Real Time Biologically-Inspired Depth Maps from Spherical Flow

Chris McCarthy, Nick Barnes and Mandyam Srinivasan

Abstract— **We present a strategy for generating real-time relative depth maps of an environment from optical flow, under general motion. We achieve this using an insect-inspired hemispherical fish-eye sensor with 190 degree FOV, and a de-rotated optical flow field. The de-rotation algorithm applied is based on the theoretical work of Nelson and Aloimonos [12], who outline an algorithm for obtaining all rotational components of motion on a sphere about any great circle. From this we may obtain the translational component of motion, and construct full relative depth maps on the sphere. We demonstrate the robustness of this strategy in both simulation and real-world results. To our knowledge, this is the first demonstrated implementation of the Nelson and Aloimonos algorithm working in real-time, over real image sequences. These preliminary results provide a compelling argument for the global interpretation of optical flow under spherical projection when inferring scene structure. They also demonstrate the uses of real time optical flow for depth mapping and obstacle avoidance.**

I. Introduction

Essential to autonomous navigation is the ability to perceive depth. While absolute measures of distance are useful, they are not necessary for achieving most navigation tasks. Relative measures of distance to surfaces have been shown to be sufficient for autonomously navigating corridors [14], avoiding obstacles [15], [3], and docking with objects in the environment [11], [14]. In addition, obtaining depth maps across a wide field of view provides a means of perceiving environmental structure, which in turn may be used for higher level navigation tasks such as invoking appropriate navigational subsystems and mapping.

It is well known that biological vision systems perceive depth from a variety of cues, depending on the configuration and the geometry of the eye. Among the potential cues are (a) stereo information (b) depth from focus (c) depth from convergence and (d) depth from optical flow [18]. Insects, with their immobile, fixed-focus eyes and low interocular separation, rely heavily on optical flow cues to obtain depth information [19]. Information from the optical flow that is generated in the eyes by the insects' motion in the environment is used to (i) navigate safely through narrow gaps (ii) detect and avoid collisions with objects (iii) distinguish objects from their immediate backgrounds and (iv) orchestrate smooth landings [19]. Estimating depth from optical flow is simpler, computationally, than estimating depth from stereo, and is thus an attractive strategy for the relatively simple nervous systems of insects. Given many robotic vision systems are equipped with only a single camera, much attention has been given to those cues that do not rely on two or more simultaneous views of the scene, such as optical flow.

In computer vision, the use of optical flow for scene reconstruction has been problematic. While a significant body of theoretical work exists [9], [1], [13], [7], in practice, these approaches generally lack the speed and robustness required for a real-time navigation system. One issue is the intolerance of such strategies to noisy flow estimates in local regions [20]. Optical flow estimation is notoriously noisy, and difficult to compute accurately under real-world conditions. For this reason, the examination of local flow vectors to infer scene structure has been largely abandoned.

An additional issue for depth map recovery is the reliable extraction of the translational component of flow (referred to as *de-rotation*), from which relative depth can be inferred. This is particularly difficult when the field of view is narrow, due to increased coupling between translation and rotation [5]. For this reason, many flow-based depth perception strategies either assume pure translational motion of the sensor (*e.g.* [4]), or apply planar models to extract surfaces from the scene (*e.g.* [15], [16]). While de-rotation algorithms exist, these are largely constrained to a single rotation, or are not fast or robust enough for real-time depth mapping. To generate full 3D depth maps from optical flow, under general motion, a de-rotation strategy is required for each rotational component. To obtain workable depth maps for navigation from flow, the derotation algorithm must be sufficiently accurate, and must run in real-time. There presently exists no such real-world system capable of achieving this.

Traditionally, a perspective camera model has been used when inferring scene structure from optical flow. This is in contrast to insect vision, where the compound eye structure of most insects provides an almost global view of the scene [4]. There is a growing body of theoretical work suggesting a spherical projection model, as an approximation to this view, may offer distinct advantages when inferring scene structure and self-motion from optical flow [6], [12]. Geometric properties of the sphere have been shown to facilitate more efficient and robust interpretations of optical flow. Brodsky *et al.* [2] show that on a full view sphere, optical flow can be un-ambiguously interpreted on the basis of the direction of flow vectors alone. Nelson and Aloimonos [12] highlight specific advantages gained through the existence of both a focus of expansion (FOE) and focus of contraction (FOC) in a single spherical image. From such observations, a potentially real-time de-rotation algorithm is derived for the complete recovery of rotational velocity components

C. McCarthy and N. Barnes are with the Vision, Science and Technology Programme,
National ICT. Australia
M. Srinivasan is with the Centre for Visual Sciences,
The Australian National University

from the optical flow on a full view sphere. While some theoretical analysis of the algorithm's likely robustness in real-world conditions is provided, there exists no published results to date reporting the algorithm's performance in real-time, and over real image sequences. Given the potential of this algorithm to support a wide variety of navigation tasks, it is of interest to examine its plausibility for facilitating the real-time recovery of 3D depth maps.

To this end, we have implemented the Nelson and Aloimonos de-rotation algorithm, and applied it to the task of generating 3D relative depth maps from a spherical sensor. In this paper we report preliminary results showing the algorithm does provide adequate support for real-time depth map recovery. In simulation, we show its application over a full view sphere undergoing general motion. Using a hemispherical view fish-eye camera with a 190 degree field of view, we also present preliminary results over two real image sequences captured during the ground-based motion of the hemispherical sensor. We show that the recovery of robust 3D relative depth maps can be achieved in real-time, and without the need for camera calibration. These experimental results appear to support existing theoretical arguments in favour of a spherical projection model when inferring scene structure, and self-motion, from optical flow.

II. Generating depth maps from spherical flow

We briefly outline the theory for recovering 3D depth maps from the spherical projection of optical flow. First, we define optical flow on the sphere. We can define the position of any point on a view sphere, θ, in terms of its angular location on three great circles lying in orthogonal planes, such that:

$$\theta = \left[\begin{array}{ccc} \theta_x & \theta_y & \theta_z \end{array} \right], \tag{1}$$

where θ_x, θ_y and θ_z are angles in the direction of each orthogonal great circle, $\mathbf{E_x}$, $\mathbf{E_y}$ and $\mathbf{E_z}$, in the range $[0, 2\pi]$, as shown in Figure 1.

Using this representation, we may express any optical flow vector on the sphere in terms of the components of motion in the direction of each orthogonal great circle, such that:

$$f(\theta) = \left[\begin{array}{ccc} e_x(\theta) & e_y(\theta) & e_z(\theta) \end{array} \right], \tag{2}$$

where e_x, e_y and e_z are components of flow in the direction of each great circle indicated by its subscript. We define each of these as:

$$e_x(\theta) = \frac{v}{R(\theta)} sin(\phi_x - \theta_x) + \omega_x, \tag{3}$$

$$e_y(\theta) = \frac{v}{R(\theta)} sin(\phi_y - \theta_y) + \omega_y, \tag{4}$$

$$e_z(\theta) = \frac{v}{R(\theta)} sin(\phi_z - \theta_z) + \omega_z, \tag{5}$$

where ω_x, ω_y and ω_z are rotational velocity components about each axis perpendicular to the plane of each great circle, $R(\theta)$ is the radial depth to the scene point projecting to θ and v is the translational velocity of the sphere in the direction $[\phi_x, \phi_y, \phi_z]$.

It is important to note that the above equations are defined for any three great circles lying on orthogonal planes, and are not limited to equators about the X, Y and Z axis. As such, the above equations show that in addition to the translation, flow in the direction of any great circle is effected by only a single rotation about the axis perpendicular to the great circle's plane. This observation has lead to the development of a full de-rotation algorithm for optical flow on the sphere.

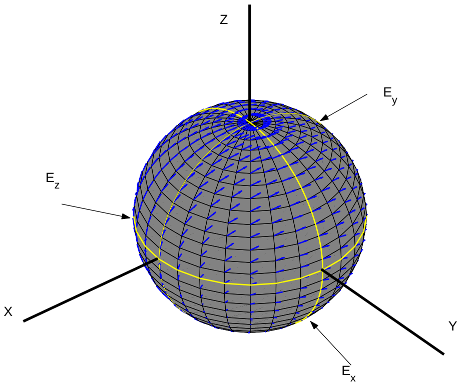

Fig. 1. Optical flow on the view sphere.

A. De-rotating flow on the sphere

In the late eighties, Nelson and Aloimonos [12] proposed an algorithm for recovering the full 3D rotational velocities of a view sphere, in real-time, by exploiting three key geometric properties of optical flow on the sphere:

1) the component of flow parallel to any great circle is effected only by the rotational component about its perpendicular axis, thus decoupling it from rotations about orthogonal axes.
2) under pure translation, both the FOE and FOC will co-exist at antipodal points on the sphere, and will evenly partition flow along any great circle connecting these two point, into two distinct directions of motion (*i.e.* clockwise and counter-clockwise).
3) the existence of any rotational motion along a great circle causes the FOE and FOC to converge, thus ensuring the two points will only lie at antipodal locations under pure translation.

The first observation indicates that each component of rotation about can be resolved independently, and thus each may be considered in turn. From the second and third observations, Nelson and Aloimonos propose an algorithm for recovering the rotational component of flow, ω, about any great circle of flow $e(\theta)$. For reference, we reproduce the algorithm in pseudo code here (see Algorithm (1)).

In words, the algorithm describes a simple search-based strategy for resolving rotation. For each discrete point, θ_c, on a circle of flow, $e(\theta)$, a range of rotations are searched

Algorithm 1 Nelson and Aloimonos De-rotation Algorithm [12]

```
for $\omega_c = \omega_{min}$ to $\omega_{max}$ do
    $D[\omega_c] = 0$
    for $\theta_c = 0$ to $2\pi$ do
        $a = e(\theta_c) - \omega_c$
        $\beta = \phi - \theta_c$
        if $a < 0$ and $0 \leq \beta < \pi$ then
            $result = -a$
        else if $a > 0$ and $\pi \leq \beta < 2\pi$ then
            $result = a$
        else
            $result = 0$
        end if
        $D[\omega_c] = D[\omega_c] + result$
    end for
end for
$\omega = min\_index(D[\omega_{min} : \omega_{max}])$
return $\omega$
```

through, where each candidate, ω_c, is used to de-rotate flow along the circle. After derotation, the sum of the residual flow on the circle is taken. Note that the sign of the flow indicates its direction on the circle, therefore, a perfect split of clockwise and counter-clockwise flow will yield a sum of 0. Accounting for noise and quantisation errors, the chosen rotation is therefore the one which yields the smallest sum of flow on the circle after derotation.

By applying this algorithm to great circles about each rotational axis, the complete recovery of the sphere's rotation is achieved. After de-rotation, the direction of translation is also given by the line passing through the FOE and FOC.

Notably, the algorithm's run time performance is dictated primarily by the quantisation of discrete locations on the great circle, and the range of possible rotations for each great circle. Given reasonable choices, the algorithm should provide fast execution [12].

It is interesting to note that despite the potential use of this algorithm for recovering self-motion and scene structure in real-time, no published results currently exist reporting its application to real-time navigation tasks, in real-world scenarios. The authors themselves provide results from simulation, and some theoretical analysis of the algorithm's robustness to noisy and/or absent flow. Given the encouraging results they report, it is of interest to apply this algorithm to real-time navigation tasks, such as generating relative depth maps.

B. Generating relative depth maps

After the removal of all rotation components, all optical flow vectors follow great circles passing through the FOE and FOC. Thus, we may express the magnitude of the residual translational optical flow t a discrete location θ on such a great circle as:

$$f(\theta) = \frac{v}{R(\theta)} sin(\phi - \theta). \tag{6}$$

TABLE I

SIMULATION ERROR MEASURES

Gauss noise (std dev)	Rotational error			Trans dir error	Depth error
	ω_x	ω_y	ω_z		
0^o	0.003	0.003	0.003	5.9^o	8.9%
2^o	0.003	0.003	0.004	9.0^o	13.0%
4^o	0.006	0.005	0.007	10.0^o	25.3%
10^o	0.009	0.008	0.012	16.6^o	39.2%

Assuming a static environment, we may define an equation for obtaining the radial distance to any scene point projecting onto a great circle passing through the FOE and FOC as:

$$R(\theta) = v\frac{sin(\phi - \theta)}{f(\theta)}. \tag{7}$$

Given the Nelson and Aloimonos algorithm also recovers the direction of translation ϕ, the only remaining unknown is the sensor's translational velocity. Knowing this allows the complete recover of absolute scene depth. Under general motion however, this is typically unavailable, and so only a relative measure of depth can be obtained, whereby the depth of points in the scene are scaled by v such that:

$$\frac{R(\theta)}{v} = \frac{sin(\phi - \theta)}{f(\theta)}. \tag{8}$$

While not an absolute measure, the above definition is sufficient for recovering scene structure, and for most navigational tasks.

It is important to note that (8) is only defined where optical flow exists (*i.e.* $f(\theta) \neq 0$). Thus, range cannot be reliably measured where a lack of texture exists, or where flow magnitude tends to zero such as at the FOE and FOC. Given a spherical view of a sufficiently textured environment, enough features should exist to obtain workable depth maps for scene structure recovery.

C. Simulation results

Simulation tests were conducted to examine the robustness of depth map estimates obtained using the strategy outlined above. For this, a model of a unit view sphere was defined, and immersed in a virtual 3D boxed space. A ground truth depth map was then obtained over the whole sphere, from which accuracy could be measured. Optical flow fields were computed on the view sphere for multiple sets of twenty randomly chosen translation directions, with randomly chosen components of rotation about each principle axis. Rotational velocity values were within the range $[-0.5, 0.5]$. To examine robustness, increasing levels of Gaussian noise were added to the angular component of each flow vector estimate for each set of twenty trials. On each equator, 112 discrete points, and 100 possible rotations were used for derotation. Table I provides mean errors obtained during each simulation runs. Rotational errors are given as the mean of the absolute difference between the estimated and true rotation velocities for each simulation run. The translational direction error is given as the mean angular error between the estimated translational direction and ground truth. Depth map estimation errors are given as the mean of relative errors against ground truth.

It is interesting to note from Table I, that mean rotational errors exhibit high stability, suggesting high robustness to angular errors in flow estimation. Translational direction errors also appear to remain stable. It is important to note that quantisation effects dictate that estimates of translational direction can only estimate the the true direction of motion to within 3.2^o of the actual direction, hence the presence of non-zero errors when no noise is introduced.

It is clear that depth estimation errors increase with noise. This is influenced further by errors in translational direction. It was found during simulation runs, however, that depth estimation errors were particularly large around the FOE and FOC, where flow magnitudes approach zero. These were not filtered out during simulation runs, and thus have significant influence on the overall accuracy levels reported. Under less precise, real-world conditions, this issue is less likely to influence accuracy due to noise levels preventing flow magnitudes from diminishing to such small values. Under real-world conditions, it can also be expected that motion will not be as random and discontinuous as is depicted here. Rather, it is likely to exhibit significantly greater amounts of translational motion, thereby providing larger contributions of translational flow about the great circles from which the direction of motion is estimated. This should improve both the translation estimates obtained, and relative depth estimates.

III. Real-world experiments

The 3D depth map algorithm was implemented for use with a single fish-eye camera (Unibrain Fire-i BCL 1.2[1]) undergoing ground-based motion. Given motion was approximately planar, only a single hemispherical view was required to account for rotation on the ground plane. The fish-eye camera has a 190 degree FOV, and thus provides a suitable approximation to a hemispherical projection of the scene, thereby alleviating the need for mirrors.

In all experiments, full estimates of optical flow were acquired using Lucas and Kanade's gradient-based method [8], in combination with Simoncelli's multi-dimensional paired filter [17] for pre-filtering and gradient estimation. This combination was chosen on the basis of strong performances in a recent comparison of optical flow methods for real-time robot navigation [10].

Two image sequences were constructed for the depth map experiments. In both experiments, the camera's forward velocity was kept approximately constant, while rotation about the axis perpendicular to the ground plane was introduced. No camera calibration was performed or post-filtering of flow estimates, translational direction or relative depth estimates. On a 2.1 GHz machine, depth maps were generated at rate of 1.2 updates per second over circular image regions of radius 110 pixels, with an input frame rate of 15Hz.

A. De-rotation

Given ground-based motion, the Nelson and Aloimonos derotation algorithm was implemented for a single rotation, using a circle of evenly distributed image points (radius 110 pixels) around the estimated projective centre of the camera. Due to limitations imposed by the size of the camera's image plane, the true great circle could not be used. For planar motion, however, the component of flow in the direction of any concentric circle about the rotational axis is sufficient for derotation.

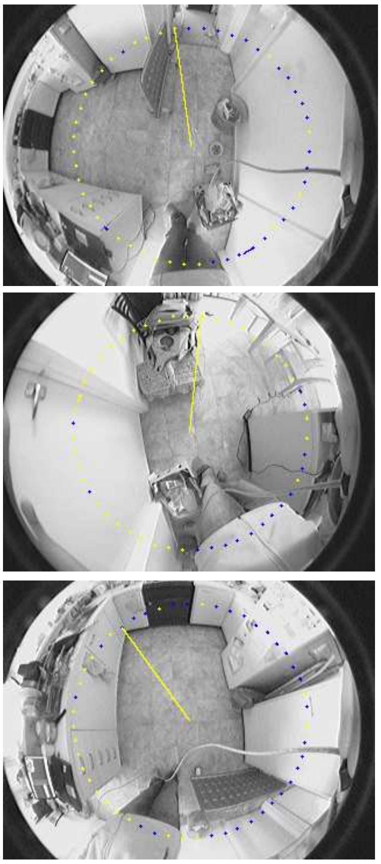

Fig. 2. Sample frames showing the circle used to de-rotate the fbw. Blue pixels indicate fbw that is travelling clockwise and yellow counter-clockwise (before de-rotation). The yellow line indicates the estimated direction of translation after de-rotation.

Figure 2 provides sample frames showing the output of the derotation algorithm over a real image sequence (described later). The blue and yellow points indicate the locations on the circle used for derotation. Blue points indicate clockwise flow, and yellow, counter-clockwise. The yellow line signifies the estimated direction of translation after de-rotation is applied. In Figure 2(a), it can be seen that the direction of flow is evenly partitioned by the estimated direction of translation, thus indicating almost pure translational motion is present. In contrast, the imbalance of partitioned flow in Figures 2(b) and (c) suggests significantly greater amounts of rotation exist. This is also reflected in the estimated rotation for both samples.

While a quantitative assessment of the algorithm's accuracy over real images is yet to be conducted, preliminary observations are encouraging. Highly robust performances have been observed over a wide range of real-world im-

[1] Omni-tech robotics

age sequences, where the estimated direction of translation provides a good visual indication of de-rotation accuracy. Video 1 shows the application of the derotation algorithm over the entire sequence from which the above samples were drawn. Note that in Video 1, a simple derivative filter is applied to the estimated translation direction for clarity of viewing. This filtering was not applied during the depth map experiments described below as rotation spirals very quickly, and is therefore difficult to filter accurately, particularly for the hand-held camera sequence.

B. Corridor navigation experiment

The first image sequence was obtained by mounting the camera on-board a mobile robot platform, with omni-directional motion. The camera was fixed as close as possible to the robot's rotation axis, facing upwards. Frames were captured as the robot was manually driven around corridors in our lab.

Figure 3 shows sample depth maps obtained during the corridor experiment. The first column shows the central image (320×240 pixels) from the buffered frames used to compute the optical flow for the corresponding depth map. The second column shows a grayscale map of the relative depths of objects in the scene (brighter is closer) estimated from the de-rotated flow field. The third column provides a top-down view of the relative depths of scene points, projected onto the ground plane (we refer to these as structure maps). The centre of the structure map gives the location of the camera. For this, thresholding was applied to extract only the closest surfaces in the scene (and thus omit depth estimates from the ceiling).

The relative depth maps obtained over the corridor navigation sequences provide a good qualitative representation of the environment. An abundance of clear structural cues resulting from the motion of surface boundaries such as corners, doorways and windows can be seen. In addition, there appears to be good visual evidence of objects in close proximity being detected. This is particularly evident in Figures 3(c) and (d) where the wall edge in (c), and column (and fire hydrant) in (d) show up as the brightest areas in the grayscale depth maps.

The structure maps in the third column of Figure 3 further support the accuracy of the relative depth measures for inferring basic scene structure. Most evident is the extraction of free space from obstructed space in the local area about the robot. This is evident in all samples. It is important to note, however, that space marked as unobstructed may also be the result of a lack of measurable flow in the area. Thus, some surface areas have only a few depth measures associated with them.

Notably, the sequence involves significant variation in lighting conditions as the robot travels beneath fluorescent lights, and past sun lit rooms. While optical flow estimates in these regions are generally unreliable (and often discarded by Lucas and Kanade's eigenvalue thresholding), the wide field of view ensures enough features exist so as to extract the overall scene structure, despite the noise inevitably introduced by these effects.

C. Cluttered environment experiment

A second sequence was constructed depicting the camera's motion through a cluttered kitchen environment. For this, the camera was hand-held, facing toward the ground plane as it was walked through the kitchen. Thus, the motion of the camera is less constrained than in the previous experiment. Relative grayscale depth maps, and ground-plane structure maps were again produced. Four samples from these results are given in Figure 4.

From Figure 4 it can be seen that depth maps obtained exhibit less structural definition than the corridor sequence. This, however, is unsurprising given the relatively unstructured nature of the environment, and the greater abundance of objects in close proximity to the camera.

The camera's orientation towards the ground plane appears to significantly improve the extraction of free space from obstructed space. While evident in the grayscale depth maps, this is made particularly clear in the structure maps, particularly Figure 4(a), where the structure map provides a highly detailed map of free space over a considerable portion of the viewing area. In addition, the structure map shows an abundance of structural cues. Other samples from the sequence also exhibit clear and accurate extractions of free space.

These results are particularly encouraging when considering the camera was hand held and walked through the scene. While motion was predominantly in the horizontal plane, the camera was subject to both rotational motions off the plane, and changes in height throughout the sequence. Despite this, the Nelson and Aloimonos algorithm appears to have provided workable derotation for robust depth mapping in real world conditions.

D. Discussion

The quality of depth maps obtained in both these preliminary experiments are encouraging. While more quantitative testing is needed, it is clear from these results that basic 3D scene structure can be reliably inferred from optical flow estimated over relatively low resolution images, and in real-time. At the very least, these results suggest clear distinctions between free and obstructed space can be obtained. For most low-level navigation tasks where features are available, this is sufficient. While we do not test the use of the generated depth maps in closed-loop control of a vehicle, these results suggest this is entirely plausible.

It should be noted that for a ground-based mobile robot, the entire viewing angle need not be used to generate depth maps sufficient for navigation. In the case of the corridor sequence for example, only the peripheral area of the projected view shows objects that obstruct free space. Thus, considerations of both the robot's physical height, and constraints on its motion may be exploited to limit the depth map generation to the visual range that matters. The resulting

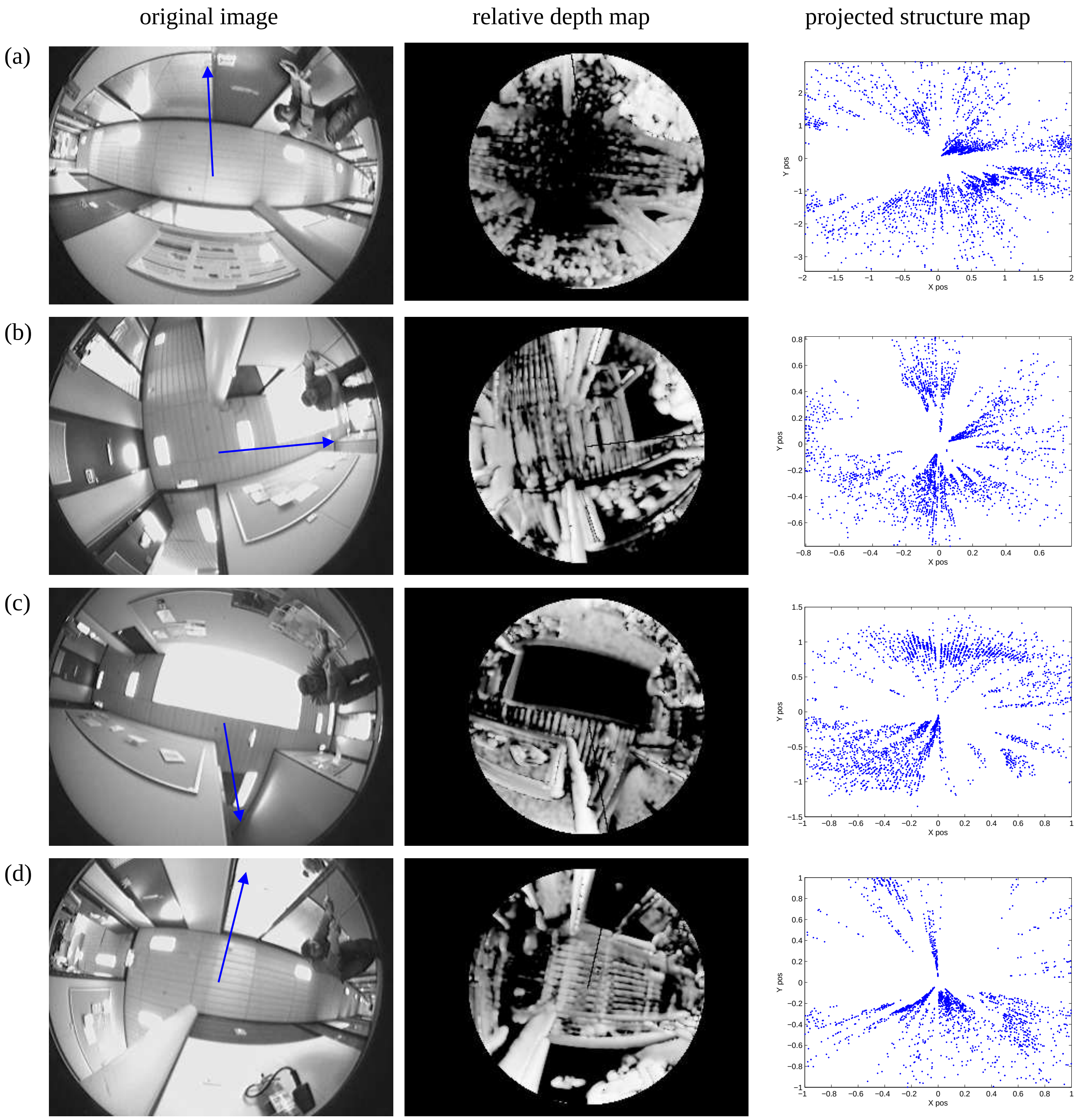

Fig. 3. Sample depth maps obtained on-board the mobile platform (camera facing up). The left column shows the original image, and estimated direction of translation obtained from derotation. The middle column shows grayscale relative depthmaps computed from the translational flow. The right column shows structure maps, obtained by projecting relative depth estimates into 3D space, and then orthographically onto the ground plane.

speed-up in execution time may also allow the possible use of higher resolution in these regions.

These results suggest the Nelson and Aloimonos derotation algorithm is performing well over real-world images. Both sequences depict significant rotations, yet little ill-effects appear in the depth maps obtained. While a thorough examination of the algorithm's accuracy over real-image sequences is still needed, it is evident from both Video 1, and the depth maps generated in both experiments, that the algorithm provides sufficient accuracy to facilitate the real-time recovery of both the direction of ego-motion, and complete 3D relative depth maps.

IV. Conclusion

In this paper, we have presented a strategy for generating 3D relative depth maps from optical flow, in real-time. In so doing, we have demonstrated for the first time, the use of the Nelson and Aloimonos de-rotation algorithm over

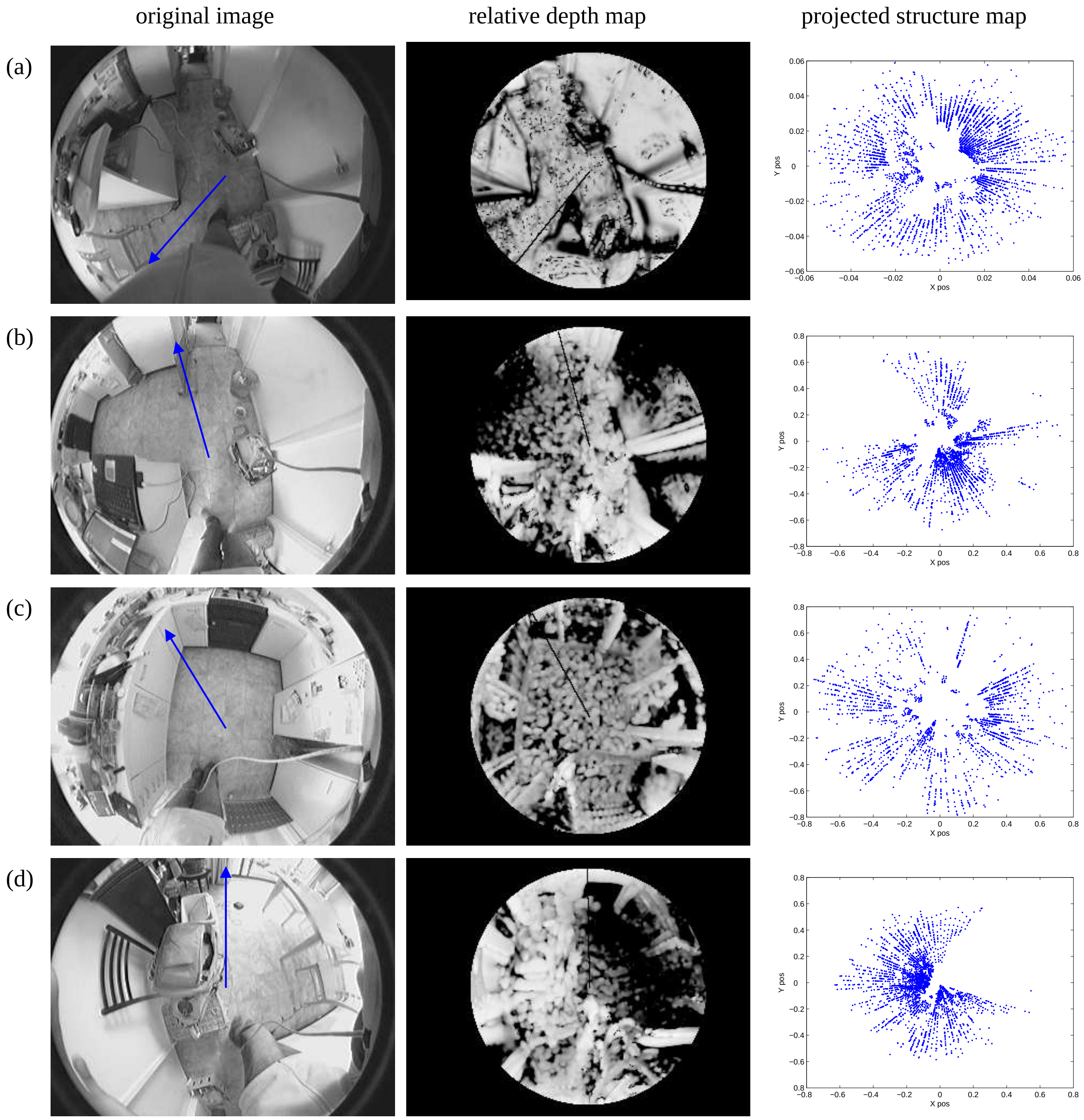

Fig. 4. Sample frames and depth maps from ”Belco” kitchen sequence (hand held camera facing towards ground plane). The left column shows the original image, and estimated direction of translation obtained from derotation. The middle column shows grayscale relative depthmaps computed from the translational flow. The right column shows structure maps, obtained by projecting relative depth estimates into 3D space, and then orthographically onto the ground plane.

real images, depicting real-world environments. Results from simulated full general motion of a sphere, and from real-world experiments suggest this strategy may be a useful base for many navigational sub-systems. In addition, these results further support theoretical arguments in favour of a spherical projection when attempting to infer scene structure and self-motion from optical flow. By exploiting basic geometric properties of optical flow on the sphere, less dependency exists on the accuracy of individual flow vectors, thus improving the overall robustness of the system.

References

[1] G. Adiv, “Inherent ambiguities in recovering 3-d motion and structure from a noisy flow field,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 5, pp. 477–489, 1989.

[2] T. Brodsky, C. Fermüller, and Y. Aloimonos, “Direction of motion fields are hardly ever ambiguous,” *International Journal of Computer Vision*, vol. 26, no. 1, pp. 5–24, 1998.

[3] T. Camus, D. Coombs, M. Herman, and T.-H. Hong, "Real-time single-workstation obstacle avoidance using only wide-field flow divergence," in *Proceedings of the 13th International Conference on Pattern Recognition*, Vienna, Austra, 1996, pp. 323–30.

[4] J. S. Chahl and M. V. Srinivasan, "Range estimation with a panoramic visual sensor," *Journal of the Optical Society of America A*, vol. 14, no. 9, pp. 2144–50, sep 1997.

[5] K. Daniilidis and H.-H. Nagel, "The coupling of rotation and translation in motion estimation of planar surfaces," in *1993 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1993, pp. 188–193.

[6] C. Fermüller and Y. Aloimonos, "Geometry of eye design: Biology and technology," in *Multi Image Search and Analysis*, *Lecture Notes in Computer Science*, T. H. R. Klette and G. Gimelfarb, Eds. Springer Verlag, Heidelberg, 2000.

[7] H. C. Longuet-Higgins and K. Prazdny, "The interpretation of a moving retinal image," *Proceedings of the Royal Society of London*, vol. B208, pp. 385–397, 1980.

[8] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision." in *Proceedings of DARPA Image Understanding Workshop*, 1984, pp. 121–30.

[9] S. Maybank, *Theory of Reconstruction from Image Motion*. Springer, Berlin, 1993.

[10] C. McCarthy and N. Barnes, "Performance of optical flow techniques for indoor navigation with a mobile robot," in *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*, 2004, pp. 5093–8.

[11] ——, "A robust docking strategy for a mobile robot using flow field divergence," in *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (in press)*, 2006.

[12] R. C. Nelson and J. Aloimonos, "Finding motion parameters from spehrical motion fields (or the advantages of having eyes in the back of your head)," *Biological Cybernetics*, vol. 58, pp. 261–73, 1988.

[13] J. H. Rieger and H. T. Lawton, "Processing differential image motion," *Journal of the Optical Society of America A*, vol. 2, no. 2, pp. 354–60, 1985.

[14] J. Santos-Victor and G. Sandini, "Divergent stereo in autonomous navigation: From bees to robots," *International Journal of Computer Vision*, vol. 14, no. 2, pp. 159–77, 1995.

[15] ——, "Uncalibrated obstacle detection using normal flow," *Machine Vision and Applications*, vol. 9, no. 3, pp. 130–37, 1996.

[16] ——, "Visual behaviors for docking," *Computer Vision and Image Understanding: CVIU*, vol. 67, no. 3, pp. 223–38, 1997.

[17] E. P. Simoncelli, "Design of multi-dimensional derivative filters," in *First Int'l Conf on Image Proc*, vol. I. Austin, TX USA: IEEE Sig Proc Society, 1994, pp. 790–3.

[18] M. V. Srinivasan, "How insects infer range from visual motion," in *Visual Motion and its Role in the Stabilization of Gaze*, F. Miles and J. Wallman, Eds. Elsevier, Amsterdam, 1993, pp. 139–156.

[19] M. V. Srinivasan and S. W. Zhang, "Visual motor computations in insects," *Annual Review of Neuroscience*, vol. 27, pp. 679–696, 2004.

[20] A. Verri and T. Poggio, "Motion field and optical flow: qualitative properties," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 5, pp. 490–498, 1989.