

GUMS: A Generalized Unified Model for Stereo Omnidirectional Vision (Demonstrated via a Folded Catadioptric System)

Carlos Jaramillo¹, Roberto G. Valenti², and Jizhong Xiao², *Senior Member, IEEE*

Abstract—This paper introduces GUMS, a complete projection model for omnidirectional stereo vision systems. GUMS is based on the existing generalized unified model (GUM), which we extend in order to satisfy a tight relationship among a pair of omnidirectional views for fixed baseline sensors. We exemplify the proposed model’s calibration via a single-camera coaxial omnistereo system in a joint bundle-adjusted fashion. We compare our *coupled* method against the naive approach where the calibration of intrinsic parameters is first performed individually for each omnidirectional view using existing monocular implementations, to then solve for the extrinsic parameters as an additional step that has no effect on the intrinsic model solutions initially computed. We validate GUMS and its calibration effectiveness using both real and synthetic systems against ground-truth data. Our calibration method proves successful for correcting the unavoidable misalignment present in vertically-configured catadioptric rigs. We also generate 3D point clouds employing the calibrated GUMS systems in order to demonstrate the qualitative outcome of our contribution.

I. INTRODUCTION

A *catadioptric* system is a popular way of achieving omnidirectional vision by combining a monocular camera and a curved reflector. Various applications and mirror-camera configurations exist and have been disseminated in the works by Baker and Nayar [3] and by Yagi [32] among others. The most practical omnidirectional catadioptric arrangement consists of aligning the camera’s optical axis with the axis of symmetry of a non-degenerate quadric mirror [13]. We redirect the reader to [5] and [27] as to survey the vast field of omnidirectional vision sensors (ODVS) and camera models. Here, we are interested in improving the state-of-the-art of omnidirectional catadioptric systems to achieve higher 3D sensing capabilities from error-prone omnidirectional stereo (*omnistereo*) vision exhibiting lack of construction and assembly precision.

Usually, a multi-view of the scene can be used to obtain dense or sparse depth maps from the overlapping region(s). We demonstrate the proposed model for omnistereo cameras using a *folded* configuration where a single-camera observes two hyperbolic mirrors at once. The characteristics and functionality of this particular rig is analyzed in [16]. One

of our latest prototypes is shown in Fig. 3. We are aware that a *folded* omnistereo system sacrifices spatial resolution on the imaging sensor (as analyzed in [28]), but it provides practical advantages such as reduced cost, size and weight, and truly-instantaneous stereo correspondences in dynamic environments.

A. Motivation for an Omnistereo Vision Model

At first, several methods for the calibration of a “monocular” ODVS have been proposed such as [10],[4],[8],[19],[7],[12],[25],[2], [26] and [31], and an extensive comparison of some of these techniques for the available implementations is given in [24]. However, only a few calibration methods for “omnistereo” catadioptric systems have been proposed, such as in [23],[21], [20],[6], [18], [29], [1], and [26]. Most of them assume a multi-camera configuration. For example, Lei et al. [18] use the distortion-based model [25] to calibrate each individual camera and then compute the essential matrix between them. We consider this as a “decoupled” approach to omnistereo calibration. Alternatively, for single-camera multi-axial configurations with spherical mirrors, the geometric model given by Agrawal et al. [2] employs the analytical forward projection (AFP) method in order to find the extrinsic parameters of an array of specular spheres (demonstrated by Taguchi et al. [29]). Although Agrawal and Ramalingam improved this calibration method in [1], it is still only applicable to array of spheres with a coplanarity constraint and requires the intrinsic parameters of the camera.

At the time of this writing, the only available implementation for omnistereo calibration is given by Schönbein et al. [26], which is applicable to quasi-central or slightly non-SVP systems. Their method introduces a centered model (an SPV approximation) conceived as an optimization post process based on the computed geometric model of the system (non-central calibration via AFP). It takes advantage of the disproportionate distance between the camera to the 3D scene points in relation to the deviation of each viewing ray (geometric) and the approximated center of projection (the centered camera model). This implementation requires knowledge of the mirror parameters specified by the ODVS manufacturer in order to initialize the geometric model.

Unfortunately, none of these methods have been applied to vertical and *folded* catadioptric systems, and most of them are isolated calibration routines of monocular models that usually perform suboptimally when the end goal of the system is to measure 3D space from triangulation. The lack of a generalized calibration method for the aforementioned

*This work is supported in part by U.S. Army Research Office grant No. W911NF-09-1-0565, U.S. National Science Foundation grants No. IIS-0644127 and No. CBET-1160046, Federal Highway Administration (FHWA) grant No. DTFH61-12-H-00002, and sponsorship by Ascending Technologies.

¹Carlos Jaramillo is with the Department of Computer Science, The Graduate Center of The City University of New York (CUNY), 365 Fifth Avenue, New York, NY, U.S.A. cjaramillo@gradcenter.cuny.edu

²Roberto G. Valenti and Jizhong Xiao are with the Department of Electrical Engineering, The City College of New York (CUNY), Convent Ave & 140th Street, New York, NY, U.S.A. rvalenti,jxiao@ccny.cuny.edu

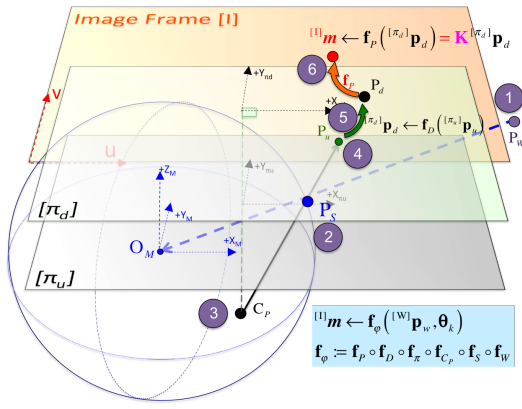


Fig. 1: Projection pipeline of the Generalized Unified Model (GUM) for the monocular case: (1) project point P_w towards the sphere's focus O_M , (2) normalize it as ${}^{[M]}P_S$, (3) change of coordinates with respect to $[C_P]$ such that ${}^{[C_P]}P_S$, (4) project onto normalized plane as ${}^{[\pi_u]}p_u$, (5) apply radial distortion to get ${}^{[\pi_d]}p_d$, and (6) transform to pixel ${}^{[I]}m$ in the image.

omnistereo systems motivates the following projection model we dub **GUMS** as it stands for **Generalized Unified Model for Stereo omnidirectional vision**. We use GUMS as the unified projection model that couples the omnistereo views in a holistic solution that is the basis for the calibration technique exemplified with both real and synthetic data. First, GUMS is delineated in Section II. In the attempt to generalize the several omnistereo configurations using GUMS, our preliminary work tackles the practical problem of calibrating vertical-folded omnistereo systems (Section III). When available, we validate our calibration results against ground-truth data acquired with a motion-capture system. We also employ simulated (ray-traced) imagery where ground-truth is inherent. We find the adequate GUMS parameters by employing a joint optimization approach, where a popular error metric is the overall 2D pixel error of the estimated control points (against ground-truth data), but we also measure 3D point differences between triangulated point correspondences and ground-truth poses (where applicable). These experimental results concerning calibration are presented in Section IV. In Section V, we demonstrate qualitative results of 3D dense point clouds generated via the calibrated systems.

II. THE PROJECTION MODEL

A. Overview of GUM for Monocular Views

The unifying sphere model was introduced in 2000 by Geyer and Daniilidis [11]. For central catadioptric systems, they theorized the existence of a unit sphere model for projection that is equivalent to the nonlinear analytical solution via a quadric reflector (or lens). Around the same time, Barreto and Araújo [4] presented similar mapping parameters onto a sphere model. In 2006, Mei and Rives [19], improved the unified sphere model by adding a distortion step (using radial and tangential distortion parameters) to the projection pipeline. An extension to Mei's sphere model was given by Xiang et al. [31] in 2013. in order to generalize the unified

model for non-central catadioptric cameras. They removed the axial constraint for the center of projection C_P as well as the unnecessary tangential distortion parameters. Relative to the unit sphere's center O_M , the free position for the projection point becomes ${}^{[M]}c_P = [\xi_X, \xi_Y, \xi_Z]^T$. Fig. 1 depicts the GUM projection process of point P_w as pixel ${}^{[I]}m$.

Since we are going to couple a pair of GUM's via a common frame of reference [C] (section II-B), we modify the original GUM by adding the pertaining translation ${}^{[M]}t$ of O_C with respect to O_M . Generally, θ_k is the vector of parameters for a GUM $_k$. In the vertically aligned configuration, it is convenient to add a co-linearity constraint for ${}^{[M]}t = [0, 0, t_z]^T$ along the Z_M -axis (Fig. 2), so that

$$\theta_k = \begin{bmatrix} t_{z,k} & \mathbf{x}_k \end{bmatrix}_{(1 \times 11)}, \text{ where } \mathbf{x}_k = \begin{bmatrix} \xi_k & \mathbf{d}_k & \mathbf{c}_k \end{bmatrix}_{(1 \times 10)}$$

with

$$\xi_k = [\xi_X, \xi_Y, \xi_Z]_k; \mathbf{d}_k = [k_{dist1}, k_{dist2}]_k; \mathbf{c}_k = [\alpha, \gamma_1, \gamma_2, u_c, v_c]_k$$

Assuming that coordinates of point ${}^{[M]}p_w$ are already given with respect to [M], the projection function \mathbf{f}_ϕ is the composition of various subroutines:

$${}^{[I]}m \leftarrow \mathbf{f}_\phi ({}^{[M]}p_w, \theta) := \mathbf{f}_p \circ \mathbf{f}_D \circ \mathbf{f}_\pi \circ \mathbf{f}_{C_P} \circ \mathbf{f}_S \quad (1)$$

In sum, the following steps are taken:

- 1) Given ${}^{[W]}p$, change its coordinates with respect to [M]

$${}^{[M]}p_w \leftarrow \mathbf{f}_W ({}^{[W]}p) := \frac{[M]}{[W]} \mathbf{T} p \quad (2)$$

Note that this is only possible if $\frac{[M]}{[W]} \mathbf{T}$ is known.

- 2) Normalize ${}^{[M]}p_w$ (onto the unit sphere) by

$${}^{[M]}p_S \leftarrow \mathbf{f}_S ({}^{[M]}p_w) := \frac{[M] p_w}{\| [M] p_w \|} \quad (3)$$

- 3) Change to coordinates with respect to the center of projection $[C_P]$

$${}^{[C_P]}p_S \leftarrow \mathbf{f}_{C_P} ({}^{[M]}p_S) := [M] p_S - [M] c_P = \begin{bmatrix} [M] x_S - \xi_X \\ [M] y_S - \xi_Y \\ [M] z_S - \xi_Z \end{bmatrix} \quad (4)$$

- 4) Project onto the undistorted normalized plane as $[\pi_u]$

$${}^{[\pi_u]}p_u \leftarrow \mathbf{f}_\pi ({}^{[C_P]}p_S) := \begin{bmatrix} [C_P] x_S \\ [C_P] z_S \\ [C_P] y_S \\ [C_P] z_S \end{bmatrix} \quad (5)$$

- 5) Apply radial distortion terms k_1, k_2 to ${}^{[\pi_u]}p_u = [x_u, y_u]^T$

$${}^{[\pi_d]}p_d \leftarrow \mathbf{f}_D ({}^{[\pi_u]}p_u) := [{}^{[\pi_u]}p_u + [{}^{[\pi_u]}p_u (k_1 \rho_u^2 + k_2 \rho_u^4)] \quad (6)$$

$$\text{where } \rho_u = \sqrt{x_u^2 + y_u^2} \quad (7)$$

- 6) Finally, we obtain the pixel point ${}^{[I]}m$ in the image via

$${}^{[I]}m \leftarrow \mathbf{f}_p ({}^{[\pi_d]}p_d) := \mathbf{K} [{}^{[\pi_d]}p_d, h] \quad (8)$$

$$\text{where } \mathbf{K} = \begin{bmatrix} \gamma_1 & \gamma_1 \alpha & u_c \\ 0 & \gamma_2 & v_c \\ 0 & 0 & 1 \end{bmatrix} \quad (9)$$

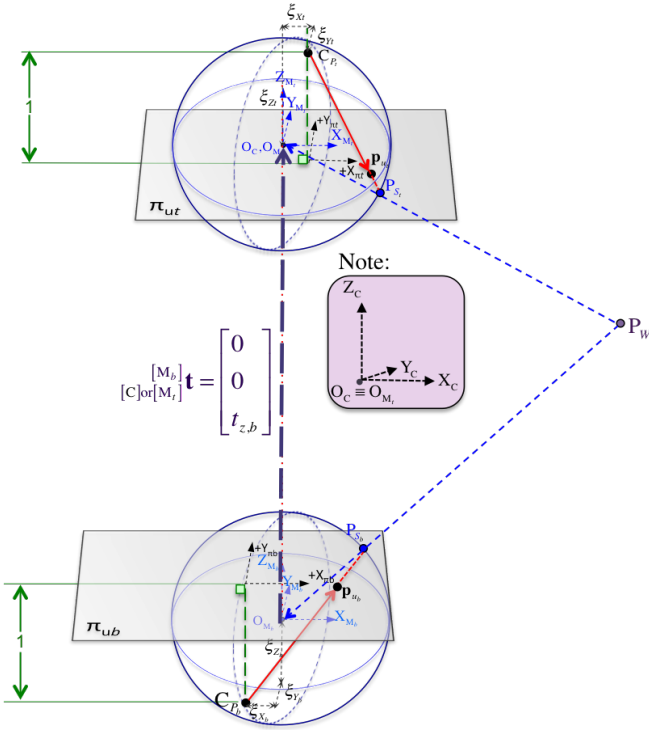


Fig. 2: The proposed GUMS with a coaxial constraint for the vertically-aligned omnidirectional configuration.

B. GUMS for OmniStereo Views

We build the proposed GUMS upon the generalized unified model for the monocular case (Section II-A). Simply put, GUMS links two models, GUM_k , via a common reference frame, $[C]$. Here, $k \in \{t, b\}$ is used to indicate the top or bottom models, respectively. Conveniently, we set $[C]$ coincident with $[M_t]$, so $t_{z,t} \leftarrow 0$. Here, we have a total of 21 model parameters:

$$\boldsymbol{\theta}_{GUMS} = \begin{bmatrix} t_{z,b}, & \mathbf{x}_t, & \mathbf{x}_b \end{bmatrix}_{(1 \times 21)} \quad (10)$$

NOTE: In the more general case, for independent cameras, we must add a parameter to account for the azimuthal misalignment between both image planes of the GUMS.

III. CALIBRATION

First, we introduce how calibration is performed in the case of a monocular system using a single GUM. Then in section III-B, we couple the pair of models as shown in section II-B. In both cases, we use an L number of calibration grids, each containing a chessboard pattern of N corner points. The pose of each calibration grid $[G]_g$ for $g \in \{1, \dots, L\}$ is

$$\mathbf{g}_g = \begin{bmatrix} [C] \\ [G_g] \end{bmatrix} \hat{\mathbf{q}}^T, \quad \begin{bmatrix} [C] \\ [G_g] \end{bmatrix} \mathbf{t}^T = \begin{bmatrix} [C] \\ [G_g] \end{bmatrix} [q_0, q_1, q_2, q_3, t_x, t_y, t_z] \quad (11)$$

where the normalized rotation quaternion is of the form

$$\hat{\mathbf{q}} := q_0 + q_1 \hat{\mathbf{i}} + q_2 \hat{\mathbf{j}} + q_3 \hat{\mathbf{k}} \quad \text{and} \quad \|\hat{\mathbf{q}}\| = 1$$

A. Monocular Calibration via GUM

The goal is to find a parameter vector

$$\mathbf{v} = \begin{bmatrix} \mathbf{g}_g, & t_z, & \mathbf{x} \end{bmatrix}_{(1 \times (11+7L))} \quad (12)$$

that minimizes the objective f_J , the scalar-valued function accumulating the square of the pixel residuals r_m :

$$\mathbf{v}_k^* = \arg \min_{\mathbf{v}} (f_J), \quad \text{where } f_J(\mathbf{v}) := \frac{1}{2} \sum_{g=1}^L \sum_{i=1}^N r_{m_{ig}}^2 \quad (13)$$

where

$$r_{m_{ig}} \leftarrow f_r(\tilde{\mathbf{m}}_{ig}, \mathbf{m}_{ig}) := \|\mathbf{r}_{m_{ig}}\| = \sqrt{(\tilde{u}_{ig} - u_{ig})^2 + (\tilde{v}_{ig} - v_{ig})^2} \quad (14)$$

Recall that \mathbf{m}_{ig} is the true (detected) image position of corner point i from pattern g , and $\tilde{\mathbf{m}}_{ig} \leftarrow \mathbf{f}_\varphi \left(\begin{smallmatrix} [C] \\ \tilde{\mathbf{p}}_{ig}, \tilde{\boldsymbol{\theta}} \end{smallmatrix} \right)$ where \mathbf{f}_φ projects $\tilde{\mathbf{p}}_{ig}$ as $\tilde{\mathbf{m}}_{ig}$ via the hypothesized parameters in $\tilde{\boldsymbol{\theta}}$.

B. Omnistereo Calibration via GUMS

As done in section III-A, we now need to find parameters

$$\mathbf{v}_{tb} = \begin{bmatrix} \mathbf{g}_g, & t_{z,b}, & \mathbf{x}_t, & \mathbf{x}_b \end{bmatrix}_{(1 \times (21+7L))} \quad (15)$$

minimizing the objective function $f_{J_{GUMS}}$ that accumulates the pixel residuals r_m computed by (14), but for both models

$$f_{J_{GUMS}}(\tilde{\mathbf{v}}_{tb}) := \frac{1}{2} \sum_{k=\{t,b\}} \sum_{g=1}^L \sum_{i=1}^N f_r(\tilde{\mathbf{m}}_{igk}, \mathbf{m}_{igk})^2 \quad (16)$$

For L grid patterns of N points, we have $2xNxL$ points on the image. We employ bound-constrained minimization algorithms from the *SciPy* module [17] such as TNC or L-BFGS-B (See [22] and [14] for extensive details). Instead of letting the solver estimate the gradient values numerically, the search can be vastly sped up by providing the gradient (Jacobian) of the objective function indicated as equation (16) with respect to the parameters in $\tilde{\mathbf{v}}_{tb}$. Recall that each point on the grid $^{[G_g]} \mathbf{p}_i$ is projected via \mathbf{f}_{φ_t} and \mathbf{f}_{φ_b} as estimated images points $\tilde{\mathbf{m}}_{igt}$ and $\tilde{\mathbf{m}}_{igb}$, respectively. Generally, without making $[C]$ coincide with a particular $[M_k]$, we get

$$\frac{\partial f_r}{\partial \mathbf{v}_k} = \begin{bmatrix} \frac{\partial f_r}{\partial \mathbf{f}_{\varphi_k}} & \frac{\partial f_r}{\partial \mathbf{v}_k} \end{bmatrix}_{(1 \times (17+7L))}$$

$\underbrace{\frac{\partial \mathbf{f}_{\varphi_k}}{\partial \mathbf{v}_k}}_{\text{Jacobian of } \mathbf{f}_\varphi} \quad (2 \times (11+7L))$

for $k \in \{t, b\}$, where

$$\frac{\partial f_r}{\partial \mathbf{f}_{\varphi_k}} = \begin{bmatrix} \frac{\partial f_r}{\partial \tilde{u}}, & \frac{\partial f_r}{\partial \tilde{v}} \end{bmatrix}_{(1 \times 2)} = \begin{bmatrix} \tilde{u} - u, & \tilde{v} - v \end{bmatrix}_{igk}$$

Pairwise, in each search iteration, each point $^{[C]} \tilde{\mathbf{p}}_{ig}$ provides:

$$\frac{\partial f_r}{\partial \mathbf{v}_{tb}} = \begin{bmatrix} \frac{\partial f_r}{\partial \mathbf{f}_{\varphi_t}} \left[\frac{\partial \mathbf{f}_{\varphi_t}}{\partial \mathbf{g}} & \frac{\partial \mathbf{f}_{\varphi_t}}{\partial t_{z,t}} & \mathbf{0}_{(2 \times 1)} & \frac{\partial \mathbf{f}_{\varphi_t}}{\partial \mathbf{x}_t} & \mathbf{0}_{(2 \times 10)} \right] \\ \frac{\partial f_r}{\partial \mathbf{f}_{\varphi_b}} \left[\frac{\partial \mathbf{f}_{\varphi_b}}{\partial \mathbf{g}} & \mathbf{0}_{(2 \times 1)} & \frac{\partial \mathbf{f}_{\varphi_b}}{\partial t_{z,b}} & \mathbf{0}_{(2 \times 10)} & \frac{\partial \mathbf{f}_{\varphi_b}}{\partial \mathbf{x}_b} \right] \end{bmatrix}$$

$(2 \times (7L+22))$
 \cup
 $(2 \times (7L+22))$

C. Parameter Initialization

A good initialization of parameter values at t_0 aids the convergence of the optimization process. Assuming a-priori knowledge of the theoretical mirror profiles, we can compute the unified model parameters $\mathbf{x}_k^{(t_0)}$ with the equations for the central case tabulated in [19]. In lack of such knowledge, our folded omnistereo rig could initialize each center of projection point to $\xi_t^{t_0} \leftarrow [0, 0, 1]$ and $\xi_b^{t_0} \leftarrow [0, 0, -1]$ without affecting the final convergence of the search as how is done with Mei's or Xiang's existing monocular implementations in MATLAB. We estimate each initial grid pose $\mathbf{g}_g^{(t_0)}$ as follows:

- 1) Back-project (lift) detected grid points ${}^{[k]} \{ \mathbf{m}_i \}_g$ from image to the theoretical quadric surface or initial GUM.
- 2) A virtual image of the grid pattern g is constructed by projecting these surface points orthographically to a plane π_g described by the horizontal and vertical vectors formed between the "extreme" corner points on the mirror surface. The optical axis' direction $\hat{\mathbf{z}}_{M_k}$ of a "virtual" pinhole camera coincident with O_{M_k} is normal to plane π_g . The versor $\hat{\mathbf{u}}_{M_k}$ is the aforementioned horizontal vector, while $\hat{\mathbf{v}}_{M_k} \leftarrow \hat{\mathbf{z}}_{M_k} \times \hat{\mathbf{u}}_{M_k}$. The focal length of the virtual camera is the orthographic distance between O_{M_k} and the plane π_g .
- 3) Since points on the virtual image plane π_g are given in [C], transform their coordinates onto $[M_k]$ by a rotation ${}^{[M_k]} \tilde{\mathbf{R}}$ using direction cosine matrix (DCM) and a translation ${}^{[M_k]} \tilde{\mathbf{t}}$.
- 4) Approximate the rigid transform ${}^{[M_k]} \mathbf{T}$ using either planar homography or PnP from the set of virtual points.
- 5) We put the grid's pose back to [C], so given ${}^{[C]} \mathbf{g}_k$ for $k \in \{t, b\}$, the average pose ${}^{[C]} \mathbf{g}^{(t_0)}$ is found. The orientation value is the result of a spherical linear interpolation (SLERP) between the approximated both poses.

D. Parameter Optimization - A Coupled Approach

In a "decoupled" approach, calibration is performed naively by parts: first, intrinsic parameters for each projection model (GUM $_k$) are found independently according to (13); then, the relative poses of the common frame [C] with respect to both model frames $\left({}^{[M_k]} \mathbf{T}, {}^{[M_b]} \mathbf{T} \right)$ are obtained via a third error-minimization process (but without affecting the intrinsic parameters previously found). On the other hand, our calibration is "coupled" because the whole optimization of equation (16) happens simultaneously. In Tables II and III, we provide a quantitative contrast between the coupled and decoupled approaches for GUMS calibration.

Regarding time complexity, the number of parameters plays an important factor during calibration. In an unconstrained and decoupled omnistereo calibration, the total number of parameters adds up to $2 * (7L + 17)$, where L is the number of poses of the calibration grid and each model has 17 parameters: 10 intrinsic (solved for first) + 7 due to ${}^{[M_k]} \mathbf{T}$ (solved for in the third optimization process). By coupling the optimization of both models such that GUMS: (GUM $_t$ | GUM $_b$), and by coaxially constraining all ${}^{[M_k]} \mathbf{T}$ as depicted in Fig. 2 with just

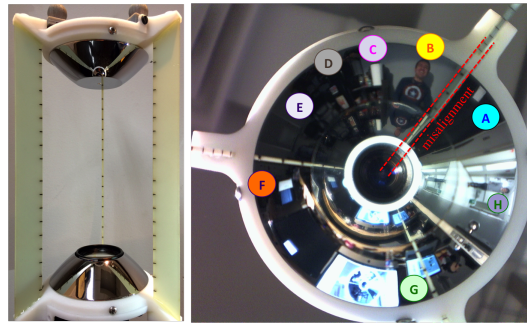


Fig. 3: Left: omnistereo rig prototype. Right: omnidirectional image captured by the real-life *far-sighted* prototype. Notice the annotated coaxial misalignment.

TABLE I: Rig Geometric Parameters (Theoretical Values)

Parameter	Dimension on Rig Type	
	Near-Sighted	Far-Sighted & Synthetic
r_{sys} [mm]	37.0	40.0
r_{ref} [mm]	17.23	19.0
r_{cam} [mm]	7	18.0
b [mm]	131.61	150.0
h_{sys} [mm]	150.0	176.6
α_{sys} [°]	66.8	40.9
α_{SROI} [°]	25.0	33.5

translations ${}^{[M_k]} \mathbf{t} = [0, 0, t_{z,k}]^T$, the number of parameters is reduced to $7L + (2 * 11)$ being solved within a single optimization process. Hence, the L grid poses \mathbf{g}_g defined in (11) are not recomputed.

IV. CALIBRATION EXPERIMENTS

The validation experiments shown in this manuscript are based on two folded catadioptric systems with hyperbolic mirrors like the real-life rig shown in Fig. 3. Table I provides the relevant system dimensions such as height h_{sys} , baseline b , total vertical field-of-view (vFOV) α_{sys} , and the overlapping vFOV α_{SROI} pertaining to the stereo capabilities of each rig.

A. Calibration Results

Fig. 6 and Fig. 7 exemplify variants for the simulated misalignment and the real far-sighted rig, respectively. The error analysis (RMSE) for some possible experimental variants are tabulated in Tables II and III. When compared against ground-truth (GT) data, these values agree with our claim that the decoupled approach can produce higher errors.

Fig. 5 allows us to visualize the 3D qualitative results after the calibration, where the estimated grid poses almost coincide with the ground-truth poses (from a motion-capture system). We also visualize the initial grid pose predictions (in blue) computed with the method outlined in Section III-C.

TABLE II: Error Analysis for Synthetic Experiments

Aligned	Coupled	2D Error [px]		3D Error from GT [mm]	
		GT	no GT	Triang.	[G] Pose
Yes	Yes	0.09	0.08	2.26	0.42
	No	1.82	0.64	28.06	4.64
No	Yes	5.70	0.10	23.52	20.59
	No	8.94	7.74	1098.8	20.68

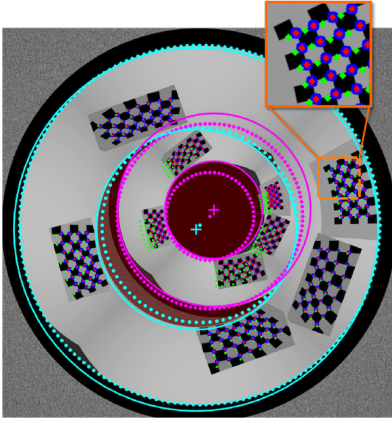


Fig. 4: Overlay of all point projections to image pixels for the misaligned synthetic experiment. Dotted circles and center points belong to the theoretical (uncalibrated) radial bounds. Solid circles and cross-hair centers relate to the calibrated results. Color annotations: center and boundaries of top mirror (cyan) and bottom mirror (magenta); projected grid points from ground-truth poses (green) and from estimated poses after calibration (red); detected grid corner pixels (blue).

TABLE III: Error Analysis for Real-Life Experiments

Sight	Coupled	2D Error [px]		3D Error from GT [mm]	
		GT	no GT	Triang.	[G] Pose
Near	Yes	4.86	2.56	6.06	3.66
	No	11.02	10.94	22.34	4.64
Far	Yes	NA	2.58	NA	NA
	No	NA	12.41	NA	NA

V. 3D SENSING FROM OMNISTEREO IMAGES

Stereo vision from images at distinct locations is a popular method for obtaining 3D range information via triangulation of point correspondences, which can be sought densely via block scanning [9] or sparsely via feature description [30]. Due to our model’s coaxial configuration, it is convenient to work on rectified panoramic images, where the search for correspondences can be performed using any of the methods available for perspective cameras. In [16], we demonstrate how these panoramic images are produced and how we triangulate correspondences to provide 3D information. Regardless of the matching technique employed vertically between panoramas $[\Xi_k]$, we resolve for $(^{[l]}m_i, ^{[b]}m_i)$ by the disparity map visualized at the top of Fig. 6a and Fig. 7a.

VI. DISCUSSION AND FUTURE WORK

In this work, we presented a tightly-coupled model (GUMS) for omnistereo vision systems. For now, we only demonstrated GUMS on folded catadioptric rig that we have custom-designed in [16]. Our focus here is to show that the “coupled” calibration using the proposed GUMS can provide more accurate results than the naive decoupled approach because we simultaneously optimize all the model parameters. We demonstrated the calibration accuracy of GUMS for both real and synthetic experiments, especially while dealing with coaxial misalignment that arises often in practice. In

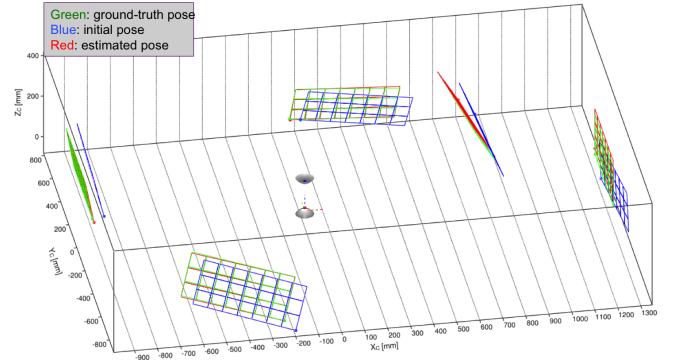
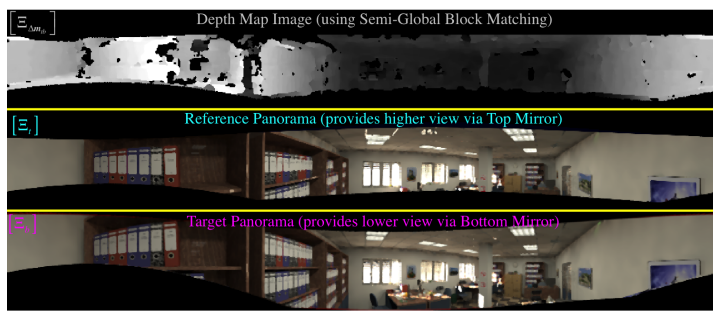


Fig. 5: Visualization of estimated 3D poses for the grids using the near-sighted omnistereo rig. Color annotations: ground-truth poses (green), estimated poses after calibration (red), initialized poses for calibration (blue).

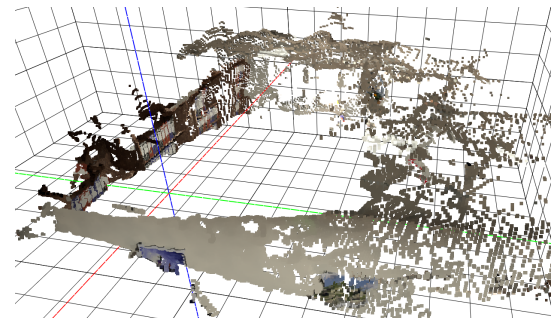
order to validate the precision of the real sensor, we used ground truth data obtained from a motion capture system. Our main contribution is not a direct application of a calibrated multiview omnidirectional sensor for 3D related fields such as visual odometry and structure-from-motion (known as SLAM in robotics), but we provide a necessary extension to the state-of-the-art generalized unified model (GUM) presented in Section II-A. A well calibrated omnistereo rig is more reliable as a 3D sensor for autonomous navigation, especially for robots such as MAVs that employ omnidirectional motion models and simultaneous observation of the 3D environment is preferable. The 3D point-clouds we have shown in this manuscript mainly intends to illustrate the triangulation results at a given omnistereo instance using the calibrated GUMS for those rigs. Our ultimate goal is to extend the GUMS model and the calibration toolkit to fit other configurations of omnidirectional systems. A Python implementation is available at the author’s public repository [15] and a video demonstration is provided in the supporting materials.

REFERENCES

- [1] Amit Agrawal and Srikumar Ramalingam. Single Image Calibration of Multi-axial Imaging Systems. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1399–1406. Ieee, jun 2013.
- [2] Amit Agrawal, Yuichi Taguchi, and Srikumar Ramalingam. Beyond Alhazen’s problem: Analytical projection model for non-central catadioptric cameras with quadric mirrors. In *Computer Vision and Pattern Recognition 2011*, 2011.
- [3] Simon Baker and Shree K. Nayar. A theory of single-viewpoint catadioptric image formation. *International Journal of Computer Vision*, 35(2):175–196, 1999.
- [4] Joao P. Barreto and Helder Araújo. Issues on the Geometry of Central Catadioptric Image Formation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. Published by the IEEE Computer Society, 2001.
- [5] Ryad Benosman and Sing B. Kang. *Panoramic vision : sensors, theory, and applications*. Springer, New York, New York, USA, 2001.
- [6] Stefano Cagnoni, Monica Mordonini, Luca Mussi, and Giovanni Adorni. Hybrid Stereo Sensor with Omnidirectional Vision Capabilities: Overview and Calibration Procedures. In *Image Analysis and Processing*, 2007.
- [7] Bernardo Cunha, José Azevedo, and Nuno Lau. Calibration of Non-SVP Hyperbolic Catadioptric Robotic Vision Systems. In *Tech: Robot Vision*, pages 307–324, 2006.

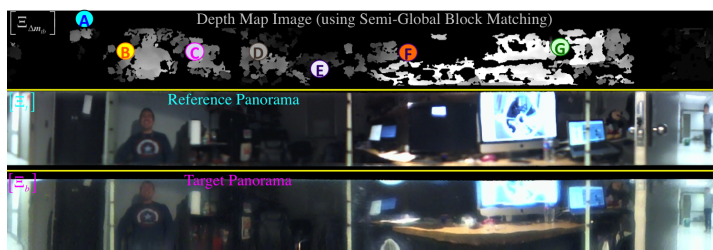


(a) Rectified panoramic images

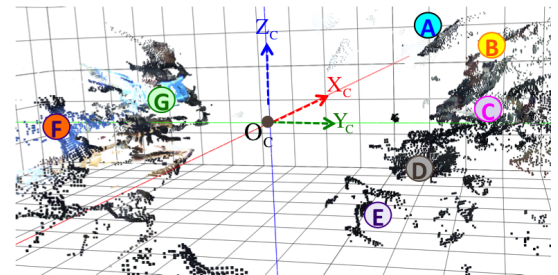


(b) Point Cloud

Fig. 6: Synthetic results. The circular image is obtained from a synthetic rig with extreme misalignment (Fig. 4).



(a) Rectified panoramic images



(b) Point Cloud

Fig. 7: Real-life rectified panoramas, disparity image, and point-cloud after calibration of the image shown in Fig. 3.

- [8] J. Fabrizio and J.-P. Tarel. Calibration of panoramic catadioptric sensors made easier. *Vision, 2002. Proceedings.*, pages 45–52, 2002.
- [9] Anders Forsgren, PE Gill, and MH Wright. Interior Methods for Nonlinear Optimization. *Society for Industrial and Applied Mathematics (SIAM Review)*, 44(4):525–597, 2002.
- [10] Christopher Geyer and Kostas Daniilidis. Catadioptric camera calibration. *Proceedings of the Seventh IEEE International Conference on Computer Vision*, pages 398–404 vol.1, 1999.
- [11] Christopher Geyer and Kostas Daniilidis. A unifying theory for central panoramic systems and practical implications. *European Conference on Computer Vision (ECCV)*, pages 445–461, 2000.
- [12] F. Huang, S.K. Wei, and R. Klette. Calibration of line-based panoramic cameras. In *Imaging Beyond the Pinhole Camera*, pages 55–84. Springer, 2006.
- [13] Hiroshi Ishiguro. Development of Low-Cost Compact Omnidirectional Vision Sensors and their applications. In *Proc. Int. Conf. Information systems, analysis ...*, 1998.
- [14] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, NY, USA, second edition, 2006.
- [15] C. Jaramillo. Omnistereo sensor design repository. https://github.com/ubuntuSlave/omnistereo_sensor_design [Online; accessed 2016-08-01].
- [16] Carlos Jaramillo, Roberto G Valenti, Ling Guo, and Jizhong Xiao. Design and Analysis of a Single-Camera Omnistereo Sensor for Quadrotor Micro Aerial Vehicles (MAVs). *Sensors*, 16(2):217, jan 2016.
- [17] Eric Jones, Travis Oliphant, Pearu Peterson, and Others. SciPy: Open source scientific tools for Python. <http://www.scipy.org/> [Online; accessed 2016-08-01].
- [18] Jie Lei, Xin Du, Yun-fang Zhu, and Ji-lin Liu. Unwrapping and stereo rectification for omnidirectional images. *Journal of Zhejiang University SCIENCE A*, 10(8):1125–1139, 2009.
- [19] Christopher Mei and Patrick Rives. Single View Point Omnidirectional Camera Calibration from Planar Grids. In *IEEE International Conference on Robotics and Automation (ICRA)*, number April, pages 3945–3950. Ieee, apr 2007.
- [20] Branislav Micusik. *Two-view geometry of omnidirectional cameras*. PhD thesis, 2004.
- [21] Branislav Micusik and Tomas Pajdla. Estimation of omnidirectional camera model from epipolar geometry. *Computer Vision and Pattern Recognition*, 1:1–485, 2003.
- [22] Stephen G Nash. Newton-Type Minimization via the Lanczos Method. *SIAM Journal on Numerical Analysis*, 21(4):770–788, 1984.
- [23] Yoshiro Negishi, Jun Miura, and Y. Shirai. Calibration of omnidirectional stereo for mobile robots. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, volume 3, pages 2600–2605. Ieee, 2004.
- [24] Luis Puig, J. Bermúdez, Peter Sturm, and J.J. Guerrero. Calibration of omnidirectional cameras in practice: A comparison of methods. *Computer Vision and Image Understanding*, 116(1):120–137, sep 2011.
- [25] Davide Scaramuzza, Agostino Martinelli, and Roland Siegwart. A Toolbox for Easily Calibrating Omnidirectional Cameras. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5695–5701. IEEE, oct 2006.
- [26] Miriam Schönbein, Tobias Strauss, and Andreas Geiger. Calibrating and Centering Quasi-Central Catadioptric Cameras. In *International Conference on Robotics and Automation (ICRA)*, 2014.
- [27] Peter Sturm, Srikumar Ramalingam, Jean-Philippe Tardif, Simone Gasparini, and Joao P. Barreto. Camera models and fundamental concepts used in geometric computer vision. *Foundations and Trends® in Computer Graphics and Vision*, 6(1-2):1–183, 2010.
- [28] Rahul Swaminathan, Michael D. Grossberg, and Shree K. Nayar. Non-Single Viewpoint Catadioptric Cameras: Geometry and Analysis. *International Journal of Computer Vision*, 66(3):211–229, 2006.
- [29] Yuichi Taguchi, Amit Agrawal, Ashok Veeraraghavan, Srikumar Ramalingam, and Ramesh Raskar. Axial-cones: modeling spherical catadioptric cameras for wide-angle light field rendering. In *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2010)*, volume 29, pages 172:1–172:8, 2010.
- [30] Tinne Tuytelaars Tuytelaars and Krystian Mikolajczyk. Local Invariant Feature Detectors- A Survey. *Foundations and Trends® in Computer Graphics and Vision*, 3(3):177–280, 2008.
- [31] Zhiyu Xiang, Xing Dai, and Xiaojin Gong. Noncentral catadioptric camera calibration using a generalized unified model. *Optics letters*, 38(9):1367–1369, 2013.
- [32] Yasushi Yagi. Omnidirectional sensing and its applications. *IEICE Transactions on Information and Systems*, 1(3):568–579, 1999.