Correspondenceless Ego-Motion Estimation Using an IMU*

Ameesh Makadia and Kostas Daniilidis
GRASP Laboratory
University of Pennsylvania, Philadelphia, PA 19104
{makadia, kostas}@grasp.cis.upenn.edu

Abstract— **Mobile robots can be easily equipped with numerous sensors which can aid in the tasks of localization and ego-motion estimation. Two such examples are Inertial Measurement Units (IMU), which provide a gravity vector via pitch and roll angular velocities, and wide-angle or panoramic imaging devices which capture $360°$ field-of-view images. As the number of powerful devices on a single robot increases, an important problem arises in how to fuse the information coming from multiple sources to obtain an accurate and efficient motion estimate. The IMU provides real-time readings which can be employed in orientation estimation, while in principle an Omnidirectional camera provides enough information to estimate the full rigid motion (up to translational scale). However, in addition to being computationally overwhelming, such an estimation is traditionally based on the sensitive search for feature correspondences between image frames. In this paper we present a novel algorithm that exploits information from an IMU to reduce the five parameter motion search to a three-parameter estimation. For this task we formulate a generalized Hough transform which processes image features directly to avoid searching for correspondences. The Hough space is computed rapidly by re-treating the transform as a convolution of spherical images.**

Index Terms— **Omnidirectional vision, localization, inertial sensors.**

I. INTRODUCTION

The commoditization of peripheral sensors like GPS units, range finders, wide-angle or panoramic cameras, and Inertial Measurement Units (IMU) has made it fairly effortless to equip mobile robots with a plethora of sensing devices (see Figure 1). This advance has altered the way longstanding problems of robot self-localization and ego-motion estimation are approached. Instead of collecting enough reliable information to perform any of these motion-based tasks, the problems are now geared towards the fusion of information already available. Since many devices on a robot independently capture enough data to provide partial or even complete solutions to these motion estimation problems, the target is to find the optimal fusion of available sensors in order to reach the objectives of accuracy and efficiency. In this paper we assume that a robot is outfitted with two useful devices: an IMU and a single viewpoint Omnidirectional camera system.

An IMU generally combines three orthogonal accelerometers to deliver the angular velocities of motion. The mea-

surements of the roll and pitch angles can be used to form the gravity vector, which could alternatively be obtained directly from an on-board inclinometer. In terms of motion estimation, the gravity vector reduces our rotational search space down to a single rotation about a known axis.

As opposed to IMUs, cameras generally capture enough information in a pair of images to estimate both rotational and translational components of motion. However, due to the projective nature of image formation, distance information is lost and only the translational direction, but not magnitude, can be recovered. The successful fusion of information from inertial sensors with image information has numerous applications. For example, for augmented reality, accurate registration is required to seamlessly overlay artificial objects on a real view. Azuma et al. [3] created a hybrid inertial-vision tracker where a vision-based algorithm refines orientation estimates provided by an inertial sensor. Recently Burschka and Hager developed a vision approach to SLAM [4]. This approach circumvented the problems caused from drift in the inertial measurements by using a vision algorithm to estimate directly the relative pose of the camera between frames. In addition to these examples, there is also the InerVis Workshop [1] dedicated solely to the integration of inertial and vision sensors.

One drawback to the approaches just mentioned is that the vision component requires some successful feature or target tracking. In [4] the pose is obtained by locating and tracking landmarks, while in [3] tracked image features provide the refinement to the orientation estimate. Sophisticated feature extraction and matching algorithms [11], [8] must often be tailored in different environments to achieve satisfactory results. Notwithstanding an impressive volume of research dedicated towards this problem, there are many scenarios (depending on properties of the imaging sensor, or scenes and objects with repeated textures) for which features cannot be successfully matched. Take for example Omnidirectional camera systems, which have become synonymous with mobile robots. The panoramic view which makes such sensors so appealing is also being represented by relatively fewer pixels (per viewing angle). Combined with the projection geometry of such sensors, this complicates successful feature tracking.

Although we will not rely on finding correspondences between two images, we still must not abandon local image characteristics altogether. Previously, Aloimonos and Hervé [2] showed the rigid motion of planar patches can be

estimated without correspondences using a binocular stereo setup. More directly related to the task of planar ego-motion in arbitrary scenes, Roy and Cox [10] treated this approach by statistically modeling the variance of intensity differences between points relative to their Euclidean distance. This model is used to estimate the likelihood of assumed motions. Geyer [6] proposed a 6D Radon transform on the space of Essential matrices parameterized by ordered pairs in the rotation group $SO(3)$. Whereas both proposed techniques address the general motion problem, we are concerned with the restricted 3-parameter subproblem enabled by the presence of a gravity vector obtained from an onboard IMU.

We treat this motion estimation problem with a generalized Hough transform on the space of three-parameter motions. This transform processes features directly, rendering the search for matching pairs between images unnecessary. By decoupling the search for the one remaining rotational parameter and the translational direction, we reduce the complexity of the Hough computation down to a convolution of two spherical images. This is advantageous because spherical convolution enjoys a marked speedup when performed in the Fourier domain.

The organization of this paper is as follows. In section II we will generalize the Hough transform to the space of three-parameter motions in which we are interested. We will mold this Hough into a spherical convolution in Section III (an introductory exposition on spherical harmonic analysis is offered in the appendix), and we will cover the full motion estimation algorithm in Section III-A. Experiments follow in Section IV and we discuss planned future work along with some concluding remarks in Section V.

II. MOTION ESTIMATION AS HOUGH

Images processed on board mobile robots are increasingly likely to come from wide-angle or panoramic imaging devices such as Omnidirectional camera systems. The projection models for these camera systems are quite different from the traditional planar-perspective model. In fact, for many Omnidirectional systems combining digital cameras with conic mirrors, the natural image modality is often the surface of a parabola or hyperbola rather than a planar surface. For a large class of such single-viewpoint sensors, the projection geometry is equivalent to first a projection of scene points onto a sphere, followed by a stereographic projection to the image plane. This enables us to unify images from many Omnidirectional sensors under the canopy of spherical images. In this setting, the imaging surface takes the shape of a sphere, while the single viewpoint of the camera would lie at the center of this sphere. The spherical perspective projection model maps scene points $P \in \mathbb{R}^3$ to image points $p \in \mathbb{S}^2$, where $p = P/||P||$.

Now consider such a spherical camera observing a scene from two different locations and orientations which are related by the rigid camera transformation $(R, T), R \in SO(3), T \in \mathbb{R}^3$. If P and Q are the 3D coordinates of the
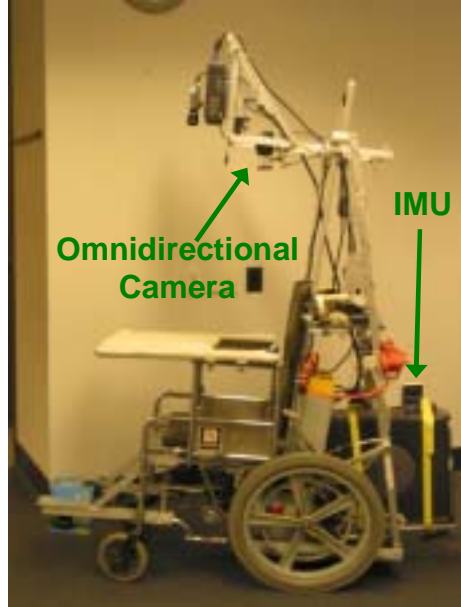

Fig. 1. A computer controlled wheelchair sporting numerous sensing devices, including a laser range finder, IR proximity sensors, encoders for odometry, an IMU, and an Omnidirectional camera.

same world point as viewed from the two camera frames, then the relationship between P and Q can be determined from the relative pose of the cameras: $Q = RP + T$. Since the projection of P, Q to image points p, q will not retain any information about the distances of the points from the camera center, we can only infer that three vectors Rp, q, and T lie on the same plane. This condition gives rise to the traditional two-view epipolar constraint:

$$(Rp \times q)^T T = 0 \qquad (1)$$

Notice that this constraint is independent of the translation vector's scale, so only the direction of translation can be recovered without prior knowledge of the environment. To reflect this we will write T as a unit vector explicitly defined by two rotations: $T = R_z(\gamma)R_y(\beta)\vec{e_3}$. Here $R_z(\gamma)$ is a rotation about the Z-axis by the angle γ, and $\vec{e_3}$ is the unit vector representing the north pole (Z-axis). The epipolar constraint (1) then becomes

$$(Rp \times q)^T (R_z(\gamma)R_y(\beta)\vec{e_3}) = 0 \qquad (2)$$

This motion space is five dimensional (three for orientation and two for direction), but we would like to remove some complexity by integrating information from an onboard IMU. For many off-the-shelf IMUs, the recorded inertial measurements come in the form of angular velocities about the three canonical axes of $\mathbb{R}^3$. Two of these measurements, the angles of Pitch and Roll, can be combined to determine the gravity vector. We would like to fuse this partial orientation information with our image information to reduce the dimensionality of our unknown parameter space. The gravity vector acts as somewhat of a reference vector since it has the effect that we can rectify our camera coordinate

frames so that the rotational component linking the two cameras is just a rotation about a known axis (without loss of generality we will always choose this axis to be the Z-axis). With the assistance of the IMU readings, our unknown 3D rotation is reduced to a one-parameter rotation about the Z-axis $R_z(\alpha)$. The epipolar constraint for this restricted motion is

$$(R_z(\alpha)p \times q)^T (R_z(\gamma)R_y(\beta)\vec{e_3}) = 0 \quad (3)$$

We now continue with a discussion on how to exploit this reduced constraint to obtain an ego-motion solution without having to find image correspondences first.

A. Hough on the space of motions

We shall begin with an introduction to the traditional Hough transform as it applies to identifying lines in planar perspective images before illustrating how similar intuition can be used to identify the correct motion parameters of a spherical camera. Specifically, we will examine the continuous case of the Hough transform which is often identified with the Radon transform. For identifying lines on a planar image the Radon transform is given as

$$G(\rho, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x,y)\delta(\rho - x\cos\theta - y\sin\theta)dxdy$$

Here $g(x,y)$ is the weighting function, in this case an edge image. This could be computed simply as a gradient magnitude image. δ is a soft characteristic function which measures how close the edge pixel (x,y) lies to the line defined by (ρ, θ). Conceptually, for any line (ρ, θ), $G(\rho, \theta)$ counts the number of edge pixels (x,y) which belong to the line given by $\rho - x\cos\theta - y\sin\theta = 0$, weighted by the gradient magnitudes. As the Radon transform identifies lines in images, we would like to formulate a conceptually similar transform that will identify the free parameters describing the motion of a spherical camera.

A restricted camera motion as described by the constraint (3) is identified by three individual rotations. If we parameterize the full rotation group SO(3) with ZYZ Euler angles α, β, and γ so that $R(\alpha, \beta, \gamma) = R_z(\gamma)R_y(\beta)R_z(\alpha)$, we can rewrite (3) as

$$(R_2^T R_1 p \times R_2^T q)^T \vec{e_3} = 0, \quad (4)$$

where $R_2 = R(0, \gamma, \beta)$ and $R_1 = R(\alpha, 0, 0)$. A camera motion is now identified with the rotation pair (R_1, R_2). For each such motion, we we want to count the number of point pairs (p, q), where p is an image point of the first image, and q the second, such that (p, q) satisfies the motion constraint (4), weighted by the similarity of the points p, q.

Unlike motion estimation algorithms which rely on a best-fit minimization, accumulator algorithms like the Hough transform implicitly account for a large percentage of outliers. Thus we need not search for the perfect similarity measure which will eliminate false positives. The objective is simply to identify a measure which is slightly more discriminating than a simple image-based comparison. Our proposal is to use SIFT image features [9] which compute distinguishing characteristics such as local gradient orientation distributions. SIFT features typically associate a 128 dimensional feature vector with each feature location. Using this notion of a similarity between features, we can formulate an integral transform to compute the validity of each rigid motion:

$$G(R_1, R_2) = \int_p \int_q g(p,q)\delta(((R_2^T R_1 p \times R_2^T q)^T \vec{e_3})dpdq \quad (5)$$

Here the soft characteristic function δ measures how close the feature pair (p, q) comes to satisfying the motion constraint (4). The weighting function $g(p,q)$ measures the similarity between features p and q, and is given as

$$g(p,q) = \begin{cases} e^{-||p-q||} & \text{if features have been extracted at } p \text{ and } q \\ 0 & \text{otherwise} \end{cases}$$

where $||p-q||$ is the measure of difference between two features. When using SIFT, this difference can be as simple as the Euclidean distance between two feature vectors.

Before going further, we should make note that the lone remaining rotational parameter $R_1 = R_z(\alpha)$ is not a complete unknown. In fact, we can always integrate the angular velocity from the IMU to obtain an estimate for α. Although this measurement may be corrupted by drift, in some circumstances it may still provide an initial starting value to help reduce the possible search space. This scenario motivates us to decouple the rotational term R_1 from the translational component of motion. In the following section we will see how fixing and searching over α can reduce the computation of our Hough transform (5) to a simple convolution of two spherical images.

III. HOUGH AS CONVOLUTION

Suppose we fix the rotation angle α. This implies that we have completely rectified or "de-rotated" our spherical images so that our search is now limited to just a translational direction. We can rewrite our Hough transform without the rotational term α, leaving us with

$$G(R_2) = \int_p \int_q g(p,q)\delta(((R_2^T p \times R_2^T q)^T \vec{e_3})dpdq \quad (6)$$

$$= \int_p \int_q g(p,q)\delta((R_2^T (p \times q))^T \vec{e_3})dpdq \quad (7)$$

This formulation is quite different than the one we originally started with (5). Previously, for each motion we were interested in accumulating all the possible feature pairs (p, q) between two spherical images which satisfied the epipolar constraint. In this new simplified Hough, we are interested in accumulating points $(p \times q) \in \mathbb{R}^3$ which satisfy the constraint $(R_2^T (p \times q))^T \vec{e_3} = 0$. We can equivalently consider normalized points $\omega = \frac{p \times q}{||p \times q||}$ such that $(R_2^T \omega)^T \vec{e_3} = 0$. ω is ill-defined when $q = \pm p$, but this can occur for only a negligible subset of possible point pairs, which are easily omitted. Now that we are considering points $\omega \in \mathbb{S}^2$, our similarity function $g(p,q)$ must change accordingly. Since the projection $(p_1 \times p_2) \in \mathbb{R}^3 \mapsto \omega \in \mathbb{S}^2$ is not unique, our weights are generated by summing over all pairs which are equivalent in this mapping:

$$g(\omega) = \sum_{p \in I_1} \sum_{q \in I_2} e^{-||p-q||}\delta(||\omega \times (p \times q)||) \quad (8)$$

Re-examining our Hough transform we see that

$$G(R_2) = \int_\omega g(\omega)\delta((R_2^T \omega)^T \vec{e_3})d\omega. \quad (9)$$

This Hough is a correlation between two functions defined on the sphere $\mathbb{S}^2$, where the correlation *shift* in this case is performed by elements of the rotation group $SO(3)$. We proceed to show how this correlation can alternatively be phrased as a convolution of two spherical signals.

Our characteristic function $\delta(\omega^T \vec{e_3})$ is just the image of the equatorial great circle, which corresponds to a camera translating along the Z-axis. Now consider what happens to δ as it is rotated by an element of $SO(3)$. We can write $\omega \in \mathbb{S}^2$ as a rotation of the north pole vector $\vec{e_3}$, just as we did for the translation vector T (2). By making the substitution $\omega = R_3\vec{e_3}, R_3 \in SO(3)$, we have

$$G(R_2) = \int_{R_3} g(R_3\vec{e_3})\delta((R_2^T R_3\vec{e_3})^T \vec{e_3})dR_3 \quad (10)$$

Since δ is just the image of the equator, a rotation of δ by an element $R \in SO(3)$ is equivalent to a rotation by its inverse R^T:

$$\delta((R_2^T R_3)\vec{e_3})^T \vec{e_3}) = \delta((R_3^T R_2)\vec{e_3})^T \vec{e_3}) \quad (11)$$

Remember that $R_2 = R_z(\gamma)R_y(\beta)$ is the rotation that determines the direction of camera translation as $T = R_2\vec{e_3}$ (here we have normalized the translation vector to unit length). (10) becomes

$$G(T) = \int_{R_3} g(R_3\vec{e_3})\delta((R_3^T T)^T \vec{e_3})dR_3, \quad (12)$$

which is the exact definition of the convolution of two spherical signals. From the appendix we know that the Spherical Fourier coefficients of $G(T)$ can be computed as

$$\hat{G}_m^l = 2\pi\sqrt{\frac{4\pi}{2l+1}}\hat{g}_m^l\hat{\delta}_0^l \quad (13)$$

where $\hat{g}$ and $\hat{\delta}$ can be obtained from (15). Subsequently, we can take an inverse transform (14) of $\hat{G}$ to obtain the real values of the original Hough space $G(T)$.

A. Algorithm

We now present the full ego-motion estimation algorithm in Figure (2). It is important to note that the resolution of our Hough space $G(T)$ depends directly upon the band-limit we assume for g and δ when computing the Spherical Fourier Transform. If the band-limit is chosen to be L, we will obtain a result for $G(T)$ with $2L$ uniformly spaced samples in both angles ($\beta \in [0,\pi], \gamma \in [0,2\pi]$). We are now ready to discuss experimental results displaying the effectiveness of this approach.

IV. EXPERIMENTS

In this section we will address some practical considerations regarding the computation of our ego-motion parameters, followed by experimental results.

INPUT
1) Omnidirectional images I_1, I_2, and gravity vector from two robot locations

OFFLINE
1) Compute the spherical Fourier coefficients $\hat{\delta}$

ONLINE
1) Generate SIFT feature sets p, q from images I_1, I_2.
2) Map feature point locations from Omni image to sphere: $p_i \mapsto P_i \in \mathbb{S}^2$, $q_i \mapsto Q_i \in \mathbb{S}^2$
3) Rectify spherical features P, Q using gravity vector from IMU.
4) Discretize search space for $\alpha \in [0, 2\pi)$
5) Initialize $\alpha_{\max} = 0, G_{\max} = 0$.
6) Replace feature sets with Cartesian product of feature sets: $P \times Q$.
7) For each α_i in search space:
    a) $\omega_i = R_z(\alpha_i)P_i \times Q_i$.
    b) Compute $g(\omega)$ from (8) and then $\hat{g}$ from SFT.
    c) Compute $\hat{G}$ from (13).
    d) Obtain $G(T)$ from inverse SFT of $\hat{G}$.
    e) if $\max(G(T)) > G_{\max}$:
        • $\alpha_{\max} = \alpha_i$.
        • set $T_{\max}$ to peak position of $G(T)$.
        • $G_{\max} = \max(G(T))$.
8) Motion estimate is $\alpha_{\max}$ and $T_{\max}$.

Fig. 2. The algorithm to estimate three unknown parameters of motion

A. Spherical Images

As we have discussed earlier, the projections of catadioptric systems with a unique effective viewpoint are equivalent to a projection on the sphere followed by a projection from a point on the sphere axis to the plane [7]. If calibrated, such a sensor enables us to interpolate spherical perspective images. We used a system consisting of a Canon Powershot G2 digital camera along with a parabolic mirror attachment produced by Remote Reality. The mirror's field of view is $212°$. The images are mapped to the sphere by interpolating onto the θ-ϕ grid, where angular sampling is uniform. Figure (3) shows a sample catadioptric image obtained from a parabolic mirror and its corresponding projection onto the sphere.

Our first experiment is designed to test the merits of the full algorithm given in Figure 2 as an effective estimator of motion. Our input data set is a sequence of synthetic spherical images simulating a camera moving along a tilted circle in space. Ten images were generated along this circle, and so the first and last image positions were identical. The orientation of the camera at each point was selected at random. The motion between all consecutive pairs of images was estimated using the full algorithm of Figure 2. Each successive run of this experiment was performed with an increasing corruption applied to the IMU input gravity vector. Figure 4 shows the estimated
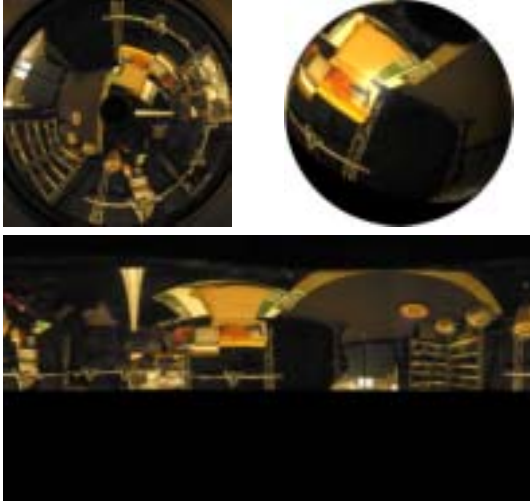
Fig. 3. Top Left: a parabolic catadioptric image. Bottom: the corresponding spherical image on a uniformly sampled θ-ϕ grid. Top Right: the spherical image as it would appear on the surface of the sphere
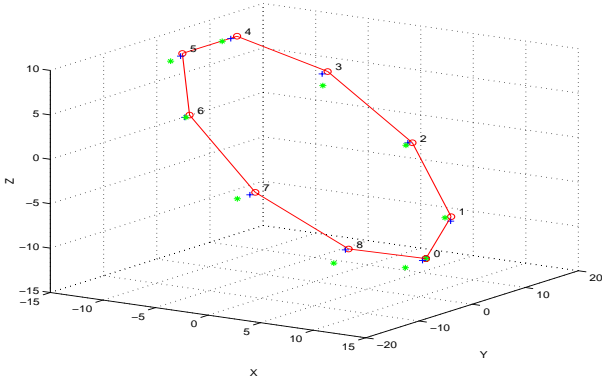

Fig. 4. A simulated path of a camera moving in a circle in space. The motion was estimated between every two consecutive images. Positions marked by circles correspond to noise-free simulation. The path traced by the '+' is for noise of $3°$ in the simulated IMU measurement. The '*' is for noise of $6°$ in the IMU readings. Note that for such a significant IMU error ($6°$) there is still very little drift in the measurements by the time we return full circle to position 0.

camera location for three different trials: first with the input gravity vector as ground truth, second with $3°$ error in the input gravity vector, and third with $6°$ error in the gravity vector. The motion estimate from each tested pair of images was then concatenated to produce an estimate of the trajectory followed by the camera. We are able to display this path because the magnitude of translation was kept fixed between each position along the circle. As the results indicate, the estimation is reliable even in the presence of what would be significant IMU error.

In Figure 5 we show the estimation result for a purely translational sequence. To test the stability for larger rotations against ground truth we also perform artificial rotations to one image before estimation. The small errors in the estimated angle of rotation α for rotations up to $60°$ show that for the estimation is accurate even for large rotations.
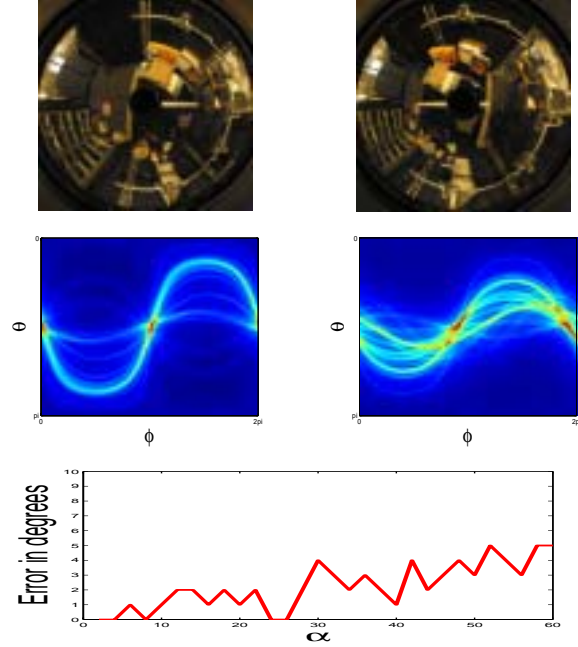

Fig. 5. Top Row: Omnidirectional images taken from camera locations where relative motion is a pure translation along the X axis. Middle Left: the translational Hough space for the estimated motion from the images on the top row. The translational direction of the X-axis is clearly identified with strong peaks in the middle. Middle Right: Simulated rotations were tested using the pair of real translational images. Through all rotations up to $60°$, the largest rotational error found was $5°$ and this is the translational Hough space for one such instance. Bottom: Error in estimated angle of rotation α for rotations up to $60°$.

Figure 6 shows the estimation results for a pair of images separated by an arbitrary one-parameter rotation and translational direction. In lieu of reliable ground truth the fully rectified spherical images are shown overlayed with the estimated translational direction and epipolar circles (see figure caption for details).

V. CONCLUSION

In this paper we have presented a novel ego-motion estimation algorithm fusing IMU measurements with visual information. The vision component of the algorithm successfully estimates ego-motion without first having to compute feature correspondences between images. Preliminary results indicate the Hough transform is a viable tool for motion estimation, and in our future work we will attempt to export this algorithm to a mobile robot equipped with an Omnicam and IMU.

APPENDIX

This section is designed to provide a cursory introduction to the theory of harmonic analysis on the sphere. Readers are referred to [5] for a thorough exposition regarding the computation of a discrete Spherical Fourier Transform and a proof of the spherical convolution theorem.

As the angular portion of the solution to Laplace's equation in spherical coordinates, the spherical harmonic
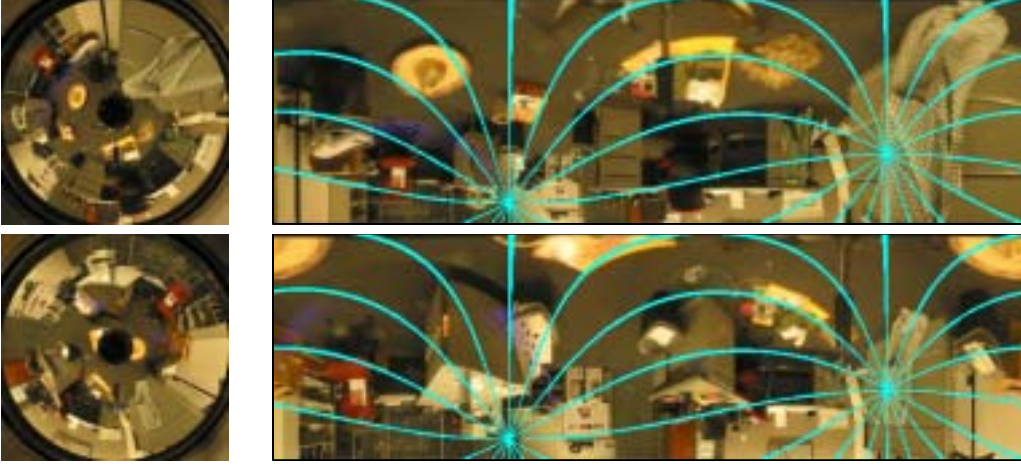
Fig. 6. On the left is a pair of Omnidirectional images separated by a rotation $R_{\vec{z}}(\alpha)$ and a free translation. After the ego-motion algorithm was run, the spherical images were rectified for the correct α. On the right are the rectified spherical images (only the visible band on the sphere is shown here). The great circles representing a subset of the epipolar circles are overlayed on the images. Points that lie on a great circle in the first image will lie on the same great circle in the second image if the estimation is correct. From visual inspection this result is very accurate but not perfect. The two antipodal spherical points representing the intersection of all the circles is in fact also the direction of translation.

functions Y_m^l form a complete orthonormal basis over the unit sphere:

$$Y_m^l(\theta, \phi) = (-1)^m \sqrt{\frac{(2l+1)(l-m)!}{4\pi(l+m)!}} P_m^l(\cos\theta)e^{im\phi},$$

where $P_m^l(\cos(\theta))$ are associated Legendre polynomials. Thus, for any function $f(\omega) \in L^2(S^2)$, we have a Spherical Fourier Transform (SFT) given as

$$f(\omega) = \sum_{l \in \mathbb{N}} \sum_{|m| \leq l} \hat{f}_m^l Y_m^l(\omega) \quad (14)$$

$$\hat{f}_m^l = \int_{\omega \in S^2} f(\omega)\overline{Y_m^l}(\omega)d\omega \quad (15)$$

An important property of the spherical harmonic functions Y_m^l is

$$Y_m^l(R^{-1}\omega) = \sum_{|k| \leq l} Y_k^l(\omega)U_{km}^l(R), \quad (16)$$

where the $(2l+1) \times (2l+1)$ matrices U^l are the irreducible unitary matrix representations of the transformation group $SO(3)$, whose elements are given by

$$U_{mk}^l(R) = e^{-im\alpha}P_{mk}^l(\cos\beta)e^{-ik\gamma}. \quad (17)$$

The P_{mk}^l are the generalized Legendre polynomials. From (16) we obtain a Shift Theorem relating coefficients of rotated functions :

$$h(\omega) = f(R^{-1}\omega) \Leftrightarrow \hat{h}_m^l = \sum_{|k| \leq l} \hat{f}_k^l U_{mk}^l(R) \quad (18)$$

This Shift Theorem (18) shows us that the U^l matrix representations of the rotation group $SO(3)$ are the spectral analogue to 3D rotations. As vectors in $\mathbb{R}^3$ are rotated by orthogonal matrices, the $(2l+1)$-length complex vectors $\hat{f}^l$, comprised of all coefficients of degree l, are transformed by the unitary matrices U^l.

The convolution of two functions $f, h \in L^2(\mathbb{S}^2)$ is defined as

$$(f * h)(\omega) = \int_{R \in SO(3)} f(R\vec{e_3})h(R^T\omega)dR \quad (19)$$

From the SFT (14, 15) and the Shift Theorem (18), we can write the coefficients of $f * h$ as

$$(f \hat{*} h)_m^l = 2\pi\sqrt{\frac{4\pi}{2l+1}}\hat{f}_m^l\hat{f}_0^l \quad (20)$$

This result is similar to the convolution theorem of function defined on the real line, which illustrates that the spectral analogue to the convolution integral is just the multiplication of Fourier descriptors.

REFERENCES

[1] In Markus Vincze, editor, *Workshop on Integration of Vision and Inertial Sensors*. 2003.
[2] J. Aloimonos and J. Y. Hervé. Correspondenceless stereo and motion: Planar surfaces. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12:504–510, 1990.
[3] R. Azuma, H Neely, M. Daily, and M. Correa. Visualization of conflicts and resolutions in a free flight scenario. In *Proceedings of IEEE Visualization*, pages 433–436, 1999.
[4] D. Burschka and G. Hager. V-gps(slam): Vision-based inertial system for mobile robots. In *Proc. IEEE Int. Conf. on Robotics and Automation*, 2004.
[5] J.R. Driscoll and D.M. Healy. Computing fourier transforms and convolutions on the 2-sphere. *Advances in Applied Mathematics*, 15:202–250, 1994.
[6] C. Geyer. Euclid meets fourier. In *Workshop on Omnidirectional Vision*, Prague, 2004.
[7] C. Geyer and K. Daniilidis. Catadioptric projective geometry. *International Journal of Computer Vision*, 43:223–243, 2001.
[8] D. Lowe. Sift (scale invariant feature transform): Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004.
[9] D. G. Lowe. Object recognition from local scale-invariant features. In *Proc. Int. Conf. on Computer Vision*, pages 1150–1157, Kerkyra, Greece, Sep. 20-23, 1999.
[10] S. Roy and I.J. Cox. Motion without structure. In *Proc. Int. Conf. on Pattern Recognition*, Vienna, Austria, 1996.
[11] J. Shi and C. Tomasi. Good features to track. In *IEEE Conf. Computer Vision and Pattern Recognition*, Hilton Head Island, SC, June 13-15, 1994.