Large-scale Model-Assisted Bundle Adjustment Using Gaussian Max-Mixtures

Paul Ozog and Ryan M. Eustice

***Abstract*— This paper reports on a model-assisted bundle adjustment framework in which visually-derived features are fused with an underlying three-dimensional (3D) mesh provided *a priori*. By using an approach inspired by the expectation-maximization (EM) class of algorithms, we introduce a hidden binary label for each visual feature that indicates if that feature is considered part of the nominal model, or if the feature corresponds to 3D structure that is absent from this model. Therefore, in addition to improved estimates of the feature locations, we can also label the features based on their deviation from the model. We show that this method is a special case of the Gaussian max-mixtures framework, which can be efficiently incorporated into state-of-the-art graph-based simultaneous localization and mapping (SLAM) solvers. We provide field tests taken from the Bluefin Robotics Hovering Autonomous Underwater Vehicle (HAUV) surveying the *SS Curtiss*.**

I. Introduction

Bundle adjustment (BA) is a special case of the simultaneous localization and mapping (SLAM) problem; it is an estimation problem whose unknowns consist of camera poses and the positions of visual features. This is a widespread technique used throughout computer vision and mobile robotics, due mainly to the low cost and high reconstruction quality of digital cameras [1–3].

A major drawback in the use of optical cameras in robotic perception is their susceptibility to environmental noise and poor lighting conditions. Researchers have previously proposed modifications to BA that leverage three-dimensional (3D) models of the scene (provided *a priori*) to mitigate these challenges. This practice is sometimes referred to as *model-asissted bundle adjustment*. The reconstruction of human faces has been a particularly prevalent application domain, however these techniques have certain shortcomings that are ill-suited for their application in large-scale robotic surveillance. For instance, mobile robots typically survey areas that are much larger than themselves, unlike the relative sizes of a camera and human faces. In addition, the images captured by robots operating in the field will likely contain 3D structure that is not accounted for in the prior model.

Using underwater optical imaging for 3D reconstruction is extremely challenging and would benefit from model-assisted methods. In particular, back-scatter is a well-known issue that researchers must consider when deploying autonomous underwater vehicles (AUVs) that are equipped with optical cameras [4, 5]. Despite these challenges, the benefits of model-assisted BA have yet to be explored in a large scale underwater setting. In addition, the previously employed facial reconstruction methods do not easily transition to this domain. The focus of this paper is therefore to introduce a large scale model-assisted BA framework and evaluate it in the challenging domain of *in situ* underwater ship hull inspection using optical cameras.

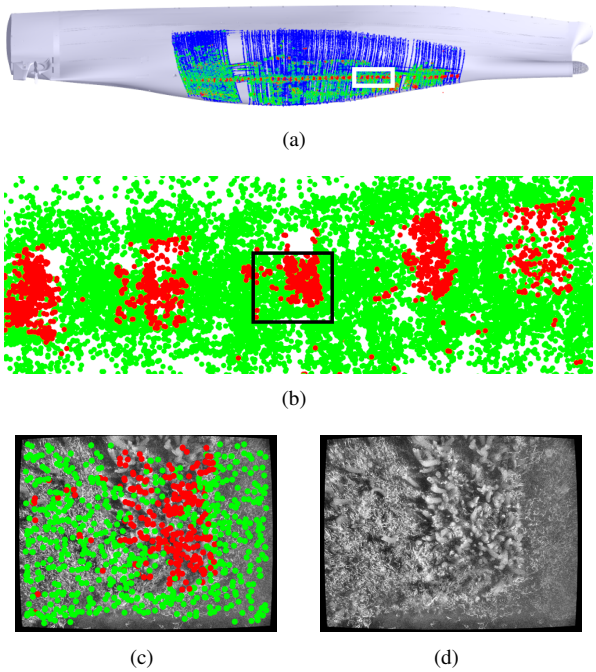


Fig. 1. (a) DVL ranges, shown in blue, allow us to localize to the prior model of the ship being inspected, shown in gray. In (b) and (c), visual features that are hypothesized to lie on the nominal surface of the prior model are shown in green. Features that correspond to 3D structure that is absent from the model are shown in red. In this example, red features correspond to biogrowth emanating from docking blocks along the hull's centerline, in (d).

For this paper, we leverage a large-scale 3D computer aided design (CAD) model of the ship that is then autonomously surveyed with a camera-equipped robot, as shown in Fig. 1. The contributions of the work are as follows:

- We propose an expectation-maximization (EM) algorithm that assigns hard binary labels to each visual feature and solves for the optimal 3D locations of cameras and features accordingly. This approach is therefore capable of identifying 3D structure that is absent from the prior model.

*This work was supported in part by the Office of Naval Research under award N00014-12-1-0092, and in part by the American Bureau of Shipping under award number N016970-UM-RCMOP.

P. Ozog is with the Department of Electrical Engineering & Computer Science, University of Michigan, Ann Arbor, MI 48109, USA paulozog@umich.edu.

R. Eustice is with the Department of Naval Architecture & Marine Engineering, University of Michigan, Ann Arbor, MI 48109, USA eustice@umich.edu.

- We show that this algorithm is a special case of the Gaussian max-mixture framework, which was originally intended for robust least-squares optimization in graph-based SLAM [6].
- To our best knowledge, this is the largest real-world model-assisted BA evaluation, both in physical scale and number of images.

II. Related Work

Model-assisted visual reconstruction methods were particularly popular during the late 1990's and early 2000's, especially in the domain of human face reconstruction [7–11]. Works by Fua [7] and Kang and Jones [8] are similar to traditional BA: a least-squares minimization over reprojection error. However, they introduce regularization terms that essentially enforce triangulated points lying close to the model's surface. Shan et al. [9] introduce an optimization problem over a set of model parameters—rather than a regularization over features—that allow the generic face model to more closely match the geometry of the subject's face. Fidaleo and Medioni [11] noted that these methods are rarely able to integrate 3D structure present in the subject that is absent from the model (such as facial hair and piercings). Instead, their approach used a prior model strictly for pose estimation, but the reconstruction of the face was entirely data-driven.

The primary application domain of these methods is in the reconstruction of human faces, however they have largely been overshadowed by modern, highly accurate, dense reconstruction methods that use either commodity depth cameras [12, 13] or patch-based multiview stereopsis using high-quality imagery [14]. These more recent methods have shown impressive reconstructions of both small-scale objects (human faces), and large scale objects (indoor environments and outdoor structures).

Recently, however, model-assisted methods have seen some re-emergence in particularly challenging areas of mobile robotics, such as the work by Geva et al. [15] in which an unmanned aerial vehicle (UAV) surveys a remote area. They used digital terrain models (DTMs) to regularize the position of 3D features observed from the camera mounted on the UAV, in a very similar fashion to the work in [7, 8]. These DTMs are freely available from the Shuttle Radar Topography project [16], and act as the prior model used in their approach. This approach is most similar to ours, however we differentiate our approach in three important ways: *(i)* our approach is capable of incorporating visual information that is absent from the nominal *a priori* model by assigning a hidden binary random variable for each visual feature; *(ii)* we use an orthogonal signed distance, rather than raycasting, to evaluate a feature's surface constraint likelihood; and *(iii)* we evaluate our approach on a dataset with several orders of magnitude more bundle-adjusted keyframes.

III. Notation

We denote the set of all unknowns, $\mathbf{X}$, as consisting of N_p poses, the relative transformation to the model frame, and N_l landmarks,

$$\mathbf{X} = \{\underbrace{\mathbf{x}_{g1} \dots \mathbf{x}_{gN_p}}_{\text{robot poses}}, \underbrace{\mathbf{x}_{g\mathcal{M}}}_{\text{model pose}}, \underbrace{\mathbf{l}_1 \dots \mathbf{l}_{N_l}}_{\text{visual landmarks (features)}}\},$$

where $\mathbf{x}_{ij}$ denotes the 6-degree of freedom (DOF) relative pose between frames i and j. The common, or global frame, is denoted as g. Visually-derived features, denoted as $\mathbf{l}_i$, are the 3D positions of features as expressed in the global frame. Finally, $\mathcal{M}$ denotes a triangular mesh consisting of a set of vertices, edges between vertices, and triangular faces.

Note that $\mathbf{X}$ may consist of additional variables, such as extrinsic parameters of the robot sensors. We leave these values out for the sake of clarity.

Let $\mathbf{Z}$ denote the set of all measurements, which consists of all odometry measurements, priors, surface range measurements (e.g., from an active range scanner), visual feature detections, and surface constraints (which will be described in §IV-B),

$$\mathbf{Z} = \{\mathcal{Z}_{\text{odo}}, \mathcal{Z}_{\text{prior}}, \mathcal{Z}_{\text{range}}, \mathcal{Z}_{\text{feat}}, \mathcal{Z}_{\text{surf}}\}.$$

We assume all measurements except $\mathcal{Z}_{\text{surf}}$ are independently corrupted by zero-mean Gaussian noise, so therefore the distributions of these observations given $\mathbf{X}$ are conditionally Gaussian. Note that our approach is applicable even if $\mathcal{Z}_{\text{odo}}$, $\mathcal{Z}_{\text{prior}}$, and $\mathcal{Z}_{\text{range}}$ are not available, however we include them due their necessity in underwater ship hull inspection.

We assign a hidden binary latent variable to each visual feature,

$$\mathbf{\Lambda} = \{\lambda_1 \dots \lambda_{N_l}\}, \lambda_i \in \{0, 1\},$$

where a value of one encodes that a visually-derived feature lies on the nominal surface of the prior model. A value of zero encodes that the visually-derived feature corresponds to physical structural that is absent from the prior model.

IV. Approach

A. Formulation as Expectation-Maximization

The goal of our work is to estimate $\mathbf{X}$ using a simplified variant of the EM algorithm, known as *hard EM* [**?**]:

1) Initialize $\mathbf{X}$
2) Repeat the following until $p(\mathbf{Z}, \mathbf{\Lambda}|\mathbf{X})$ converges:
   a) $\mathbf{\Lambda}^* = \underset{\mathbf{\Lambda}}{\operatorname{argmax}}\, p(\mathbf{Z}, \mathbf{\Lambda}|\mathbf{X})$
   b) $\mathbf{X}^* = \underset{\mathbf{X}}{\operatorname{argmax}}\, p(\mathbf{Z}, \mathbf{\Lambda}^*|\mathbf{X})$

Similar to previous work, we introduce a set of prior measurements, $\mathcal{Z}_{\text{surf}}$, that regularize the positions of 3D visual features so that they lie on the surface of $\mathcal{M}$. We expand the likelihood function using Bayes' rule and note that the odometry, prior, and feature detection observations are independent of the feature labels (and conditionally independent of each other):

$$\begin{aligned} p(\mathbf{Z}, \mathbf{\Lambda}|\mathbf{X}) &= p(\mathbf{Z}|\mathbf{\Lambda}, \mathbf{X}) p(\mathbf{\Lambda}|\mathbf{X}) \\ &= p(\mathcal{Z}_{\text{odo}}, \mathcal{Z}_{\text{prior}}, \mathcal{Z}_{\text{range}}, \mathcal{Z}_{\text{feat}}|\mathbf{X}) p(\mathcal{Z}_{\text{surf}}|\mathbf{\Lambda}, \mathbf{X}) p(\mathbf{\Lambda}|\mathbf{X}). \quad (1) \end{aligned}$$

If we assume that $p(\lambda_i|\mathbf{X})$ is uninformative (i.e., labels are equally likely to lie on or off the surface), then we

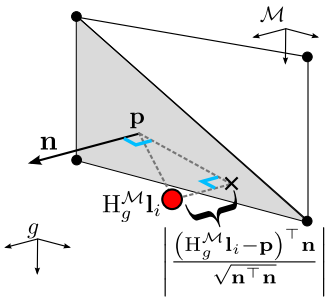


Fig. 2. Overview of the surface constraint using a simple triangular mesh $\mathcal{M}$ consisting of two triangles. The constraint converts the distance to the closest face, d_{s_i}, to a signed distance (depending on if the feature is inside or outside the triangular face).

can express the likelihood as proportional to a simpler expression:

$$p(\mathbf{Z}, \mathbf{\Lambda}|\mathbf{X}) \propto p(\mathcal{Z}_{\text{odo}}, \mathcal{Z}_{\text{prior}}, \mathcal{Z}_{\text{range}}, \mathcal{Z}_{\text{feat}}|\mathbf{X}) p(\mathcal{Z}_{\text{surf}}|\mathbf{\Lambda}, \mathbf{X})$$

Therefore, step (2a) in the hard EM algorithm simplifies to

$$\underset{\mathbf{\Lambda}}{\operatorname{argmax}}\, p(\mathbf{Z}, \mathbf{\Lambda}|\mathbf{X}) = \underset{\mathbf{\Lambda}}{\operatorname{argmax}}\, p(\mathcal{Z}_{\text{surf}}|\mathbf{\Lambda}, \mathbf{X}), \tag{2}$$

where $p(\mathcal{Z}_{\text{surf}}|\mathbf{\Lambda}, \mathbf{X})$ is described in §IV-B. In addition, step (2b) simplifies to

$$\begin{aligned}&\underset{\mathbf{X}}{\operatorname{argmax}}\, p(\mathbf{Z}, \mathbf{\Lambda}|\mathbf{X}) = \\ &\underset{\mathbf{X}}{\operatorname{argmax}}\, p(\mathcal{Z}_{\text{odo}}, \mathcal{Z}_{\text{prior}}, \mathcal{Z}_{\text{range}}, \mathcal{Z}_{\text{feat}}|\mathbf{X}) p(\mathcal{Z}_{\text{surf}}|\mathbf{\Lambda}, \mathbf{X}),\end{aligned} \tag{3}$$

which is equivalent to a least-squares optimization problem when the measurements are corrupted by additive Gaussian noise.

B. Modeling the Surface Constraint

Consider the set of all surface constraints $\mathcal{Z}_{\text{surf}} = \{z_{s_1} \dots z_{s_{N_l}}\}$. We model the conditional distribution of these constraints as follows:

$$p(z_{s_i}|\lambda_i, \mathbf{X}) = \begin{cases} \mathcal{N}(h\left(\mathbf{x}_{g\mathcal{M}}, \mathbf{l}_i\right), \sigma_0^2), & \lambda_i = 0 \\ \mathcal{N}\left(h\left(\mathbf{x}_{g\mathcal{M}}, \mathbf{l}_i\right), \sigma_1^2\right), & \lambda_i = 1 \end{cases}, \tag{4}$$

where $h(\cdot)$ computes the orthogonal signed distance of the i^{th} feature to the model. The values σ_0^2 and σ_1^2 denote the variance of the surface constraint when λ_i is 0 or 1, respectively. Intuitively, these variances are chosen such that $\sigma_1^2 \ll \sigma_0^2$, i.e., features that lie close to the model surface are more tightly pulled toward it, while features that lie away from the model are free to vary with approximately zero cost. To constrain visual features to lie on the surface, we assign $z_{s_i} = 0$ for all of the features. If we desired the surfaces to tend toward lying inside or outside the surface by distance d, we would assign z_{s_i} to $-d$ or d, respectively.

The orthogonal signed distance function $h(\cdot)$ is a nonlinear function of the pose of the model and the position of the visual feature:

$$h\left(\mathbf{x}_{g\mathcal{M}}, \mathbf{l}_i\right) = \frac{\left(\mathrm{H}_{\mathcal{M}}^{g}\mathbf{l}_i - \mathbf{p}\right)^\top \mathbf{n}}{\sqrt{\mathbf{n}^\top\mathbf{n}}},$$

where $\mathrm{H}_{\mathcal{M}}^{g}$ is an affine transformation matrix that transforms points in the global frame to points in the model frame. Intuitively, $h(\cdot)$ returns the orthogonal signed distance of a visual feature $\mathbf{l}_i$ to the surface of the closest triangular face in $\mathcal{M}$. This triangle is characterized by any point, $\mathbf{p}$, that lies on the surface of the triangle, and its surface normal, $\mathbf{n}$. This calculation is illustrated in Fig. 2.

C. Relation to Gaussian Max-Mixtures

In this section, we show how the previous formulation is a special case of Gaussian max-mixtures framework proposed by Olson and Agarwal [6]. This was mainly introduced in the area of robust SLAM backends as a probabilistically-motivated approach to rejecting incorrect loop closures [6, 17**?** , 18] and detecting wheel slippage in ground robots [6]. More recently, it has been applied in learning robust models for consumer-grade global positioning system (GPS) measurements that can reject outliers [19].

First, we note that the surface constraint likelihood $p(z_{s_i}|\mathbf{X})$ is a Gaussian sum-mixture due to marginalizing λ_i and therefore not Gaussian. Even so, we can write the conditional distribution of the unknowns given the measurements as

$$\log p(\mathbf{X}|\mathbf{Z}) \propto \log \prod_i p(z_i|\mathbf{X}), \tag{5}$$

where z_i denotes the i^{th} measurement; either an odometry, prior, range, feature, or surface constraint. By maximizing this distribution, we arrive at a maximum *a posteriori* (MAP) estimate for $\mathbf{X}$, as shown by [20].

Though the labels, $\mathbf{\Lambda}$, are absent from (5), we can approximate the *sum-mixture* surface constraint likelihood using the similar *max-mixture* distribution proposed by Olson and Agarwal [6]. The likelihood then takes the form

$$p(z_{s_i}|\mathbf{X}) = \eta \max_{\lambda_i} p(z_{s_i}|\lambda_i, \mathbf{X}). \tag{6}$$

The logarithm can be brought inside the product from (5), and again inside the max operator from (6). This distribution can therefore be thought of as a binary Gaussian max-mixture with equal weights for each component of the mixture.

This conditional distribution essentially combines steps (2a) and (2b) from the hard EM algorithm so that the labels are determined whenever the likelihood term is evaluated. The distribution from (6) is therefore equivalent to a binary max-mixture of Gaussians with equal weights. This conforms to our earlier formulation from §IV-A that assigns equal prior probability to a surface lying on or off the mesh's surface. The only two parameters used in our approach are therefore σ_0^2 and σ_1^2 from (4). We illustrate this distribution in Fig. 3 using values that are representative of the structure that we typically observe on ship hulls.

Note that the distribution from (6) contains an unknown normalization constant, η, that ensures a valid probability distribution. However, for the purposes of maximizing the likelihood, computing the specific value of this scale factor is not necessary [6]. Additionally, we represent the distribution from (5) using a factor graph [20], as shown in Fig. 4. To solve the corresponding least-squares problem, we use the freely-available Ceres library [21].

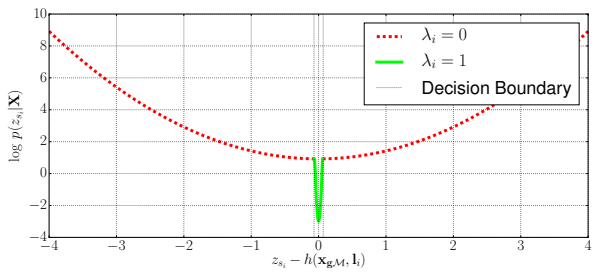


Fig. 3. Decision boundary for $\sigma_0 = 1$ m, $\sigma_1 = 0.12$ m overlayed on the log probability (i.e., cost computed during optimization). If desired, features could be biased toward the "inside" or "outside" of the model by assigning a nonzero value to z_{s_i} to shift this curve left and right, respectively. For our experiments, however, we assign $z_{s_i} = 0$ for all features.

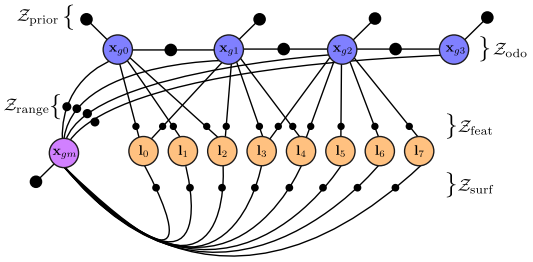
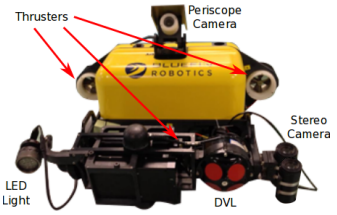


Fig. 4. Representation of our method as a factor graph. The factor nodes denoted with $\mathcal{Z}_{\text{surf}}$ denote the surface constraints, which represent binary Gaussian max-mixtures distributions from §IV-C. These factors constrain the pose of the prior model $\mathbf{x}_{gm}$ and the location of visual features $\mathbf{l}_i$.



(a)

(b)

Ship Length	183 m
Ship Beam	27 m
Ship Draft	9.1 m
AUV trajectory length	0.963 km
Number of images	44,868
Number of DVL raycasts	96,944
Number of feature reprojections	974,144
Number of features	243,536

(c)

Fig. 5. The HAUV sensor payload is shown in (a). The vessel being surveyed is the *SS Curtiss*, for which we have access to a CAD-derived 3D mesh. The size of the dataset used in this paper is summarized in (c).

D. Localizing to the prior model

Our approach, like all model-assisted BA frameworks, requires a good initial guess of the alignment between the camera trajectory and the prior model. This is typically done using triangulated features from the camera imagery, but for autonomous ship hull inspection there are many portions of the ship where no features can be detected because the hull is not visually salient [22, 23].

In our case, the underwater robot observed sparse range measurements (that is, $\mathcal{Z}_{\text{range}}$) using a Doppler velocity log (DVL). These range returns are rigidly aligned to the prior model using generalized iterative closest point (GICP) [24], which serves as an initial guess (i.e., the prior factor connected to node $\mathbf{x}_{gm}$ in Fig. 4). Individual poses can be further optimized using raycasting techniques to compute the likelihood of $\mathcal{Z}_{\text{range}}$ and the surface constraint measurements $\mathcal{Z}_{\text{surf}}$ from §IV-B. Note that dense connectivity to the variable node representing $\mathbf{x}_{gm}$ from Fig. 4. If this quantity was assumed known (an unrealistic presumption for ship hull inspection), the factors corresponding to $\mathcal{Z}_{\text{range}}$ and $\mathcal{Z}_{\text{surf}}$ would be unary (instead of binary) and the graph would not have any dense connectivity.

V. Results

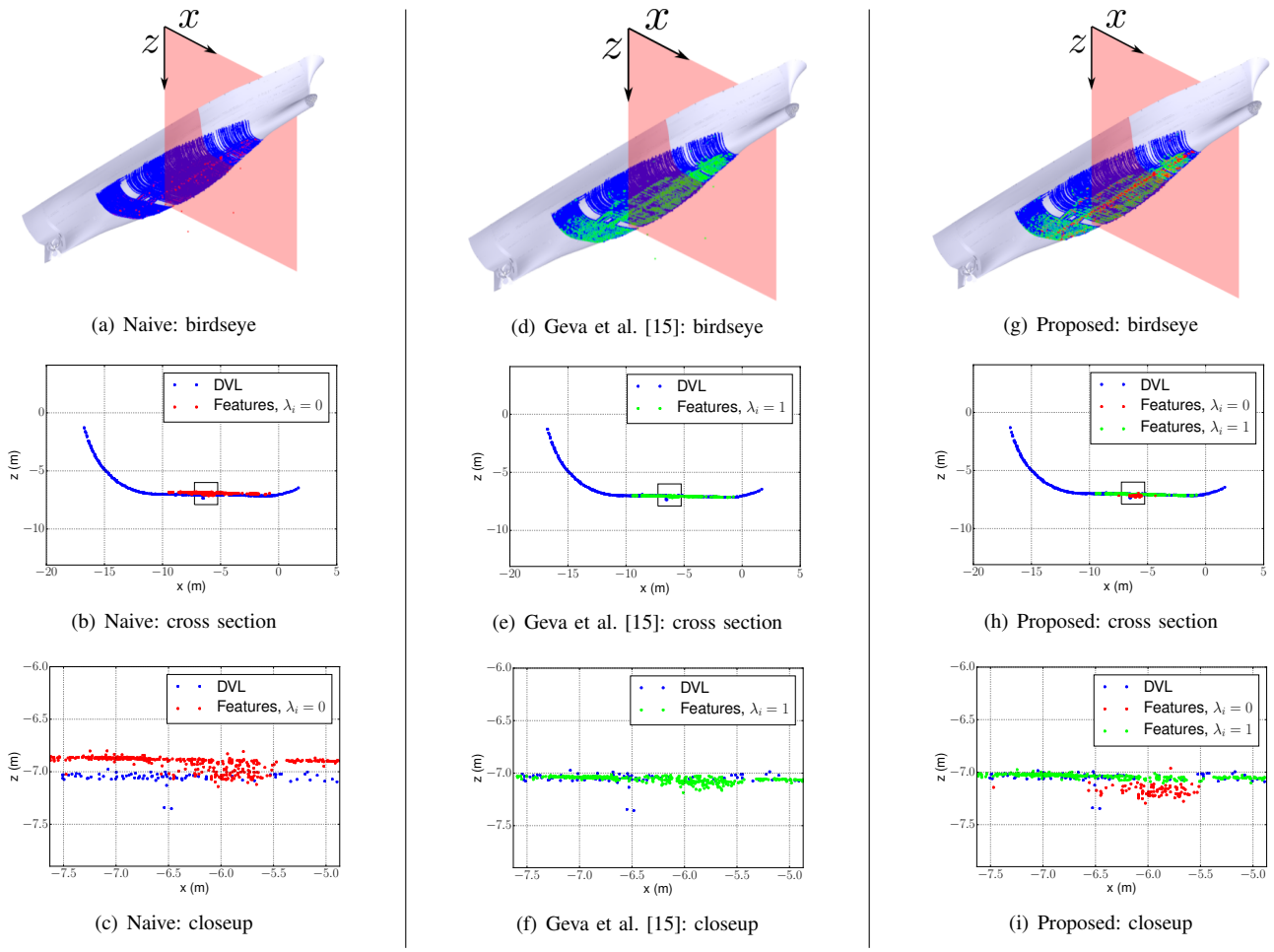
The field data used in our experimental evaluation is taken from the Bluefin Robotics Hovering Autonomous Underwater Vehicle (HAUV) surveying the *SS Curtiss*, shown in Fig. 5. A 3D triangular mesh was derived from CAD drawings, and serves as the prior model in our model-assisted framework. In this section, we evaluate the performance of three approaches when processing this single large dataset:

1) A naive BA framework where the measurements consists of $\mathcal{Z}_{\text{prior}}$, $\mathcal{Z}_{\text{odo}}$, $\mathcal{Z}_{\text{range}}$, and $\mathcal{Z}_{\text{feat}}$. All surface constraints are disabled, i.e., $\lambda_i = 0$ for every feature.
2) The approach based on Geva et al. [15], which consists of the measurements $\mathcal{Z}_{\text{prior}}$, $\mathcal{Z}_{\text{odo}}$, $\mathcal{Z}_{\text{range}}$, and $\mathcal{Z}_{\text{feat}}$, in addition to surface constraints, $\mathcal{Z}_{\text{surf}}$, such that $\lambda_i = 1$ for every feature.
3) The proposed algorithm discussed in §IV-A, implemented using Gaussian max-mixtures, where each hidden label λ_i is assigned from (6).

We used the scale invariant feature transform (SIFT) feature descriptor [25] to assign visual correspondence. The size of the bundle adjustment problem is shown in Table 5(c).

A. 3D Reconstruction Evaluation

We provide a visualization of the reconstruction that highlights the advantages of our approach in Fig. 6. These plots show cross sections of the ship hull, from starboard-to-port, to highlight the relevant portions of the visual features and range returns from the DVL. In these figures, we see some general trends. 1) In the reconstruction derived from the naive approach, the visual features do not lie on the same surface as the range returns from the DVL. The features are underconstrained in the naive case because there is zero information relating the pose of the prior model to the position of the visual features. 2) Using the approach from Geva et al. [15], the visual features lie on the same surface as the DVL range returns, as expected. Because *all* features



(a) Naive: birdseye (d) Geva et al. [15]: birdseye (g) Proposed: birdseye

(b) Naive: cross section (e) Geva et al. [15]: cross section (h) Proposed: cross section

(c) Naive: closeup (f) Geva et al. [15]: closeup (i) Proposed: closeup

Fig. 6. Each column represents a different method, with each column showing the same representative cross section of the reconstruction. For the naive approach shown in (a) through (c), there is zero information between the visual features and prior model resulting in an obvious misregistration with the DVL. Using the method from Geva et al. [15] shown in (d) through (f), the features and DVL ranges are well-registered, but the docking block is visibly "squished" into the surface. Our method, shown in (g) through (i), combines the favorable qualities of each method, aligning the visual features and DVL returns while also preserving 3D structure that is absent from the prior model.

are constrainted to lie on the surface, the algorithm does not capture 3D structure present on the actual ship hull that is not present in the prior model (e.g., the docking blocks along the bottom of the hull). 3) Our approach combines the benefits of both approaches: the visual features and DVL-derived point cloud lie on the same surface, and our visual reconstruction yields 3D structure that would have been heavily regularized using the approach from Geva et al. [15].

In addition, the camera used in this dataset is a calibrated underwater stereo rig, so we use a stereo-derived depth image of a docking block shown in Fig. 7 as an indication that the 3D structure preserved in Fig. 6(i) is correct. Indeed, the 3D structure inferred from our proposed method is 0.10 m to 0.20 m closer to the camera than the rest of the scene. This agrees with the depth image from Fig. 7.

Finally, we provide results that suggest the identification of feature labels stabilizes after about ten iterations, as shown in Fig. 8. Clearly, for this dataset, the vast majority of features lie on the prior model, suggesting that the weights in each mixture (or, equivalently, the last multiplicand in (1)) can be optionally tuned to reflect this trend, rather than conservatively assigning equal weights.

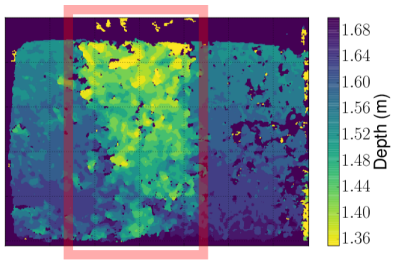


Fig. 7. A dense stereo matching algorithm shows definite 3D structure around the area of the hull highlighted in the bottom row of Fig. 6. Using our proposed method, the object highlighted in the pink box is preserved in the sparse reconstruction, shown in Fig. 6(i).

B. Computational Performance

We assessed the computational performance by performing timed trials on a consumer-grade four-core 3.30 GHz processor. We report the timing results of each of the three methods in Fig. 9. From this plot we draw two conclusions. 1) The computational costs of our method impose total performance loss of 22.3% compared to the naive approach (511 seconds versus 418 seconds). 2) The computational costs of the approach from Geva et al. [15] imposes a performance

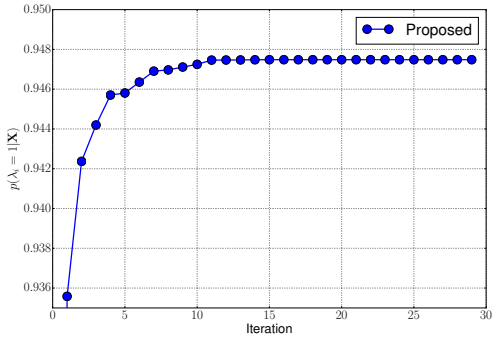
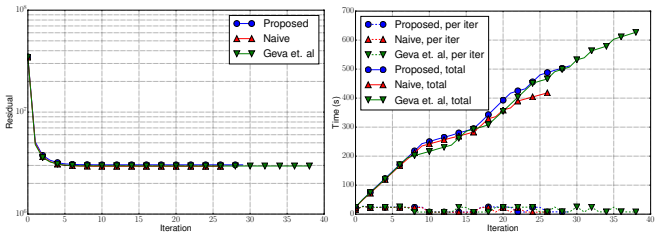


Fig. 8. For this dataset, the relative frequency of all features with $\lambda_i = 1$ as a function of the current least-squares iteration stabilizes after about ten iterations. Note that the first iteration, the percentage is 82.4%, however this data point is omitted to highlight the subtle changes in subsequent iterations.



(a) Cost vs. iteration (b) Timing results

Fig. 9. Cost (a) and timing (b) comparison for each method evaluated in §V. Our algorithm imposes some extra computation time compared to the naive approach, but is still competitive. The naive, Geva et al. [15], and proposed approaches converge in 26, 38, and 29 iterations, respectively.

loss of 50.0% compared to the naive approach. The latter case is a result of the optimizer performing more iterations until convergence. Intuitively, by forcing visual features that protrude from the prior model to lie flush, the optimizer must perform more iterations to satisfy the corresponding reprojection errors. Even though our method devotes additional processing time when evaluating (6), this is overcome by the added cost performing additional iterations.

VI. Conclusion

We proposed a model-assisted bundle adjustment framework that assigns binary labels to each visual feature. Using an EM algorithm with hard hidden variable assignments, we iteratively update these variables along with the current state estimate. We show that this algorithm is a special case of the Gaussian max-mixtures framework from earlier work in robust pose graph optimization. We compared our approach to recent work in model-assisted methods, and showed our algorithm has favorable properties when evaluated in the context of autonomous ship hull inspection.

References

[1] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski, "Building Rome in a day," *Comm. of the ACM*, vol. 54, no. 10, pp. 105–112, Oct. 2011.

[2] M. Bryson, M. Johnson-Roberson, O. Pizarro, and S. Williams, "Colour-consistent structure-from-motion models using underwater imagery," in *Proc. Robot.: Sci. & Syst. Conf.*, Sydney, Australia, Jul. 2012.

[3] S. Daftry, C. Hoppe, and H. Bischof, "Building with drones: Accurate 3D facade reconstruction using MAVs," in *Proc. IEEE Int. Conf. Robot. and Automation*, Seattle, WA, USA, May 2015, pp. 3487–3494.

[4] F. Aguirre, J. Boucher, and J. Jacq, "Underwater navigation by video sequence analysis," in *Proc. Int. Conf. Pattern Recog.*, vol. 2, Atlantic City, NJ, USA, Jun. 1990, pp. 537–539.

[5] R. Campos, R. Garcia, P. Alliez, and M. Yvinec, "A surface reconstruction method for in-detail underwater 3D optical mapping," *Int. J. Robot. Res.*, vol. 34, no. 1, pp. 64–89, 2015.

[6] E. Olson and P. Agarwal, "Inference on networks of mixtures for robust robot mapping," *Int. J. Robot. Res.*, vol. 32, no. 7, pp. 826–840, 2013.

[7] P. Fua, "Using model-driven bundle-adjustment to model heads from raw video sequences," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 1, Kerkyra, Greece, Sep. 1999, pp. 46–53.

[8] S. B. Kang and M. Jones, "Appearance-based structure from motion using linear classes of 3D models," *Int. J. Comput. Vis.*, vol. 49, no. 1, pp. 5–22, 2002.

[9] Y. Shan, Z. Liu, and Z. Zhang, "Model-based bundle adjustment with application to face modeling," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2, Vancouver, Canada, Jul. 2001, pp. 644–651.

[10] A. K. R. Chowdhury and R. Chellappa, "Face reconstruction from monocular video using uncertainty analysis and a generic model," *Comput. Vis. Img. Unders.*, vol. 91, no. 12, pp. 188–213, 2003.

[11] D. Fidaleo and G. Medioni, "Model-assisted 3D face reconstruction from video," in *Analysis and modeling of faces and gestures*. Springer, 2007, pp. 124–138.

[12] R. A. Newcombe, A. J. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, D. Molyneaux, S. Hodges, D. Kim, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *IEEE Intern. Symp. on Mixed and Aug. Reality*, Basel, Switerland, Oct. 2011, pp. 127–136.

[13] T. Whelan, M. Kaess, M. Fallon, H. Johannsson, J. Leonard, and J. McDonald, "Kintinuous : Spatially extended KinectFusion," in *RSS Workshop on RGB-D: Advanced Reasoning with Depth Cameras*, Sydney, Australia, Jul. 2012.

[14] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 8, pp. 1362–1376, 2010.

[15] A. Geva, G. Briskin, E. Rivlin, and H. Rotstein, "Estimating camera pose using bundle adjustment and digital terrain model constraints," in *Proc. IEEE Int. Conf. Robot. and Automation*, Seattle, WA, USA, May 2015, pp. 4000–4005.

[16] T. G. Farr, P. A. Rosen, E. Caro, R. Crippen, R. Duren, S. Hensley, M. Kobrick, M. Paller, E. Rodriguez, L. Roth *et al.*, "The shuttle radar topography mission," *Reviews of geophysics*, vol. 45, no. 2, 2007.

[17] P. Agarwal, G. D. Tipaldi, L. Spinello, C. Stachniss, and W. Burgard, "Robust map optimization using dynamic covariance scaling," in *Proc. IEEE Int. Conf. Robot. and Automation*, Karlsruhe, Germany, May 2013, pp. 62–69.

[18] N. Sunderhauf and P. Protzel, "Switchable constraints for robust pose graph SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, Algarve, Portugal, 2012, pp. 1879–1884.

[19] R. Morton and E. Olson, "Robust sensor characterization via max-mixture models: GPS sensors," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, Tokyo, Japan, Nov. 2013, pp. 528–533.

[20] F. Dellaert and M. Kaess, "Square root SAM: Simultaneous localization and mapping via square root information smoothing," *Int. J. Robot. Res.*, vol. 25, no. 12, pp. 1181–1203, 2006.

[21] S. Agarwal, K. Mierle, and Others, "Ceres solver," http://ceres-solver.org.

[22] A. Kim and R. M. Eustice, "Real-time visual SLAM for autonomous underwater hull inspection using visual saliency," *IEEE Trans. Robot.*, vol. 29, no. 3, pp. 719–733, Jun. 2013.

[23] S. M. Chaves, A. Kim, and R. M. Eustice, "Opportunistic sampling-based planning for active visual SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, Chicago, IL, USA, Sep. 2014, pp. 3073–3080.

[24] A. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP," in *Proc. Robot.: Sci. & Syst. Conf.*, Seattle, WA, USA, Jun. 2009.

[25] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[26] A. Kim and R. M. Eustice, "Pose-graph visual SLAM with geometric model selection for autonomous underwater ship hull inspection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, St. Louis, MO, USA, Oct. 2009, pp. 1559–1565.