Visual Navigation aid for the blind in dynamic environments

Tung-Sing Leung, Gérard Medioni
Computer Science Department
University of Southern California
Los Angeles, CA 90089
{*tungsinl, medioni*}*@usc.edu*

We describe a robust method to estimate egomotion in highly dynamic environments. Our application is a head mounted stereo system designed to help the visually impaired navigate. Instead of computing egomotion from 3D point correspondences in consecutive frames, we propose to find the ground plane, then decompose the 6DoF egomotion into the motion of the ground plane, and a planar motion on the ground plane. The ground plane is estimated at each frame by analysis of the disparity array. Next, we estimate the normal to the ground plane. This is done either from the visual data, or from the IMU reading. We evaluate the results on both synthetic and real scenes, and compare the results of the direct, 6 DoF estimate with our plane-based approach, with and without the IMU. We conclude that the egomotion estimation using this new approach produces significantly better results, both in simulation and on real data sets.

***Keywords*-visually impaired; visual odometry; dynamic environment;**

I. Introduction

The term "visual impairment" describes any kind of vision loss which can't be cured by standard glasses or medical treatment. According to the statistics from World Health Organization (WHO) [26], there are 285 million people currently suffering from visually impaired worldwide. Visual impairment leads to loss of independence in performing several routine and life-enabling tasks. For instance, indoor and outdoor mobility continues to be a major challenge for the visually impaired such as detecting and avoidance of obstacle. With advances technology, Electronic Travel Aids (ETA) [8] are sophisticated electronic displacement aids designed to improve mobility for the visually impaired. For instance, GPS-based ETA solution was proposed such as the latest *blind map* [34]. However, GPS reception may be unreliable or not available in the presence of trees or large buildings. Therefore, vision-based ETA solutions are proposed to bridge over GPS outages with different imaging sensors, such as monocular [25], RGBD [22] and stereo cameras [27], [28]. Some of them show promising results but are mostly restricted to static environments.

Here, we present a real-time visual odometry algorithm using wearable stereo cameras as shown in Figure 1a to facilitate human navigation for the visually impaired even in *dynamic* outdoor environments such as busy urban area.

(a) (b)

Figure 1: (a) Wearable stereo camera (b) Crowded scenes

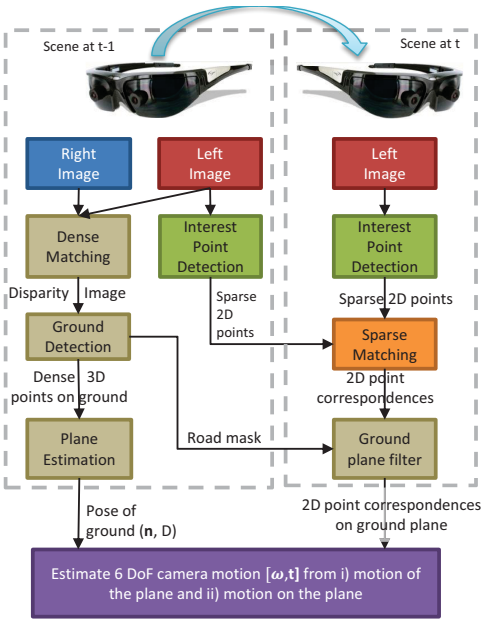


Figure 2: Block diagram of our approach

Many stereo-based visual odometry methods have been proposed and benchmarked with the automobile dataset in KITT Vision Benchmark Suite [11]. However egomotion es-

timation from a walking user (first person vision) is radically different from that of a moving vehicle. The motion of the head mounted camera on a walking person is significantly more complex than that of a car-mounted camera: the former is a combination of 6-Dof head motion and body motion, the latter is dominated only by the forward velocity and yaw angle. In addition, visually impaired applications require a higher rate of visual odometry update. Hence the dead-reckoning error accumulated over time may grow faster than the moving speed of the camera. Furthermore wearable camera has trade-off for small form factor from the image quality and the length of stereo baseline.

In order to overcome the restriction of the static environment assumption made by standard approaches, we propose to detect the ground plane first, then compute the motion of the ground plane followed by solving the motion on the plane as illustrated in Figure 2. This approach is very different from standard methods which make use of all 3D points from the scene. The final output is the relative camera motion (or egomotion) in six-degrees-of-freedom (6DoF) from the previous to current frame which can be used by other high level navigation tasks such as obstacle avoidance, path planning, visual SLAM [1] and self-localization frameworks [5].

Our contributions are:

- an effective pipeline to compute real-time egomotion in cluttered dynamic environments as shown in Figure 1b. This is accomplished by decomposing 6 DoF egomotion using the ground plane.
- we present two robust methods to estimate the ground plane.
- we perform our experiments based on real datasets collected in different areas.

The structure of the rest of this paper is as follows. A summary of the related works is given in Section II. Section III gives an overview of our method. The issue and our method in ground plane detection are discussed in Section IV-A. The two different methods in plane normal estimation are presented in Section IV-B and IV-C. The egomotion algorithm is described in Section V. Experiment results are demonstrated in Section VI. Finally, the conclusion of the study is summarized in Section VII.

II. Related work

Many stereo-based visual odometry works have been published recently. The open-source Libviso2 [12] computes visual odometry of the vehicle by minimizing the sum of reprojection errors over four views obtained from two consecutive stereo image pairs. To handle dynamic scene, the wheeled robot in [30] computes egomotion only from the points sampled on ground region. All these methods rely on RANSAC [10] to handle outliers induced by object movements and they will work as long as outliers are a minority of observations. Beside detect-and-reject, the position and velocity of moving objects can be estimated by Extended Kalman filters (EKF) [2]. Some projects make use of the prior knowledge of the environment. The method in [3] recovers structure-from-motion from two views of a piecewise planar scene. [24] computes relative camera motion with weak Manhattan world assumption and IMU measurements.

Instead of using stereo, PTAM [19] is a monocular algorithm which estimates camera pose from localization. While PTAM produces good egomotion in small-scale environment, we found that the system cannot detect good keyframes robustly and fails to expand the map continuously in larger outdoor environments.

Recently some visual odometry works have been proposed to make use of the ground surface. The system in [18] uses ground plane to resolve the scale of the visual odometry by using prior knowledge of fixed camera height and depression angle. The flying robot in [20] uses the geometry of the ground plane to speed up bundle adjustment. Some systems [6], [7], [23], [17] exploit the ground plane constraint in visual odometry estimation but they are all designed for automobile platforms.

In our application, we use wearable stereo cameras with a baseline as short as 6cm while the others use longer baseline ranging from 12cm to 70cm. In addition, we captured our datasets by walking through different environments similar to [2], so there is no wheel odometry available and our system must be robust to any motion-blur caused by body motion, whereas data acquired by wheeled robots is relatively stable.

As there are many ongoing research works on visual navigation for the blind [22], [28], none of them have been tested in crowded urban areas. Experimental results show that our approach is significantly better in computing continuous visual odometry in dynamic environments, compared to the standard approach.

Although our work is similar to [30] which suggests to estimate visual odometry of the wheeled robot by tracking feature points on the ground, our proposed algorithm is designed to work with smaller stereo camera worn at the eye level for adults. The slant distance between our stereo camera and the ground plane is longer. Hence the texture of the ground surface become less prominent and the ground feature points are more likely to be occluded by pedestrians in crowded environments.

III. Overview of the method

The conventional direct stereo odometry methods first compute the 3D coordinates of the feature point by means of standard triangulation equations. Camera motion is computed by minimizing the sum of reprojection errors of 3D points found in entire scene. Outliers are detected by RANSAC during optimization. However, inconsistent motion vectors (or outliers) are difficult to detect and reject

from dynamic environments that are rich in moving objects. Hence existing visual odometry algorithms fail to produce reliable camera estimation for the rest of the navigation modules in the pipeline such as obstacle avoidance and planning.

To overcome this, we make use of the global motion field property of the ground planar surface and decompose the 6 DoF egomotion into 1) motion of the ground plane and 2) motion on the plane. Figure 2 illustrates the key components of this approach. The pose of the ground plane must be inferred in order to determine a unique camera pose from the optical flow of coplanar 3D points. To accomplish this, ground regions are detected in every frame from the disparity image. We evaluate two approaches to estimate the normal of the ground plane: one from the stereo measurements and the other from the IMU (Inertial measurement unit) measurements. We then perform robust estimation of the egomotion as a composition of these two motions.

IV. Ground plane model estimation

In this section, we detect and model the ground plane, Π, with the standard 3D plane equation:

$$\Pi : AX + BY + CZ + D = 0 \tag{1}$$

where $n = [A, B, C]^T$ is the normal unity vector of the plane and D denotes the distance from the camera origin to the plane. We assume the Z axis of the world coordinate system is aligned with the camera optical axis, the X and Y axes are aligned with the image axes x and y, focal length f is known.

A. Ground plane detection for short baseline stereo

The standard V-disparity algorithm [15] works very well in detecting the road regions with stereo camera. We modified the algorithm for our wearable stereo camera to deal with different head movements and crowded scenes. Given a calibrated head mounted stereo camera with depression angle, θ, to the horizontal ground plane and zero roll angle, the work in [15] shows that the disparity, d, of the ground plane surface is linearly related to the y-coordinate on the image plane by $hd = b(y - p_y)cos\theta + fsin\theta$ where f is focal length, b is base line, p_y is the y-coordinate of the principle point, h is the positive height of the camera above the ground plane.

We compute a dense disparity image using the Semi-Global Block Matching (SGBM) method [13] and convert it into a V-Disparity image [15] as input to the ground detection algorithm. Image regions corresponding to road surface are located by fitting a straight line in the V-disparity domain using Hough transform [14].

Due to the range limitation of short baseline stereo camera, the width of the V-disparity map is very narrow. Hence the diagonal line corresponding to the horizontal plane may be almost vertical and Hough transform may detect both as one single line.

We made two modifications to the V-Disparity method in [21] to extract the diagonal line. After thresholding the V-Disparity image, we first apply the *thin* and *clean* morphological operations to clean up the V-Disparity domain. Second, instead of detecting the diagonal line, we remove those regions in V-Disparity that do not correspond to horizontal plane. This is done by summing up the binary intensity of each column in V-Disparity domain, the column with highest sum corresponds to the vertical line. All columns of the V-Disparity image on the left of the vertical line are set to zero because nothing should appear at a distance further than that disparity value. We also remove the upper portion of the V-Disparity map spanned by the vertical line. The diagonal line can then be extracted quickly by Hough transform in the left over region. If no vertical line is detected, we proceed to diagonal line extraction with Hough transform over the entire V-Disparity image.

Finally we check the largest connected region in the disparity map that satisfies the extracted line. If the largest region covers at least 10% of the image, it is used as road mask to estimate the ground plane model in the next step. Otherwise, our algorithm assumes the camera moving in constant velocity and predicts the camera motion using Kalman filter which will be described in Section V-A. An example road mask is highlighted in yellow in Figure 3.

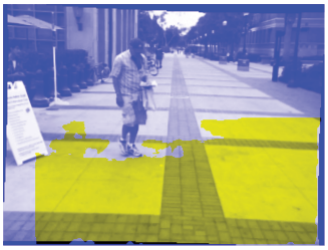


Figure 3: The output road mask is highlighted in yellow

B. Ground plane normal estimation from stereo

We first compute the 3D coordinates of all the points on the ground by means of standard stereo geometry equations. The next task is to estimate $[A, B, C, D]$ in (1) by fitting a linear model that point cloud. Since we have the dense disparity data of the road region, plane fitting is accomplished by weighted least square fitting [31] due to its speed and low computational complexity. Each 3D point is weighted by the sigmoid function $w(d) = (1 + e^{-(d-\alpha)})^{-1}$ of the disparity d with the tuning parameter α=20 and more weight, w, is assigned to points nearer to the camera.

C. Ground plane normal estimation from IMU

Alternatively, the plane normal $\boldsymbol{n} = [A, B, C]^T$ can be approximated by projecting the *up* vector of the IMU, $\boldsymbol{n}_{up} =$

$[0, -1, 0]^T$, using $\boldsymbol{n} = \mathbf{R}_{imu}\boldsymbol{n}_{up}$ where $\mathbf{R}_{imu} \in SO(3)$ is the rotation matrix form by pitch and roll angle measurements.

V. Egomotion estimation

A. Classical egomotion estimation directly from all points

If the camera is moving in space with translational vector $\boldsymbol{t} = [t_x, t_y, t_z]^T$ and angular velocity $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^T$, standard egomotion algorithm decomposes the 6 DoF camera motion into $F_{c_{k-1}}^{c_k} = F_{I_k}^{c_k} F_{I_{k-1}}^{I_k} F_{c_{k-1}}^{I_{k-1}}$ where $F_{c_{k-1}}^{I_{k-1}}$ is the camera to image plane transformation at previous frame $k-1$, $F_{I_{k-1}}^{I_k}$ is the transformation from previous frame to current frame which is directly measured by optical flow, and $F_{I_k}^{c_k}$ is the image plane to camera transformation at current frame.

By assuming pin-hole camera model with zero skew, the transformations $(F_{c_{k-1}}^{I_{k-1}}, F_{I_k}^{c_k})$ relate 3D points $\boldsymbol{P} = [X, Y, Z]^T$ to image coordinates $\boldsymbol{p} = [u, v, 1]^T$ by $\boldsymbol{p} = f\frac{\boldsymbol{P}}{Z}$ where f is focal length. Note that these two matrices are not global transformations in the presence of non-rigid structure because Z depends on disparity. Therefore in case of dynamic scene, standard visual odometry may converge to the wrong solution of $F_{c_{k-1}}^{c_k}$ when minimizing the sum of reprojection errors, E, over all 3D points in (2) [1].

$$E = \arg\min_{\{\boldsymbol{\omega},\boldsymbol{t}\}} \sum_{i=1}^{N} \|x_i - \rho(\boldsymbol{X}_i; \boldsymbol{\omega}, \boldsymbol{t})\|^2 \tag{2}$$

Here x_i denotes the feature locations in the current left images. $\rho(\boldsymbol{X}_i; \boldsymbol{\omega}, \boldsymbol{t})$ computes the homogeneous pixel coordinates $\boldsymbol{u}_i$ of 3D point $\boldsymbol{X}_i = [X, Y, Z]_i^T$ in the left image plane using $\boldsymbol{u}_i = \boldsymbol{K}(\boldsymbol{R}\boldsymbol{X}_i + \boldsymbol{t})$ with rotation matrix $\boldsymbol{R} \in SO(3)$ is formed by $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^T$ and translation vector $\boldsymbol{t} = [t_x, t_y, t_z]^T$, $\boldsymbol{K}$ denotes the projection matrix.

To handle dynamic environment, the visual odometry [30] solves the cost function (2) by sampling optical flow only on the ground instead of entire scene. In other words, 6 DoF camera motion is decomposed into $F_{c_{k-1}}^{c_k} = F_{g_k}^{c_k} F_{g_{k-1}}^{g_k} F_{c_{k-1}}^{g_{k-1}}$ where $(F_{g_k}^{c_k}, F_{c_{k-1}}^{g_{k-1}})$ are the transformations between the camera and image regions corresponding to the ground at time k and $k-1$ respectively, $F_{g_{k-1}}^{g_k}$ is measured by optical flow found in ground regions. $(F_{g_k}^{c_k}, F_{c_{k-1}}^{g_{k-1}})$ are global transformations because the ground is a rigid structure. However computing 3D motion directly from motion estimates produced by coplanar points may lead to two solutions. This is because same planar motion field can be induced by two different planes undergoing two different 3D motions [33]. We can determine unique solution by using the ground plane structure (computed in either Section IV-B or IV-C) as discussed below.

[1]Libviso2 improves the accuracy by minimizing the cost function (2) over four views obtained from two consecutive stereo image pairs.

B. Egomotion estimation from ground plane

We assume the camera is moving in space with translational vector $\boldsymbol{t}$ and angular velocity $\boldsymbol{\omega}$ while observing a planar surface, Π in (1). Let $\boldsymbol{P}$ be a point on Π, the equation of Π is $\boldsymbol{n}^T\boldsymbol{P} = D$, where $\boldsymbol{n} = [n_x, n_y, n_z]^T$ is the unit vector normal to Π, and D is the distance between Π and the center of projection. We can further decompose $(F_{g_k}^{c_k}, F_{c_{k-1}}^{g_{k-1}})$ into $(F_{p_k}^{c_k} F_{g_k}^{p_k}, F_{p_{k-1}}^{g_{k-1}} F_{c_{k-1}}^{p_{k-1}})$. Hence the 6 DoF egomotion becomes $F_{c_{k-1}}^{c_k} = F_{p_k}^{c_k} F_{g_k}^{p_k} F_{g_{k-1}}^{g_k} F_{p_{k-1}}^{g_{k-1}} F_{c_{k-1}}^{p_{k-1}}$ where $F_{p_k}^{c_k}$ and $F_{c_{k-1}}^{p_{k-1}}$ are transformation between the camera and ground plane, which is a function of $(\boldsymbol{n}, D)$.

Given the parameters of the ground plane, $(F_{g_k}^{c_k}, F_{c_{k-1}}^{g_{k-1}})$ become global transformations because the depth Z of the ground pixel (u, v) is a function of global parameters $(\boldsymbol{n}, D)$: $Z = (Df)/(n_x u + n_y v + n_z f)$

By taking the ground plane geometry $(\boldsymbol{n}, D)$, a set of pixels $\boldsymbol{G}$ within the road mask estimated in previous section, a set of tracked ground feature points $x_j = (u, v, 1)_j^T$ for $j \in \boldsymbol{G}$ in current left image, and their corresponding feature points $x'_j = (u', v', 1)_j^T$ and disparity d'_j in previous left image, we estimate the vector $\boldsymbol{\omega}$ and $\boldsymbol{t}$ by minimizing the sum of reprojection errors, E, over all points in the road mask:

$$E = \arg\min_{\{\boldsymbol{\omega},\boldsymbol{t}\}} \sum_{j \in \boldsymbol{G}} \|x_j - \rho(\boldsymbol{X}_j; \boldsymbol{\omega}, \boldsymbol{t})\|^2 \tag{3}$$

where $\boldsymbol{X}_j = [(u' - p_x)b/d' \quad (v' - p_y)b/d' \quad Z]^T$ and b denotes the baseline. Hence the original visual odometry is re-formulated as quadratic curve fitting problem.

We use Kanade Lucas Tomasi (KLT) feature tracker [4] to select and track feature points between two consecutive frames in the left camera. To improve speed and robustness, the KLT tracker only processes the image regions within the road mask. Although we also tested other feature matching algorithms such as the feature matching algorithm in Libviso2 [12] and FAST detector [29], we found that KLT is more accurate in tracking the ground features for our application. In addition blind people do not walk at high speed, so the difference between consecutive frames can be easily handled by the pyramidal implementation of KLT tracker.

We derive the Jacobian matrix from (3) and iteratively minimize it using Gauss-Newton optimization with respect to the transformation parameters $\boldsymbol{\omega}$ and $\boldsymbol{t}$. RANSAC [10] is used during optimization to improve the robustness of motion estimation against outlier motion vectors. The parameters ($\boldsymbol{\omega}$, $\boldsymbol{t}$) are first estimated for N trials based on m randomly selected motion vector on the ground plane. We choose the set parameters $\{\boldsymbol{\omega}_i, \boldsymbol{t}_i\}_{i=1}^N$ from the trials with the largest number of inliers and use all these inliers to compute the final ($\boldsymbol{\omega}$, $\boldsymbol{t}$). N is adjusted during the trials with the standard RANSAC equation: $N = log(1-p)/log(1-u^m)$ where m=3, p=0.99 and u denotes the highest ratio of inliers to the total number of motion vectors up to current trial.

We also include the same Kalman Filter as in [12] to smooth the egomotion output, as well as predict the camera motion when not enough ground region features are detected within the field of view.

VI. Results

A. Sensors

We demonstrate our approach using image sequences captured by Vuzix[2], a off-the-shelf wearable stereo camera with IMU, while walking through different outdoor environments. The stereo camera are mounted on a plastic sunglasses frame, as shown in Figure 4. The stereo rig is not only lightweight and comfortable to wear but also gives a natural appearance to the visually impaired user when navigating outdoor. The downside is that it is made up of two low quality image sensors with a very short baseline of 6cm (half of Bumblebee2).

Figure 4: The primary sensor of our system: Vuzix Wrap 920 AR with integrated IMU [9]

B. Implementation

We have implemented our algorithm in C++ and tested it in desktop with quad i7 CPU core running at 3.4 GHz. The resolution of input images is 320x240 pixels and the average number of feature points being tracked is about 370.

C. Methods

For the rest of the experiments, we run our visual odometry algorithm with and without IMU and compare the output trajectories with Libviso2 and Libfovis[16]. Libfovis is another open source stereo-based visual odometry developed for unmanned aerial vehicle.

When the IMU is not used, the ground plane normal is measured by fitting plane to the 3D points found on the ground (Section IV-B). When the IMU is used, the ground plane normal is approximated from the roll and pitch measurements using Section IV-C. Note that none of the algorithms uses any form of bundle adjustment nor loop-closure technique. Therefore, like all dead-reckoning systems, our system output degrades with elapsed time and distance travelled in all experiments. The accuracy can be easily improved by incorporating other optimization frameworks such as [32] or [27].

[2]http://www.vuzix.com/UKSITE/ar/products_wrap920ar.html/

D. Data

Although there are common benchmarking datasets [11] available for visual odometry, the data was captured by stereo camera mounted on automobiles. Since the motion of the head mounted camera on a walking person is significantly different from the car-mounted camera. We argue that meaningful comparison is to run all visual odometry algorithms on our dataset, rather than running our software on standard dataset. We will be happy to make the data publicly available.

Our datasets consist of stereo image pairs captured at 30fps as well as IMU data which was logged at 100Hz. GPS measurements are not accurate enough for ground truth because our trajectories are surrounded by buildings and trees. Therefore we extracted the ground truth trajectories using Google Earth software. This is done manually by comparing the ground texture found in the recorded images with the Google satellite map. We select *anchored* frames, which are the left camera images having some distinctive ground texture pattern visible, at regular interval along the path we walked. We approximate the spots where the frames were captured from Google Earth and use the corresponding UTM coordinates (WGS84) as the ground truth camera location for those frames. The height of all ground truth are set to zero. The locations of anchored frames are shown as red dots in three satellite maps. Rotational error cannot be evaluated quantitatively as accurate orientation measurements are not available. However the rotational error can be evaluated qualitatively by comparing the shape of the output trajectory with the ground truth path.

E. Evaluation Criteria

Since bundle adjustment is not used by any of the algorithms in evaluation, we use a similar criteria described in [11] to evaluate the accuracy of the estimated trajectory.

We compute the translational error between all possible pairs of anchored frames and take the average of them. Formally, given anchored frames $1...M$, the translational error, $T_{err}(i,j)$, between two anchored frames i and j is defined as $T_{err}(i,j) = \|(p_j - p_i) - (\hat{p}_j - \hat{p}_i)\|_2$ where $\hat{p}, p \in \mathbb{R}^3$ are the estimated camera location and the corresponding ground truth location lookup from Google Earth respectively. We normalized each translational error with the length of the corresponding subpath. The average translational error for entire trajectory with M anchor frames is defined as

$$E_{ave} = \frac{2}{M(M-1)} \sum_{i=1}^{M-1} \sum_{j=i+1}^{M} \frac{T_{err}(i,j)}{\|p_j - p_i\|_2} \qquad (4)$$

We assume the location of the first frame is known. The first heading is obtained from compass but subsequence camera orientations are computed by the visual odometry algorithms.

Algorithm/ Dataset	LibFovis	Libviso2	Ours (with IMU)	Ours (no IMU)
Cafeteria	66.31%	52.27%	**10.38%**	51.64%
Football	72.03%	68.69%	**15.47%**	29.60%
Shopping	64.71%	71.92%	**18.91%**	30.42%
Average E_{ave}	67.68%	64.29%	**14.92%**	37.22%

Table I: Average translational error (E_{ave}) for different algorithms in different environments

The quantitative results are summarized in Table I. Our proposed algorithm gives the lowest error in all three datasets when IMU is used to estimate the ground plane. It also gives better results with stereo camera alone comparing with other state-of-the-art methods. Although Libviso2 works great with the KITTI Vision Benchmark Suite, it suffers from excessive accumulated odometry error when processing at our walking data at 30 frames per sec (fps). It gives more accurate results when we subsampled our dataset to 1 fps but this frame rate is too slow for our visual impaired application. Our system alleviates the accumulated error by improving the accuracy of depth measurement using (i) KLT tracking, (ii) dense stereo disparity and (iii) depth correction using the fitted ground plane model. The details of individual dataset are given below. Video of experiment results is avilable at http://youtu.be/0gcasjQpAcY.

F. Cafeteria Area

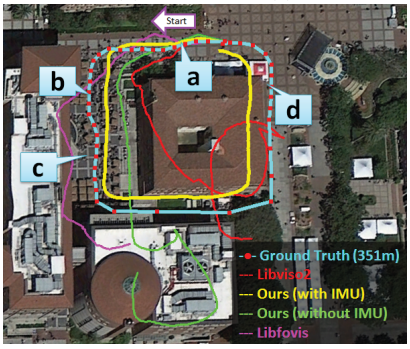
The first dataset was taken around a crowded cafeteria area as indicated by the ground truth path in light blue in Figure 5e. The route is about 218 meters. The environment was crowded and there were people walking around in front of our stereo camera except the last 50 meters of the route but the images were overexposed as shown in Figure 5d.

Some snapshots along the route are shown in the thumbnails in Figure 5. The yellow curve in Figure 5e shows the trajectory generated by our algorithm when IMU is used to approximate ground plane normal. This trajectory is closest to the ground truth comparing to Libviso2 and Libfovis. The green curve is the output of our algorithm without IMU and the ground plane was estimated using RANSAC 3D plane fitting. The output trajectory is relatively straight between corners until it drifted after the second turn (about 20m after label C in Figure 5e) before entering a shaded corridor. Figure 5f shows the average translational error of the four algorithms for different subpath lengths. Our algorithm with IMU (red curve) clearly gives lowest translational error with different path length. However when IMU is not in used, our algorithm only performed slightly better than Libviso2 on average as shown in the first row of Table I.

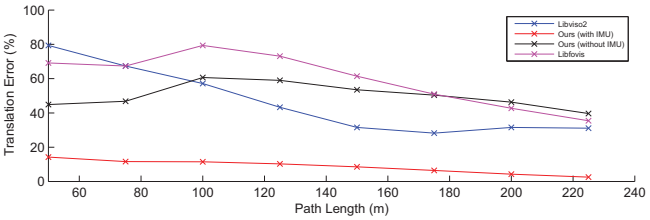
G. Football Match

The second dataset is the most challenging one, as it was collected right after a football game. The route is about 351 meters. As shown in the thumbnails in Figure 6, the environment was crowded with spectators leaving the

(a) (b) (c) (d)



(e) Ground truth and estimated trajectories



(f) Average Translational Error

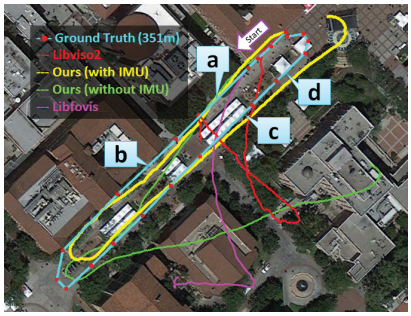
Figure 5: Cafeteria area test result

stadium. Our camera moved in the direction opposite the crowd in the first half of the route, and then followed the crowd at the second half. The yellow curve in 6e clearly shows that our visual odometry is the most insensitive to the dynamic change in the environment when IMU is used. When IMU is not used, the accuracy of the green trajectory is degraded but does not drift as badly as the other two algorithms. Although the green curve bends away from the ground truth after the first U-turn due to accumulated rotational error, the trajectory remain relatively straight until the second U-turn. Our proposed algorithm maintain the lowest translational error throughout the route as shown in 6f even though they gives large end-point error shown in the satellite map.

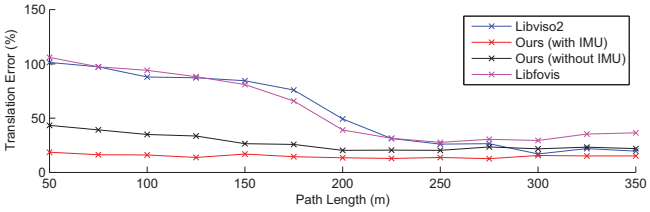
H. Shopping Area

Figure 7 shows the dataset that we collected in a shopping area. The route is about 268 meters. This dataset is less crowded but the road is surrounded by buildings as shown in the thumbnails and satellite image in Figure 7e. Due to the sunlight was blocked by some buildings, parts of the image sequence are overexposed (Figure 7c) and some are underexposed (Figure 7d). The shape of the two trajectories computed by our proposed algorithm with and without IMU come the closest to the ground truth as shown in the yellow and green curves in 7e. Figure 7f shows that our algorithm gives the lowest average translation error with and without

(a) (b) (c) (d)



(e) Ground truth and estimated trajectories



(f) Average translational Error

Figure 6: Football match dataset

IMU in ground plane estimation. Note that the translation error of Libviso2 exceeds 100%. This indicates that the navigation errors accumulate much faster than the moving speed of the camera.

VII. Conclusion

We have presented a new robust visual odometry method that works with head-mounted wearable stereo camera. In order to handle environments that are rich in moving objects, our method computes egomotion only from the optical flow observed on the ground plane. We described our robust ground detection method for short baseline stereo. We also presented two different approaches to estimate ground plane normal. Experimental results show that our system outperforms existing visual odometry which relies on motion field from the entire scene. Furthermore, by comparing the results with and without IMU, estimating the ground plane normal with IMU clearly improve the egomotion estimation. Thanks to these encouraging results, we are performing experiments with real patients at the Braille Institute's Sight Center in Los Angeles.

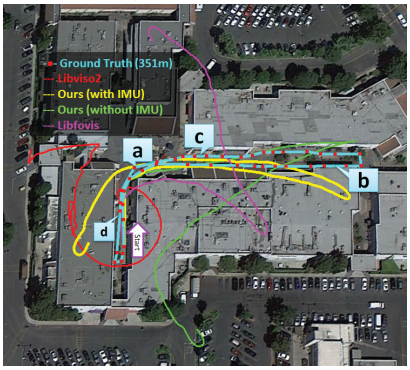
Acknowledgment

The authors would like to thank... more thanks here

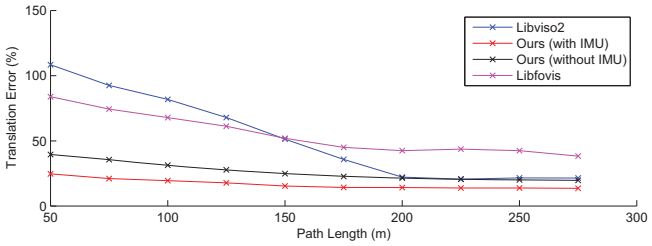
References

[1] P. F. Alcantarilla, L. M. Bergasa, and F. Dellaert. Visual odometry priors for robust ekf-slam. In *ICRA*, pages 3501–3506. IEEE, 2010.

(a) (b) (c) (d)



(e) Ground truth and estimated trajectories



(f) Average translational Error

Figure 7: Shopping area dataset

[2] H. Badino and T. Kanade. A head-wearable short-baseline stereo system for the simultaneous estimation of structure and motion. In *IAPR Conference on Machine Vision Applications (MVA)*, number CMU-RI-TR-, June 2011.

[3] A. Bartoli, P. Sturm, and R. Haraud. Projective structure and motion from two views of a piecewise planar scene. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 593–598 vol.1, 2001.

[4] J.-Y. Bouguet. Pyramidal implementation of the lucas kanade feature tracker. *Intel Corporation, Microprocessor Research Labs*, 2000.

[5] M. A. Brubaker, A. Geiger, and R. Urtasun. Lost! leveraging the crowd for probabilistic visual self-localization. In *CVPR*, pages 3057–3064. IEEE, 2013.

[6] V. Caglioti and S. Gasparini. Uncalibrated visual odometry for ground plane motion without auto-calibration. In *VISAPP (Workshop on on Robot Vision)*, pages 107–116, 2007.

[7] V. Caglioti and P. Taddei. Planar motion estimation using an uncalibrated general camera.

[8] A. M. Cook and S. Hussey. *Assistive Technologies: Principles and Practice (2nd Edition)*. Mosby, 2 edition, Dec. 2001.

[9] V. Corporation. Wrap 920ar augmented reality eyewear user guide, 2011.

[10] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981.

[11] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Computer Vision and Pattern Recognition (CVPR), 2012*

IEEE Conference on, pages 3354–3361, 2012.

[12] A. Geiger, J. Ziegler, and C. Stiller. Stereoscan: Dense 3d reconstruction in real-time. In *IEEE Intelligent Vehicles Symposium*, Baden-Baden, Germany, June 2011.

[13] H. Hirschmuller. Accurate and efficient stereo processing by semi-global matching and mutual information. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 - Volume 02*, CVPR '05, pages 807–814, Washington, DC, USA, 2005. IEEE Computer Society.

[14] P. V. C. Hough. Machine Analysis of Bubble Chamber Pictures. In *International Conference on High Energy Accelerators and Instrumentation*, CERN, 1959.

[15] Z. Hu and K. Uchimura. U-v-disparity: an efficient algorithm for stereovision based scene analysis. In *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*, pages 48 – 54, june 2005.

[16] A. S. Huang, A. Bachrach, P. Henry, M. Krainin, D. Maturana, D. Fox, and N. Roy. Visual odometry and mapping for autonomous flight using an rgb-d camera. In *Int. Symposium on Robotics Research (ISRR)*, Flagstaff, Arizona, USA, Aug. 2011.

[17] Q. Ke and T. Kanade. Transforming camera geometry to a virtual downward-looking camera: Robust ego-motion estimation and ground-layer detection. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2003)*, June 2003.

[18] B. M. Kitt, J. Rehder, A. D. Chambers, M. Schonbein, H. Lategahn, and S. Singh. Monocular visual odometry using a planar road model to solve scale ambiguity. In *Proc. European Conference on Mobile Robots*, September 2011.

[19] G. Klein and D. Murray. Parallel tracking and mapping for small AR workspaces. In *Proc. Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'07)*, Nara, Japan, November 2007.

[20] G. H. Lee, F. Fraundorfer, and M. Pollefeys. Mav visual slam with plane constraint. In *ICRA*, pages 3139–3144. IEEE, 2011.

[21] Y. H. Lee, T.-S. Leung, and G. Medioni. Real-time staircase detection from a wearable stereo system. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 3770 –3773, nov. 2012.

[22] Y. H. Lee and G. Medioni. A rgb-d camera based navigation for the visually impaired. In *RSS 2011 RGB-D: Advanced Reasoning with Depth Camera Workshop*, Los Angeles, USA, June 2011.

[23] B. Liang and N. Pears. Visual navigation using planar homographies. In *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, volume 1, pages 205–210 vol.1, 2002.

[24] M. P. Olivier Saurer, Friedrich Fraundorfer. Homography based visual odometry with known vertical direction and weak manhattan world assumption. In *Vicomor Workshop at IROS 2012*, 2012.

[25] C. Olson and A. Robinson. Camera-aided human navigation: Advances and challenges. In *Emerging Signal Processing Applications (ESPA), 2012 IEEE International Conference on*, pages 75 –78, jan. 2012.

[26] W. H. Organization. Visual impairment and blindness fact sheet, 2012.

[27] V. Pradeep, G. Medioni, and J. Weiland. Visual loop closing using multi-resolution sift grids in metric-topological slam. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, pages 1438–1445. IEEE, 2009.

[28] V. Pradeep, G. Medioni, and J. Weiland. Robot vision for the visually impaired. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 15–22, June.

[29] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. In *European Conference on Computer Vision*, volume 1, pages 430–443, May 2006.

[30] Y. Tamura, M. Suzuki, A. Ishii, and Y. Kuroda. Visual odometry with effective feature sampling for untextured outdoor environment. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 3492–3497, Oct.

[31] G. Tools. Least squares fitting of data, 2010.

[32] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle adjustment - a modern synthesis. In *Proceedings of the International Workshop on Vision Algorithms: Theory and Practice*, ICCV '99, pages 298–372, 2000.

[33] E. Trucco and A. Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1998.

[34] J. Yong. 'google maps' for the blind, 2013.