community typically assigns a Cartesian reference frame to a camera with the z-axis oriented along the optical axis, while the robotics community typically assigns a reference frame to a wheeled mobile robot with the z-axis oriented perpendicular to the plane of motion. In this paper, we follow the separate conventions for the vision system and mobile robot when discussing the development of each. The coordinate system of the camera is mapped to the robot frame through a constant rotation.

A. Formal Definition and Terminology

Navigation sensors, including cameras, make measurements with respect to an inertial frame and measured in a moving body frame, here called $\mathcal{F}_b$ for camera body frame and $\mathcal{F}_r$ for robot body frame. These measurements are then rotated to obtain localization with respect to the world frame, here called $\mathcal{F}_w$. In this paper, the orientation and position (collectively referred to as pose) of a body frame with respect to the world frame is expressed as a function of time by including the index number for a given time step in parentheses. For example, in Fig. 1 $\mathcal{F}_b(t_0)$ describes the pose of the body frame $\mathcal{F}_b$ attached to a camera at time index t_0 as measured from $\mathcal{F}_b(t_0)$, while $\mathcal{F}_b(t_k)$ describes the pose of $\mathcal{F}_b$ at time t_k as measured from $\mathcal{F}_b(t_0)$.

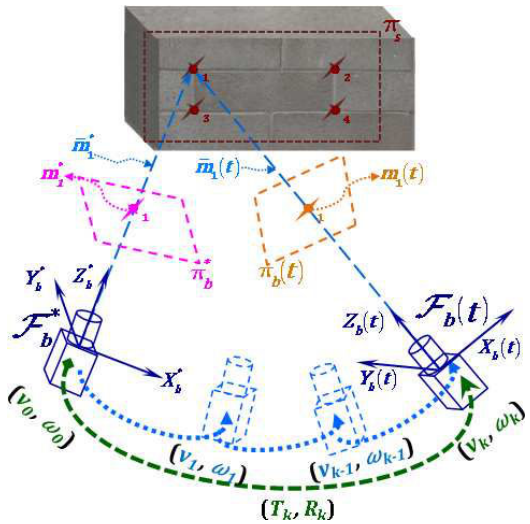


Fig. 1. Translation and Rotation (Tk, Rk) of Body/Camera Frame $\mathcal{F}_b$ int planar scenes

The changes in the pose of $\mathcal{F}_b$ that occur over the time interval $[t_0, t_1, ..., t_{k-1}, t_k]$ are described in terms of a translation vector $T_k \in \mathbb{R}^3$ and a rotation matrix $R_k \in SO(3)$. At any given time t_k, the instantaneous linear and angular velocities of the body frame $\mathcal{F}_b$ are described, as a pair of vectors (v_k, ω_k), where $v_k \in \mathbb{R}^3$ and $\omega_k \in \mathbb{R}^3$. We will use subscript k to denote the value at time step t_k for other variables in the sequel.

In this work, a camera is rigidly attached to a wheeled mobile robot. The Cartesian reference frame of the camera, $\mathcal{F}_b$, is oriented such that the z-axis is oriented along the optical axis, the x-axis is oriented along the horizontal direction of the image plane, and the y-axis is oriented parallel to the vertical direction of the image. This is illustrated in Fig. 1. The reference frame of the robot, $\mathcal{F}_r$, is attached to the robot's center of rotation, with x-axis aligned along the robot's heading, y-axis oriented to the left of the robot along the horizontal direction, and z-axis oriented upwards along vertical direction. $\mathcal{F}_r(t_k)$ denotes the pose of $\mathcal{F}_r$ at time t_k. This is illustrated in Fig. 2.

Without loss of generality, the origin of $\mathcal{F}_b$ and $\mathcal{F}_r$ are the same. As discussed, the z-axis of $\mathcal{F}_b$ is coincident with the x-axis of $\mathcal{F}_r$,and the x-axis of $\mathcal{F}_b$ is coincident with the y-axis of $\mathcal{F}_r$. Thus they are separated by a constant rotation R_{br}. An IMU is rigidly attached to the robot, with its axes aligned with $\mathcal{F}_r$. Only rotations are measured by the IMU in this work. In this paper, the static world frame $\mathcal{F}_w$ is defined as the initial pose of the robot, i.e. $\mathcal{F}_w = \mathcal{F}_r(t_0)$.

B. Robot, Wheel Encoders and IMU

The wheeled robot follows the kinematic unicycle model [27] and moves in a plane spanning the x and y axes of $\mathcal{F}_w$ and $\mathcal{F}_r$. It has two degrees of freedom. It can rotate about the z-axis of $\mathcal{F}_r$ with an angular velocity ω_{zk} and translate along the x-axis with a linear velocity v_{xk} with respect to $\mathcal{F}_r$. The location of the robot in $\mathcal{F}_w$ is the origin of $\mathcal{F}_r(t_k)$,

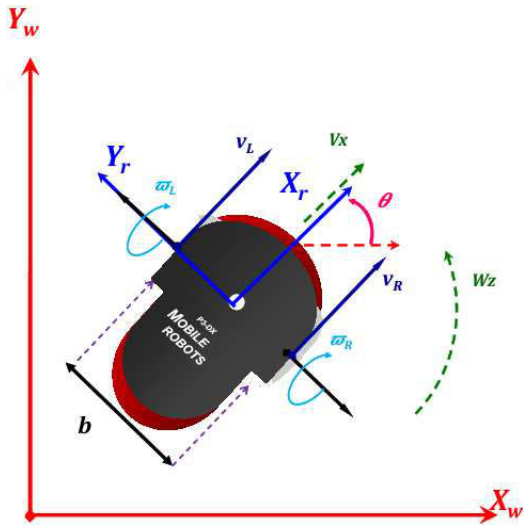


Fig. 2. Translation and Rotation of Robot Frame $\mathscr{F}_r$

given by $[x_{pk}, y_{pk}, 0]^T$. The orientation of the robot is θ_k, and is measured by the angle between the x-axes of $\mathscr{F}_w$ and $\mathscr{F}_r$. This is illustrated in in Fig. 2.

Wheel encoders are used to measure the angular velocity of each wheel and can provide information for odometry and localization. The angular velocity of the left and right wheel are given by ω_{Lk} and ω_{Rk} respectively, as shown in Fig. 2. These measurements are converted into linear velocity of the wheels by

$$v_L = \omega_L \times r$$
$$v_R = \omega_R \times r$$

where $r \in \mathbb{R}^+$ is the constant radius of the wheels. Denoting the constant distance between the wheels as $b \in \mathbb{R}^+$, the linear and rotational velocity of the robot is given by

$$v_x = \frac{v_L + v_R}{2} \tag{1}$$
$$\omega_z = \frac{v_L - v_R}{b}. \tag{2}$$

By numerically integrating the estimates of v_{xk} and ω_{zk} over time, the pose estimate of the robot in $\mathscr{F}_w$ is given by

$$\begin{bmatrix} x_{pk} \\ y_{pk} \\ \theta_k \end{bmatrix} = \sum_{i=t_0}^{t_k} \left\{ \begin{bmatrix} -sin\theta_k \Delta t & 0 \\ cos\theta_k \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} \begin{bmatrix} v_{xk} \\ \omega_{zk} \end{bmatrix} \right\}$$

where $\Delta t \in \mathbb{R}^+$ is the time different between two time steps.

The IMU detects linear acceleration using accelerometers, and detects angular velocities using gyroscopes. Due to the limited motion of the robot, we use only one of the gyros to measure angular velocity ω_z. The orientation of the robot, as estimated using the rate gyros, is given by

$$\theta_k = \sum_{i=t_0}^{t_k} \omega_{zk} \Delta t.$$

C. Pinhole Camera Model

Fig. 1 illustrates a camera taking two images from two different poses $\mathscr{F}_b^*$ and $\mathscr{F}_b(t_k)$. $\mathscr{F}_b^*$ is considered a static reference frame, such as the pose at time $t = 0$. $\mathscr{F}_b(t_k)$ is considered a moving frame or current frame. The changes which exist between the two poses are encapsulated by (T_k, R_k). Fig. 1 also shows a collection of feature points that all lie in the plane π_s. The 3D coordinates of each feature point, as measured from the reference frame $\mathscr{F}_b(t_k)$, are defined as vectors $\overline{m}_j(t_k) \in \mathbb{R}^3$. Similarly, the 3D coordinates of each feature point as measured from $\mathscr{F}_b^*$ are defined in terms of constant vectors $\overline{m}_j^* \in \mathbb{R}^3$. Formally these vectors are given as

$$\begin{aligned} \overline{m}_j^* \in \mathbb{R}^3 &= [x_j^*, y_j^*, z_j^*]^T, \forall j \in \{1, \ldots, N\} \\ \overline{m}_j(t) \in \mathbb{R}^3 &= [x_j(t), y_j(t), z_j(t)]^T, \forall j \in \{1, \ldots, N\}. \end{aligned}$$

Two images are captured by the camera at the two poses $\mathscr{F}_b^*$ and $\mathscr{F}_b(t_k)$. This is modeled by projecting the feature points onto the 2D image planes. The coordinates of the feature points in these 2D planes are expressed as a normalized set of 3D coordinates, where depth along the z-axis is normalized to one. The normalized image plane coordinates, as measured from $\mathscr{F}_b(t_k)$ are defined as vectors $m_j(t_k) \in \mathbb{R}^3$. Similarly, the normalized image plane coordinates of each feature point measured from $\mathscr{F}_b^*$ are vectors $m_j^* \in \mathbb{R}^3$. These vectors are expressed as

$$\begin{aligned} m_j^* \in \mathbb{R}^3 &= \left[\frac{x_j^*}{z_j^*}, \frac{y_j^*}{z_j^*}, 1 \right]^T, \forall j \in \{1, \ldots, N\} \\ m_j(t) \in \mathbb{R}^3 &= \left[\frac{x_j(t)}{z_j(t)}, \frac{y_j(t)}{z_j(t)}, 1 \right]^T, \forall j \in \{1, \ldots, N\}. \end{aligned}$$

D. Discrete and Continuous Euclidean Homography

In this work, a visual odometry method is developed from the discrete and continuous Euclidean Homography matrices. This gives a vision-based estimate of position, orientation, and linear and angular velocities. While the discrete homography has been used extensively, the continuous homography seems to attract less attention.

Both planar homography cases assume all feature points are coplanar[1]. Consider two images of points $\overline{m}(t_k)$ on plane π_s. The transformation between the two images is given by [7], [8]

$$\overline{m}_j = R\overline{m}_j^* + T. \tag{3}$$

The relationship in (3) can be rewritten in terms of image points as

$$m_j = (R + \frac{1}{d^*} T n^{*T}) m_j^*$$

where $n^* = [n_x^*, n_y^*, n_z^*]^T$ is the constant unit normal vector of plane π_s measured in $\mathscr{F}_b^*$, and d^* is the constant scalar distance from the optical center of the camera (i.e. the origin of $\mathscr{F}_b^*$) to the plane π_s [7]. We assume that d^* is known in this initial investigation[2]. The matrix

$$H_d = R + \frac{1}{d^*} T n^{*T}$$

[1]Methods to move beyond the assumption of coplanar points include virtual parallax [16] or using continuous and discrete Essential matrices. This is an avenue of future work.

[2]Depth estimation from motion is an open, but well investigated topic [28], and will be included in future work

is the well known Euclidean Homography Matrix, which we will refer to as the *discrete Homography Matrix*. Given four or more points, the matrix $H_d(t_k)$ can be recovered and used to estimate the translation vector T_k and the rotation matrix R_k [7].

In the continuous case, image point $m_j(t_k)$ and its optical flow $\dot{m}_j(t_k)$ are measured rather than the image pair m_j^* and $m_j(t_k)$. The time derivative of $\overline{m}_j(t_k)$ satisfies

$$\dot{\bar{m}}_j = \hat{\omega}\bar{m}_j + v \tag{4}$$

where $\hat{\omega}(t_k)$ is the skew-symmetric matrix of $\omega(t_k)$. The relationship in (4) can be rewritten as [8]

$$\dot{m}_j = (\hat{\omega} + \frac{1}{d^*} v n^{*T}) m_j - \frac{\dot{z}_j}{z_j}.$$

The matrix

$$H_c = \hat{\omega} + \frac{1}{d^*} v n^{*T}$$

is defined as the *continuous Homography Matrix*. Similar to the discrete form, an algorithm exists to solve for $H_c(t_k)$ and estimate the linear velocity v_k and angular velocity ω_k of the camera.

III. APPROACH

We propose a localization method through fusion of vision-based pose and velocity estimates with velocity estimates from an IMU and wheel odometry. The simulation results from the fusion of vision-based estimates of pose and velocity showed promise in [19]. By fusing the vision system with an IMU and wheel odometry, the addition of pose measurements and multiple velocity measurements will improve the accuracy of localization. A Kalman filter is used to perform the sensor fusion and incorporate the known kinematics of the system to reduce the effects of sensor noise.

The system is described by the discrete system equations

$$x_k = F_k \cdot x_{k-1} + q_k \tag{5}$$

$$y_k = H_k \cdot x_k + r_k. \tag{6}$$

In (5) and (6), x_k is the state vector, F_k is the state transition matrix, y_k is the measurement vector, H_k is the measurement matrix, and q_k and r_k are normally distributed random processes with zero mean and covariance matrices Q_k and R_k respectively.

Localization of a kinematic unicycle mobile robot requires knowledge the robot position x_{pk} and y_{pk} and rotation θ_k, all measured in the world frame $\mathscr{F}_w$. Robot velocity v_{xk} and angular velocity ω_{zk} are included in the state, as measured in the robot frame. A bias term $\beta_{\omega k}$ is included in the state vector to cancel out the effect of the IMU measurement drift.

The state vector is given by

$$x_k = \left[\theta_k, \beta_{\omega k}, \omega_{zk}, x_{pk}, y_{pk}, v_{xk}\right]^T \in \mathbb{R}^6.$$

The state time-varying transition matrix $F_k \in \mathbb{R}^{6\times 6}$ is given by

$$F_k = \begin{bmatrix} 1 & 0 & \Delta_t & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & \cos\theta_k \Delta_t \\ 0 & 0 & 0 & 0 & 1 & \sin\theta_k \Delta_t \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

where Δ_t is the time different between t_{k-1} and t_k. The state process noise covariance matrix $Q_k \in \mathbb{R}^{6\times 6}$ is defined as

$$Q_k = \begin{bmatrix} \sigma_3^\omega & 0 & \sigma_2^\omega & 0 & 0 & 0 \\ 0 & \sigma_1^\beta & 0 & 0 & 0 & 0 \\ \sigma_2^\omega & 0 & \sigma_1^\omega & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_3^v & 0 & \cos\theta_k \sigma_2^v \\ 0 & 0 & 0 & 0 & \sigma_3^v & \sin\theta_k \sigma_2^v \\ 0 & 0 & 0 & \cos\theta_k \sigma_2^v & \sin\theta_k \sigma_2^v & \sigma_1^v \end{bmatrix}.$$

The process noise is designed as random walks between the angular velocity and orientation, and linear velocity and position. According to the random walk model, the change of the state vector over time is constrained since some states depend on each other, i.e. θ_k depends on ω_{zk} and x_{pk}, y_{pk} depend on v_{xk}. The off-diagonal elements in Q_k reflect such constraints. The process noise on $\beta_{\omega k}$ is independent of the other states. The elements of Q_k are defined as

$$\sigma_2^\omega = \frac{(\sigma_1^\omega)^2}{2}, \quad \sigma_3^\omega = \frac{(\sigma_1^\omega)^3}{3}$$

and σ_1^ω is a tunable constant reflecting the expected amount of change in ω. The terms σ_i^v(i=1,2,3) and σ_1^β are defined in the same manner for linear velocity and bias.

In system equation (6), the measurement vector y_k is defined as

$$y_k = [\omega_w, \omega_g, v_w, v_c, \omega_c, \theta_d, x_d, y_d]^T \in \mathbb{R}^8. \tag{7}$$

In (7), ω_w, ω_g, ω_c are measurement of angular velocity ω_{zk} from the wheel encoders in (2), rate gyro, and the continuous Homography respectively. The terms v_w and v_c are measurement of linear velocity v_{xk} from wheel encoders and continuous Homography. Since different conventions were used to define camera and robot frame, all measurements made from the vision-based estimation in y_k are converted from the camera frame to the robot frame by the constant rotation matrix R_{br}. The measurement matrix $H_k \in (R)^{8\times 6}$ is defined accordingly as

$$H_k = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

The measurement noise covariance matrix $R_k \in \mathbb{R}^{8\times 8}$ is a diagonal matrix. For simplicity, the diagonal elements are listed as a vector

$$diagonal(R_k) = \left[\sigma_w^\omega, \sigma_g^\omega, \sigma_w^v, \sigma_c^v, \sigma_v^\omega, \sigma_d^\theta, \sigma_d^x, \sigma_d^y\right] \in \mathbb{R}^8.$$

The elements of R_k vary with according to the state of the robot. Namely, σ_w^ω and σ_w^v are defined as

$$\sigma_w^\omega = (\delta_w^\omega * \Delta_t * \omega_w) + (\delta_w^\omega * \Delta_t * v_w) \tag{8}$$

$$\sigma_w^v = (\delta_w^v * \Delta_t * \omega_w) + (\delta_w^v * \Delta_t * v_w) \tag{9}$$

where δ_w^ω and δ_w^v are constants. The other σ terms are constants tuned according to the variance of each measurement.

The value of diagonal elements in R_k determine the effect of the corresponding measurements at the update step of Kalman filter. A noisier measurement is considered to be less trustworthy. Increasing the corresponding element in R_k will give less weight to a less trustworthy measurement during the Kalman filter update.

It is known that the wheel encoders provide accurate measures of linear and angular velocity when the robot has no motion. However, when the robot is in motion, measurements from wheel encoders become less trustworthy. Equation (8) and (9) show that when linear and angular velocity measurements from wheel encoders are both zero, the robot is still, and σ_w^{ω} , σ_w^{v} are zero, which means the wheel encoder estimation dominates the other two sensors modalities. If the robot is in motion, the values of σ_w^{ω} , σ_w^{v} are proportional to velocity, therefore wheel encoders are trusted less, and measurements from gyros and vision-based estimation are taken into account to improve the estimates.

For comparison, a Kalman filter without visual odometry method is also designed. The state vector and matrices in (5) remain the same. There are only three measurements, namely the first three measurements in (7). Therefore, the measurement vector of the reduced system $y_k \in \mathbb{R}^3$ and the measurement matrix H_k are made of first three rows of H_k. Comparison of the experimental results for each system is given in next section.

IV. EXPERIMENTAL RESULTS

Experiments were performed to test the proposed localization method. For comparison purposes, the full system fusing vision-based estimation, IMU measurements and wheel encoders is compared to the reduced system fusing only IMU and wheel encoders. The robot used in experiments is a Pioneer 3-DX differential drive robot. A laptop with Intel Core Duo processor was fixed to the robot and collected all measurements as well as performed all signal processing and Kalman filter operations. The IMU is a Nintendo Wii Remote (Wiimote) with MotionPlus. This IMU costs approximately $42 (U.S. dollars), and represents an extremely low cost option for an IMU.

A. Testing the Reduced System

In order to establish the performance of the reduced system, simple experiments were performed. A human operator drove the robot along a simple path in the hallways of the Engineering and Computer Science building at the University of Texas at Dallas. The path begins facing east and makes four left turns of 90° to return to the starting point. The total distance is approximately 100m.

Fig. 3 is a cropped section of an image that the reduced system renders while in operation. The background image is from Google maps, showing the location of the experiment. As seen in the legend of Fig.3 the star denotes the starting point of the robot. The green dashed line was added afterwards in order to show the path that the robot was driven along. The red line represents the path estimated using only the wheel encoders to measure angular velocity. The blue

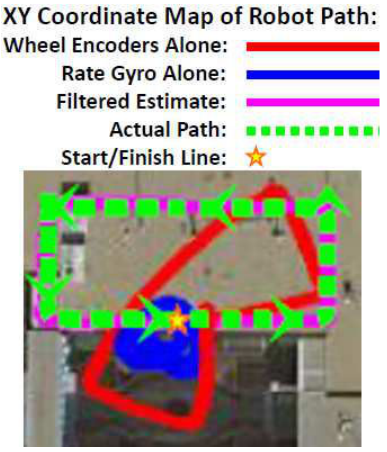


Fig. 3. Fusion of Wheel Encoders with IMU

"blob" represents the path of the robot estimated by using only the Wiimote's measurement of angular velocity. The magenta line represents the path of the robot estimated using the reduced system's Kalman filter-based fusion of both sets of angular velocity measurements. The wheel encoders have several sources of error, which cause the path estimate to appear crooked. As the distance traveled increases, the encoders become less and less reliable. The MEMS rate-gyros inside the Wiimote's have a severe bias. This causes the Wiimote's estimated path to appear as circling around in an epicyclic manner. Inspection of Fig.3 shows that the Wiimote estimate is useless on its own. Despite this, the reduced system manages to create an accurate and useful estimate of its path simply by fusing the devices' faulty measurements and estimating the bias of the IMU. This gives a good example that the reduced system is accurate. The addition of pose and orientation estimates and an additional set of velocities measurements will ideally improve the estimate further.

B. Results of the Full System and Comparison With the Reduced System

The full system is described in the body of this paper. The experimental setup includes the same equipment described earlier. The Wiimote and camera are mounted on top of the robot. The camera used is a Matrix Vision BlueFox with USB 2 interface. It has a resolution of 1024×768 pixels. The frame rate of the camera, when including all image processing and Kalman filter calculations, is 24 frames/second. The camera is calibrated to learn its intrinsic parameters and lens distortion parameters. In order to establish an accurate baseline across multiple experiments, two laser pointers are mounted on the robot facing the floor, one on the front and one on the back.

The camera is mounted such that the center of the camera's imaging surface is aligned with the robot's center of rotation. Efforts were made to align the camera axes with the robot axes. A target consisting two white rectangles provide a set of eight co-planar feature points. In this paper, the Wiimote measures only angular velocity and thus can be mounted anywhere on the robot as long as the axes are aligned and

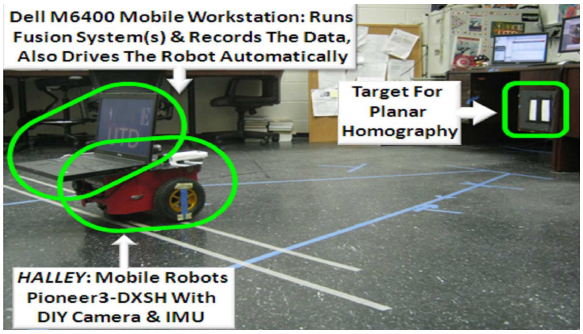


Fig. 4. Experimental Environment

it is tightly strapped down to the surface of the robot.

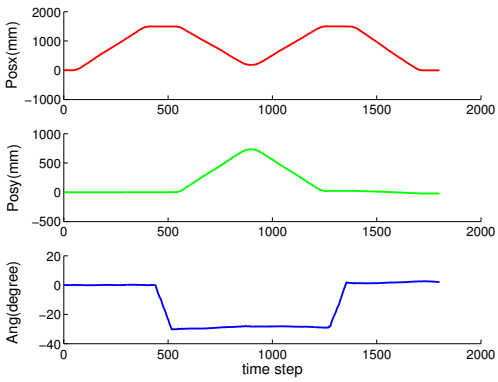
A straight-forward experiment was devised and ran for each system. The robot is placed at the starting point as pictured in Fig. 4. At the start, the robot's wheels are aligned with two marks on the start line. The laser pointers are used to make sure the robot always has the same start position, which is marked on the floor. For the full system, a human operator selects the eight planar feature points that are tracked with the Kanade-Lucas tracker [29].

The remainder of the experiment is fully automated and is the same for both systems. The robot is commanded to move along a V-shape trajectory and then go back along the same path to the starting point. In reality, the robot can not return to the exact starting point due to the drift, as shown in Fig. 5(c). The laser pointers are used to note the stopping point, which provides a ground truth. Before performing the experiments, we first measure the distances from the robot center of rotation(the mid-point between two wheels) to the front laser point (called a) and back laser point (called b) respectively. After each trial, the positions of laser points are recorded as a' and b'. Therefore the new position of robot center can be located on line $a'b'$, and the offset to the starting point is measured. The orientation offset is calculated as

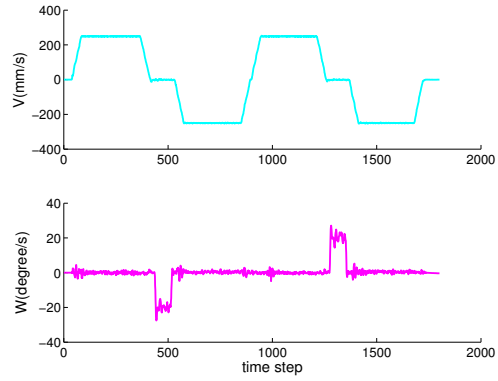
$$\theta = arctan(\frac{y_{a'} - y_{b'}}{x_{a'} - x_{b'}})$$

where $(x_{a'}, y_{a'})$ and $(x_{b'}, y_{b'})$ are the coordinates of a' and b' in the world frame.

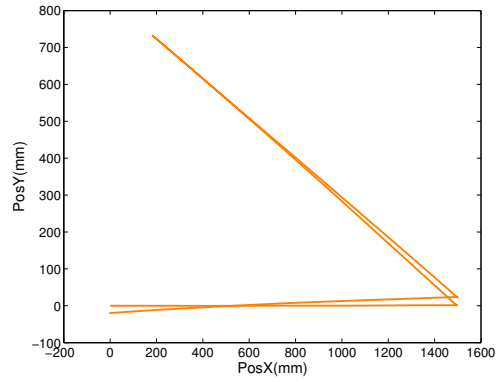
Figs. 5 (a)-(c) show typical estimation results for a run of experiment. Computation time did not allow both estimates to run at the same time. The experiment was repeated 10 times for each system. The offset of the final robot position from the staring point is measured and recorded as ground truth for each trial. Then the estimation error is calculated as the difference between the estimated offset from Kalman filter and the ground truth. The sample mean and sample standard deviation of the error along the world frame x-axis, y-axis, the norm of the pose error, and the world frame orientation θ, are presented for both systems in Table I and graphically in Fig. 6. It is clear that both the mean and variance of full system are significantly lower than that of the reduced system for all measurements. The results show that the full system with visual odometry effectively improves the accuracy of the pose estimate when a planar scene is kept in the field of view.



(a)



(b)



(c)

Fig. 5. Typical estimation results of one trial: (a) pose V.S. time; (b)velocity V.S. time; (c)position on y-axis V.S. position on x-axis

TABLE I

PERFORMANCE OF FULL SYSTEM VS. REDUCED SYSTEM

	Full System		Reduced System	
Axis	$\mu(E)$	$\sigma(E)$	$\mu(E)$	$\sigma(E)$
$\mathcal{F}_w^X$	-5.8271	12.9903	9.7607	15.7477
$\mathcal{F}_w^Y$	-7.3093	28.9184	-62.4058	53.0139
$\|\mathbb{R}^2\|$	9.3478	31.7021	63.1645	55.3034
$\mathcal{F}_w^\theta$	1.0829	1.4260	4.9971	3.4915

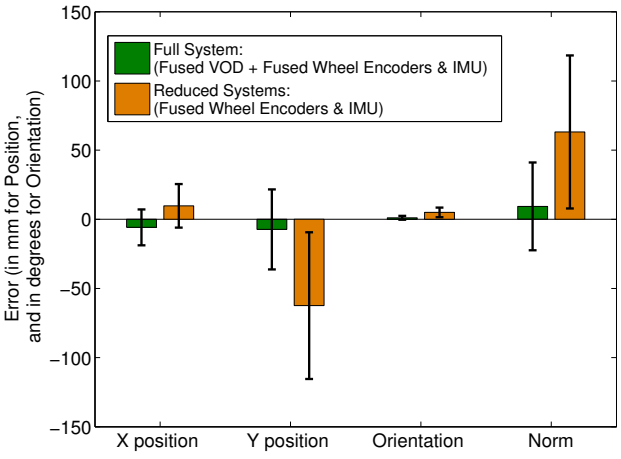


Fig. 6. Summary of Experimental Data (10 trials of each system)

V. CONCLUSION

This paper presents a novel sensor fusion approach for localization of a mobile robotic system. Continuous and discrete forms of the Euclidean Homography Matrix are used to recover pose and velocity information from camera images. Wheel encoders and a low-cost IMU (a Wii Remote controller), measure the linear and angular velocity of the robot. A Kalman filter is used to fuse the velocity and pose estimates from different sensors. Experiments were performed to explore the performance of the proposed method. It can be seen from experimental results that neither the wheel encoders nor the Wiimote by themselves can be used alone to accurately localize a robot. Fusion of wheel odometry with an IMU does give a somewhat accurate estimate of the actual robot motion. Fusing this type of system with the proposed visual odometry method further improves the accuracy of the system with respect to localization.

There are several avenues open for future work. Methods such as virtual parallax and/or the continuous and discrete Essential matrix will be used to perform visual odometry with non-coplanar feature points. A method must be determined for handling large scale movements of the robot and camera, such that feature points can be allowed to enter and leave the camera field of view without disrupting the estimation, such as in [30]. Finally, extensive experiments will be performed where the robot will navigate through large-scale environments.

REFERENCES

[1] G. Dudek and M. Jenkin, "Inertial sensors, gps, and odometry," in *Springer Handbook of Robotics*. Springer, 2008, pp. 477–490.

[2] L. Kleeman and R. Kuc, "Sonar sensing," in *Springer Handbook of Robotics*. Springer, 2008, pp. 491–519.

[3] R. B. Fisher and K. Konolige, "Range sensors," in *Springer Handbook of Robotics*. Springer, 2008, pp. 521–542.

[4] D. Nister, O. Naroditsky, and J. Bergen, "Visual odometry," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 1, 27 2004, pp. I–652 – I–659 Vol.1.

[5] D. DeMenthon and L. S. Davis, "Model-based object pose in 25 lines code," in *European Conf. on Computer Vision*, 1992, pp. 335–343.

[6] L. Quan and Z.-D. Lan, "Linear n-point camera pose determination," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 8, pp. 774–780, 1999.

[7] O. D. Faugeras and F. Lustman, "Motion and structure from motion in a piecewise planar environment," *Int. J. Pattern Recog. and Artificial Intell.*, vol. 2, no. 3, pp. 485–508, 1988.

[8] Y. Ma, S. Soatto, J. Koseck, and S. Sastry, *An Invitation to 3-D Vision*. Springer, 2004.

[9] E. Malis, F. Chaumette, and S. Boudet, "2-1/2D visual servoing," *IEEE Trans. Robot. Autom.*, vol. 15, no. 2, pp. 238–250, 1999.

[10] C. Taylor and J. Ostrowski, "Robust vision-based pose control," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2000, pp. 2734–2740.

[11] Y. Fang, D. Dawson, W. Dixon, and M. de Queiroz, "Homography-based visual servoing wheeled mobile robots," in *Proc. IEEE Conf. on Decision and Control*, 2002, pp. 2866–2871.

[12] Y. Fang, W. E. Dixon, D. M. Dawson, and P. Chawda, "Homography-based visual servo regulation of mobile robots," *IEEE Trans. Syst., Man, Cybern.*, vol. 35, no. 5, pp. 1041–1050, 2005.

[13] Z. Zhang and A. Hanson, "3D reconstruction based on homography mapping," in *Proc. ARPA Image Understanding Workshop Palm Springs CA*, 1996.

[14] N. Daucher, M. Dhome, J. Laprest, and G. Rives, "Speed command a robotic system by monocular pose estimate," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 1997, pp. 55–43.

[15] C. Baillard and A. Zisserman, "Automatic reconstruction planar models from multiple views," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1999, pp. 559–565.

[16] E. Malis and F. Chaumette, "2 1/2D visual servoing with respect to unknown objects through a new estimation scheme camera displacement," *Int. J. Computer Vision*, vol. 37, no. 1, pp. 79–97, 2000.

[17] K. Okada, S. Kagami, M. Inaba, and H. Inoue, "Plane segment finder: algorithm, implementation and applications," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2001, pp. 2120–2125.

[18] V. Chitrakaran, D. M. Dawson, W. E. Dixon, and J. Chen, "Identification a moving object´s velocity with a fixed camera," *Automatica*, vol. 41, no. 3, pp. 553–562, 2005.

[19] D. Tick, J. Shen, and N. Gans, "Fusion of discrete and continuous epipolar geometry for visual odometry and localization," in *Proc. IEEE International Workshop on Robotic and Sensors Environments*, Oct. 2010, (accepted, to appear).

[20] R. G. Brown, *Introduction to Random Signal Analysis and Kalman Filtering*. John Wiley & Sons, 1983.

[21] S. Soatto, R. Frezza, and P. Perona, "Motion estimation via dynamic vision," *Automatic Control, IEEE Transactions on*, vol. 41, no. 3, pp. 393 –413, mar. 1996.

[22] A. Azarbayejani and A. P. Pentland, "Recursive estimation motion, structure, and focal length," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 6, pp. 562–575, 1995.

[23] A. Chiuso, P. Favaro, H. Jin, and S. Soatto, "Structure from motion causally integrated over time," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 4, pp. 523 –535, apr. 2002.

[24] Z. Zhang and O. Faugeras, "A 3d world model builder with a mobile robot," *Int. J. Rob. Res.*, vol. 11, no. 4, pp. 269–285, 1992.

[25] K. Arras, N. Tomatis, and R. Siegwart, "Multisensor on-the-fly localization using laser and vision," in *2000 IEEE/RSJ International Conference*, vol. 1, 2000, pp. 462 –467 vol.1.

[26] J.-S. Gutmann, "Markov-kalman localization for mobile robots," in *16th International Conference on Pattern Recognition*, vol. 2, 2002, pp. 601 – 604 vol.2.

[27] J. Laumond and J. Risler, "Nonholonomic systems: controllability and complexity," *Theoretical Computer Science*, vol. 157, pp. 101–114,, 1996.

[28] G. Hu, N. Gans, and W. E. Dixon, *Complexity and Nonlinearity in Autonomous Robotics, Encyclopedia of Complexity and System Science*. Springer, 2009, vol. 1, ch. Adaptive Visual Servo Control, pp. 42–63.

[29] C. Tomasi and T. Kanade, "Detection and tracking of point features," Carnegie Mellon University, Tech. Rep., 1991.

[30] M. K. Kaiser, N. Gans, and W. Dixon, "Vision-based estimation and control of an aerial vehicle through chained homography," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 46, no. 3, pp. 1064–1077, 2010.