SONIC(SONar Image Correspondence): Pose Supervised Learning for Forward Looking Sonar Image Matching

Samiran Gode*[1], Akshay Hinduja*[1] and Michael Kaess[2]

***Abstract*— Underwater localization and mapping remains an unsurmounted challenge with multiple attempts, but always hindered through assumptions or lack of generalizations for every situation. What makes the underwater environment so challenging is that fact that cameras are often faced with limited visibility, which limits the field of view to a few meters of often featureless areas. Imaging sonars are hence often preferred as the perception sensor of choice. However, they too have several drawbacks to them. While imaging sonars are able to see further than their optical counterparts, the measurements they make of a scene usually do not stay consistent with change in viewpoints. This makes the data association aspect for feature based methods very difficult. In this paper we introduce a pose-supervised network which provides us with feature descriptor which are robust to changes in view points which allow us to obtain far more reliable feature matches when compared to traditional descriptors like AKAZE and camera image trained feature descriptor networks. Furthermore, these enhanced descriptors exhibit superior accuracy in matching sonar images even with substantial viewpoint variance, paving the way for efficient loop closures and bolstering sonar-based place recognition capabilities.**

I. Introduction

Feature-based methods are a popular technique to perform localization and mapping, where uniquely identified features are tracked and matched across frames. These methods work particularly well for frameworks using cameras as their choice of perception, as we can leverage the photometric consistency of camera images to generate accurate feature correspondences. Camera-based frameworks often falter in underwater environments for several reasons. One of which includes the loss of color information that occurs at increasing depths and limited visibility due to turbidity - thereby severely limiting the amount of useful data the robot can use. To combat these problems, imaging sonars are very useful to use in underwater situations. Therefore, imaging sonars are the best available sensor for underwater scenarios, but lack compatible, robust feature descriptors. Traditional feature descriptors tend to suffer due to the speckle noise in sonar images and any large changes in the viewing angle. Recent research on learned feature descriptors for cameras have shown encouraging results when used for correspondence estimation. Most supervised methods require ground truth correspondence between feature points for training, however recently a novel weakly-supervised framework has been able to train robust feature descriptors by using relative camera poses without relying on ground truth correspondences. In this work, we apply a weakly-supervised framework for sonar images by leveraging an analogue to the epipolar line for sonar to be used as a loss function. This gives a data and time efficient way of obtaining trained feature descriptors for sonar images which can outperform the traditional methods used.

The authors are affiliated with the Department of Mechanical Engineering[1], the Robotics Institute[2], Carnegie Mellon University, Pittsburgh, PA 15213, USA. *These two authors contribute equally to this work. {ahinduja, kaess}@andrew.cmu.edu, sgode@alumni.cmu.edu

II. Related Work

Feature descriptors and feature detection are a well studied area of research in the optical image space. A few popular examples of feature descriptors are SIFT [1], ORB [2] and AKAZE [3]. These feature descriptors work very well for optical images, and a large contribution to their success is the fact that images of a scene taken from moderate changes in viewing angle do not alter the pixel values of the objects drastically. This concept is commonly referred to as photometric consistency. This consistency is not applicable to images generated from imaging sonars. The same object viewed from different angles would give different intensity returns as well as different observable shapes, something which is dictated by the material and geometry of the object under observation. Feature correspondence using these descriptors requires that they remain consistent and reliable for a particular scene. The aforementioned descriptors have been used for sonar images in tasks regarding acoustic structure from motion and feature-based SLAM [4, 5]. These descriptors worked as long as the change in viewing angle was minimal. Tueller et al. [6] give a summary of different keypoint detectors on sonar images but do not discuss how well the descriptors perform for feature matching. As these feature descriptors were developed keeping RGB images in mind, which are represented in euclidean space and do not experience the large speckle noise as seen in acoustic images, they can fail under common conditions. Examples of this in simulation and real-world scenarios are given in Figure 1. The downstream effects of this failure can result in poor, or error-prone loop closure detection as well as state estimation. Thus, for reliable feature-based SLAM, a robust feature descriptor and correspondence method designed for sonar images is required.

Developing a new feature descriptor for a specific type of sensor is a possibility [8], and there has been recent development on a SIFT like descriptor made for multi-beam sonar [9]. The drawback to this process is that the descriptor would need tuned for different imaging sonar models, as each sonar make has a unique elevation, bearing

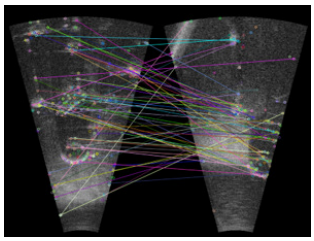
and range specification. Recent research on visual descriptors has focused on deep learning to solve this task. There have been many supervised methods such as Superpoint [10] and LOFTR [11]. In Superpoint, the feature point and feature descriptor generation is kept disjoint. Superpoint uses magicpoint to detect keypoints, whereas the superpoint network works on producing the learned feature descriptors. Follow up work in Superglue [12] utlizes the superpoint feature descriptors to form a robust feature matching framework. As one would expect, approaches similar to superpoint rely heavily on data.

These methods are hard to replicate for imaging sonar due to limited open access sonar data, as well as ground truth feature point information. As such, there is research which is focused on semi-supervised methods which leverage sensor pose information to negate the need for ground truth correspondences between feature points. They have been successful in achieving better accuracy than their fully supervised counterparts, while requiring less training data. An example of a pose supervised method is CAPSNet by Wang et al. [13]. Given the pose information between two images of the same scene, CAPSNet uses a set of two loss functions, epipolar loss and cyclic loss. The premise of their system is based on the principle that a point of interest in the first image, should lie on the epipolar line of the corresponding point on the second image.

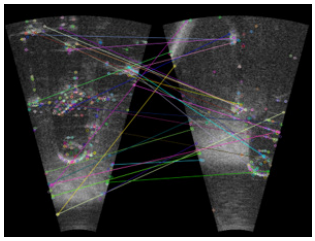
III. Preliminaries

A. Imaging Sonar Sensor Model

Imaging sonars are active acoustic sensors which work on the principle of measuring the intensity of the reflections of sound waves emitted by it. They are analogous to optical cameras in the sense that they interpret a 3D scene as a 2D image. However, the images provided by these sensors are very different. The imaging sonar sensor model uses a slightly different coordinate frame convention compared to the standard pinhole camera model for optical images. The x axis points forward from the acoustic center of the sensor, with the y axis pointed to the right and z axis pointed downward, as shown in Figure 2. This coordinate frame is preferred due to the way the image formation occurs for imaging sonars. Unlike in the pinhole camera model, where the scene from the viewable frustum is projected onto a forward-facing image plane, for imaging sonars the scene is projected into the zero elevation plane, which is the xy plane in the sonar frame.



(a) AKAZE and Brute Force Matching (BF).



(b) AKAZE with Joint Compatibility Branch and Bound (JCBB).

Fig. 1: AKAZE provides a reasonably good number of keypoints. However, when pose differences may be large, the descriptors will fail to provide matchable pairs. Even robust matchers like JCBB are unable to provide meaninful correspondences. Images are from the underwater classification dataset by Singh et al. [7].

Following the notation in [14], consider a point $\mathbf{P}$ with spherical coordinates (r, θ, ϕ) - range, azimuth, and elevation, with respect to the sonar sensor. The corresponding Cartesian coordinates are then

$$\mathbf{P} = \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} = r \begin{bmatrix} \cos\theta\cos\phi \\ \sin\theta\cos\phi \\ \sin\theta \end{bmatrix} \tag{1}$$

When considering an image formed from the pinhole camera model, an arbitrary pixel location can provide us information like the azimuth and elevation angle, but the range information is ambiguous. Every point lying along the ray in 3D would project onto the same pixel location in the image. In the imaging sonar model, a 3D point is projected onto the image plane, or zero elevation plane as

$$\mathbf{p} = \begin{bmatrix} x_s \\ y_s \end{bmatrix} = r \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} = \frac{1}{\cos\phi} \begin{bmatrix} X_s \\ Y_s \end{bmatrix}. \tag{2}$$

In the projective camera model, each pixel has a corresponding ray that passes through the sensor origin. In contrast, the sonar sensor model has a finite elevation arc in 3D space, as seen in Figure 2 as a red dotted line. A pixel in a sonar image can provide us the azimuth and range information of the arc, but loses the elevation of the arc entirely, just like how range information is lost for the projective camera model. This nonlinear projection adds siginificant complexity, coupled with the very narrow field of view most imaging sonar sensors have makes seemingly straightforward tasks for optical cameras appear rather difficult for imaging sonars. Another complication arises from the information these pixels hold. For optical images, pixels measure the intensity of the light reflected towards the sensor from a particular surface patch along the corresponding ray. Thus, each pixel would likely correspond to a single unique surface patch. This allows for different viewpoints to have similar pixel values when accounting for small motions of the camera origin. On the other hand, when it comes to acoustic images, each pixel need not correspond to a single surface patch. All surfaces along an elevation arc may reflect sound emitted by the sensor, and as such a single pixel may contain the compounded intensity of multiple surfaces, which may not be consistent even with slight changes to the sensor origin.

IV. Method

A. Sonar Epipolar Geometry

Negahdaripour introduced the concept of the epipolar contour [16]. In cameras, the *epipolar line* is defined as the intersection of the epipolar plane of a point in space with the image plane. As a result, it can also be thought of as the projection of the line joining the camera center and the

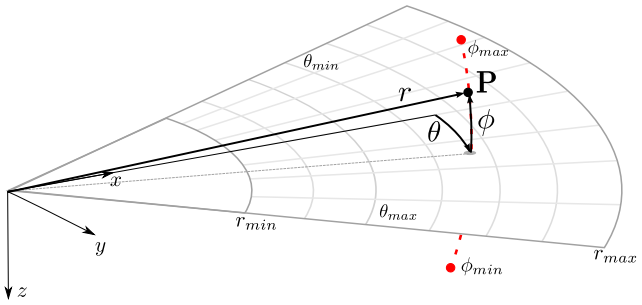


Fig. 2: The basic imaging sonar sensor model of a point feature. Each pixel provides direct measurements of the bearing / azimuth (θ) and range (r), but the elevation angle (ϕ) is lost in the projection onto the image plane - analogous to the loss of the range in the perspective projection of an optical camera. The imaged volume, called the frustum, is defined by the sensors limits in azimuth $[\theta_{min}, \theta_{max}]$, range $[r_{min}, r_{max}]$, and elevation $[\phi_{min}, \phi_{max}]$. Reprinted from [15].

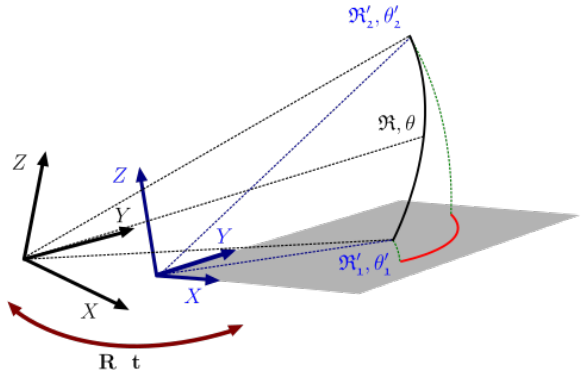


Fig. 3: Sonar Epipolar Geometry: The elevation arc of a point in the first image is transformed into the frame of the second image and then projected, which creates an epipolar contour.

point in 3D on to the image plane of the other camera . This line is a function of the line going into the plane from the first view, in other words it can be considered as the depth. When it comes to sonar stereo, epipolar geometry involves the use of the elevation arcs as the analogue to epipolar lines, and using their projection as an *epipolar contour*. We will describe the epipolar geometry in brief here, but refer the reader to [16] for a detailed explanation.

We refer to Section III-Awhere we again use Equation 1 for notation relating to the conversion of a point in polar space to the Cartesian space. However, since we are focused on training sonar images, we prefer to use pixels in the range bearing space instead. Given a point in one sonar image at some range, R and bearing θ. We know that the point in 3D will lie along the elevation arc at the same range and bearing as defined and at some arbitrary elevation angle, φ. Since we do not know the elevation angle, the ambiguity is along the elevation arc for $\varphi <= \varphi_{max}$ and $\varphi >= \varphi_{min}$. Now, the conversion of points in 3D Cartesian space to the range bearing space is as seen in Equation 3.

$$\boldsymbol{P} = \begin{bmatrix} R \\ \theta \\ \phi \end{bmatrix} = r \begin{bmatrix} \sqrt{x^2+y^2+z^2} \\ arctan2(y,x) \\ arctan2(x, \sqrt{x^2+y^2}) \end{bmatrix} \tag{3}$$

Assuming we have a feature point $\boldsymbol{p}$ in the first image, and $\boldsymbol{R_{1,2}}$and $\boldsymbol{t_{1,2}}$are the known rotation and translation between the coordinate frame of image 1 and image 2. The locus of the same feature point in image 2, $\boldsymbol{p}'$ is calculated as seen in Equation 5. This 3D feature locus can now be projected into the polar sonar image plane as in Equation 6. Figure 3 visually describes the process of projecting the elevation arc from the first frame onto the second image plane. The red line is the *epipolar contour* we use for the loss function described in a later section.

$$\boldsymbol{R_{1,2}} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}, \boldsymbol{t_{1,2}} = \begin{bmatrix} t_x & t_y & t_z \end{bmatrix}^T \tag{4}$$

$$\boldsymbol{p}'(\phi) = \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} r_1 \cdot P + t_x \\ r_2 \cdot P + t_y \\ r_3 \cdot P + t_z \end{bmatrix} \tag{5}$$

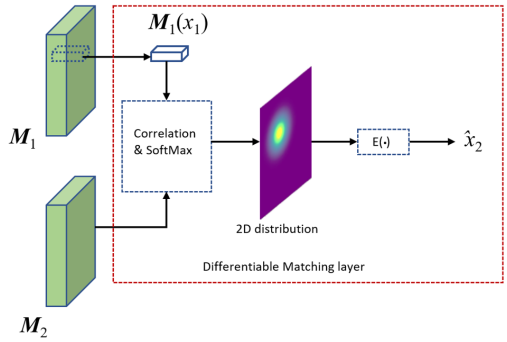
$$\boldsymbol{p}'_{\boldsymbol{proj}}(\phi) = \begin{bmatrix} r' \\ \theta' \end{bmatrix} = \begin{bmatrix} \left\| P'_p \right\|_2 \\ arctan2(y', x') \end{bmatrix} \tag{6}$$

B. Keypoint Detection

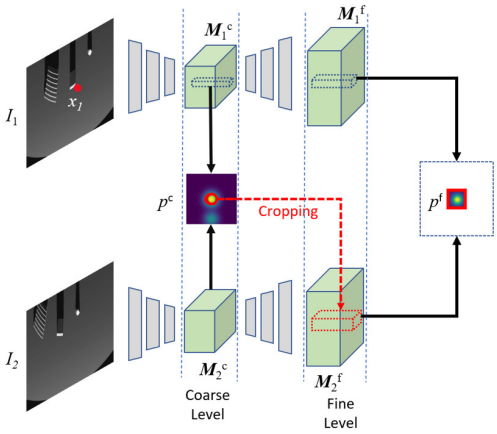
As CAPSNet is focused on training keypoint descriptors, rather than provide keypoints itself, we need to provide the initial keypoints for each frame. We propose two possible approaches for this – The first of which utilizes traditional keypoint detectors from AKAZE, ORB and SIFT. The secon approach would be using a pre-trained network for learned keypoint detectors like Magicpoint, the keypoint detector from Superpoint. As the number of feature points in sonar images vary a lot by scene and are generally limited, a combination of them to reach the minimum keypoints per frame might be necessary. It may also be necessary to train Magicpoint through a synthetic shapes dataset with range-bearing space sonar images.

C. Network highlights

We will use a CNN based encoder and decoder architecture with a differentiable matching layer similar to [13]. CAPSNet is based on the ResNet-50 architecture, truncated at `layer3`. From here, they introduce their coarse-to-fine architecture. Another convolution sets up the coarse representation which is used to calculate the loss in the in encoder representation and the decoder up-samples the coarser layer, which helps create a finer representation. The architecture is shown in Figure 4. Searching correspondences for all points over the image is very computationally costly, hence it makes most sense to sparsely sample the query points for supervision. The coarse to fine architecture improves on efficiency. At the coarse level, a correspondence distribution is computed over all the locations. However, at the finer level the distribution is only computed at the highest probability location observed from the coarse map. The loss functions



(a) Differentiable Matching Layer



(b) Coarse to Fine module

Fig. 4: Network architecture highlights: a) For each feature point, its correspondence location is represented as the expectation of a distribution computed from the correlation between the feature descriptors. The associated uncertainty also helps in reweighting training loss. (b) Searching correspondence across the entire image is costly. The location of highest probability at the coarse level is used to ascertain a local window at the fine level. This allows for greater computational efficiency.

are imposed on both levels and the trained descriptors at both levels are concatenated to give the final hierarchical descriptor.

The distribution generated at the coarse and fine level is achieved through the differential matching layer introduced by CAPSNet. Such layers are common in tasks for object tracking and image retrieval. The operating principle is as follows. We take input features from two different branches and produce a match score which shows the similarity in the inputs. Each feature descriptor x_1 in the first image's layer M_1 is given a 2D distribution indicating the probability of a location being the correspondence of $x_1$in the second image's layer M_2. They also use uncertainty obtained from this distribution to reweight points which may not be there in the second frame or be occluded, something which is highly likely in sonar images.

D. Loss Functions

Like CAPSNet, we propose two loss functions, with changes to make it more suitable for sonar images. The two loss terms are namely the sonar-epipolar and loss sonar-cyclic. While training we know the relative pose between two image frames $I_1$and I_2, and as such we can calculate the transformation between the two images. We can use Equations 5 and 6 to determine the epipolar contour. The predicted point, if predicted correctly, should lie on this epipolar contour. Thus, we use this as a distance metric that we can optimize over. The epipolar loss term, $L_{epipolar}$ in Equation 7 is defined as the shortest distance between the predicted correspondence of a feature point $x_1$in I_1 and the epipolar contour of x_1 in the second image I_2. In our implementation, we sample points along the elevation arc of the first point in the first image and then transform and project them on to the second image to create a discrete epipolar contour of the sampled points. In the polar frame, the loss is thus the minimum of the distance between the predicted point and each point on the arc as seen in Equation 11. The epipolar loss only checks for the predicted match to lie on the estimated contour, and does not necessarily look in the vicinity of the ground truth correspondence location. To further constrain the system, a cyclic consistency loss is utilized which aims to keep the forward-backward mapping of the point to be close to itself. The weighted sum of the losses $L_{epipolar}$ and L_{cyclic} is our final loss function as seen in Equation11. A point to note is that the distances found for both the losses are in the range-bearing space. Due to nature of sonar images, it is important for the network to learn this distinction, and the combined loss terms in the range bearing space help with this. A graphical representation of the losses is seen in Figure 5.

$$L_{epipolar}(x_1) = dist(h_{1\to2}(x_1), ep\_contour) \tag{7}$$

$$L_{cyclic}(x_1) = \|h_{2\to1}(h_{1\to2}(x_1) - x_1\|_2 \tag{8}$$

$$L_{(I_1,I_2)} = \sum_{i=1}^{n}[L_{epipolar}(x_1^i) + \lambda L_{cyclic}(x_1^i)] \tag{9}$$

$$Loss_{cyclic}((r_1,\theta_1),(r_2,\theta_2)) = r_1^2 + r_2^2 - 2r_1r_2(cos(\theta_1 - \theta_2)) \tag{10}$$

$$Loss_{epipolar}(p\prime_{proj}(\varphi), r_2, \theta_2) = \min_{\phi}(r(\varphi)^2 + r_2^2 - 2r(\varphi)r_2(cos(\theta(\varphi) - \theta_2))) \tag{11}$$

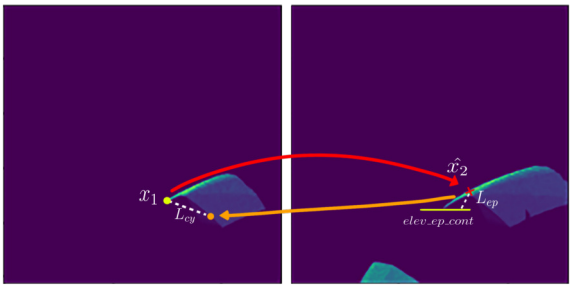
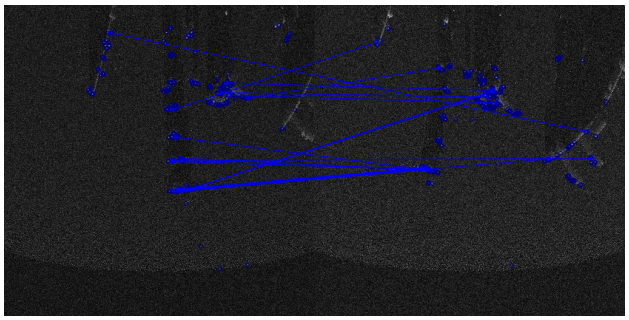
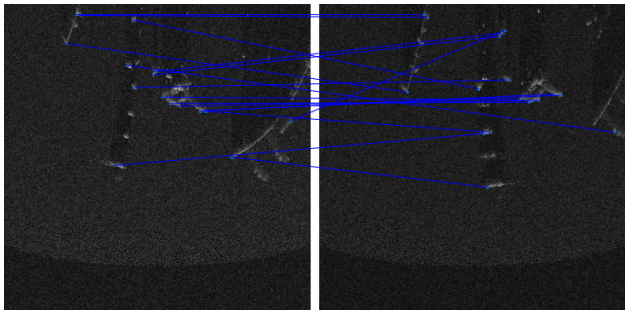


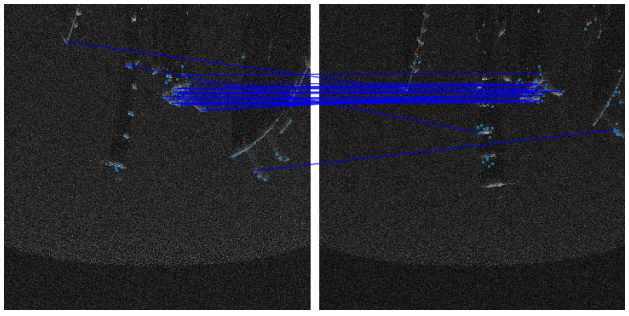
Fig. 5: Loss functions: The yellow point x_1 represents a feature point in the first image. The red cross $\hat{x_2}$is the predicted point. L_{ep} is the shortest distance to the epipolar contour, or simply the epipolar loss. L_{cy} is the cyclic loss to assert that the mapping of the feature point is close to its original position.



(a) AKAZE Descriptor Matches

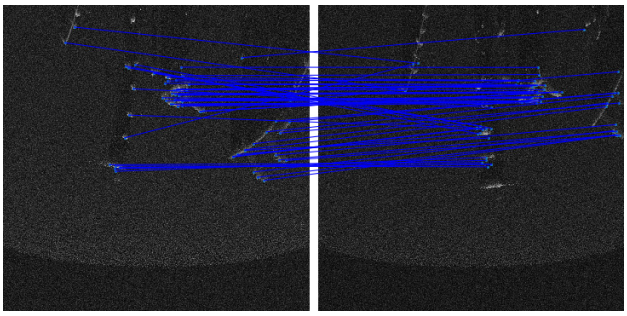


(b) Superpoint Descriptor Matches

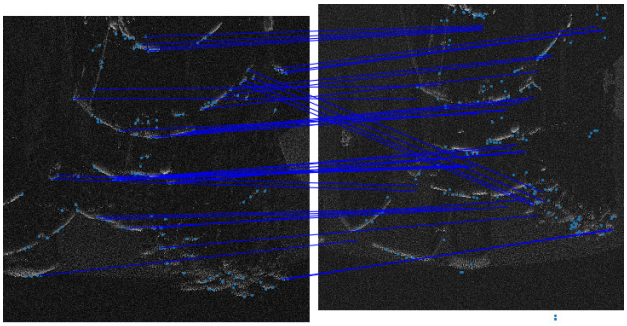


(c) SONIC Descriptor Matches

Fig. 6: Qualitative Evaluation: (a) AKAZE descriptors with a symmetrical match threshold of 0.8 is unable to provide reasonable matches. (b) Superpoint descriptors with a symmetrical match threshold 0.8 give better matches, but still unreliable. (c) SONIC descriptors with the same threshold are able to give a large percentage of accurate matches.



(a) Expectation matching



(b) Expectation matching with a large variation in viewpoint

Fig. 7: Qualitative Evaluation: (a) Using expectation matching we see all the matches made and see a very high number of them are accurate. (b) In a much tougher scenario with a large variation in sensor poses, we see our descriptors still providing good matches.

c

V. Evaluation and Future Work

A. Image Matching Results

To start our evaluation, we first see qualitative matching results for two sonar image pairs and compare it to AKAZE and Superpoint. These images are to simulate a loop closure where the robot returns to a position it has This can be seen in image In Figure 6 we see that with an aggressive threshold of 0.8 using the Lowe ratio test we see AKAZE descriptors are not able to get reasonable matches. A pre-trained network of Superpoint is able to do better but still not good enough. On the other hand, our proposed descriptors are able to perform much better.

A few more examples are also see in Figure 7. While our current results are with a training set of 60,000 pairs, we see markedly better performance. However, we aim to train on a much larger dataset of 300,000 - 400,000 pairs to improve the number of potential matches.

While qualitatively we show a marked improvment, quantitative proof is needed. In future work we will implement a two-view acoustic bundle adjustment framework to recover sensor pose information as presented by Westman et al. [17]. This will allow a more robust evaluation metric. As next steps, we aim to apply this network towards feature based SLAM. Future iterations will focus on integrating more imaging sonar makes, and also focus on cross-sonar descriptors to enable relocalization in a map made by a different imaging sonar.

REFERENCES

[1] P. C. Ng and S. Henikoff, "Sift: Predicting amino acid changes that affect protein function," *Nucleic acids research*, vol. 31, no. 13, pp. 3812–3814, 2003.

[2] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International conference on computer vision*, pp. 2564–2571, Ieee, 2011.

[3] P. F. Alcantarilla and T. Solutions, "Fast explicit diffusion for accelerated features in nonlinear scale spaces," *IEEE Trans. Patt. Anal. Mach. Intell*, vol. 34, no. 7, pp. 1281–1298, 2011.

[4] T. A. Huang and M. Kaess, "Towards acoustic structure from motion for imaging sonar," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pp. 758–765, Oct. 2015.

[5] E. Westman, A. Hinduja, and M. Kaess, "Feature-based SLAM for imaging sonar with under-constrained landmarks," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pp. 3629–3636, May 2018.

[6] P. Tueller, R. Kastner, and R. Diamant, "A comparison of feature detectors for underwater sonar imagery," in *OCEANS 2018 MTS/IEEE Charleston*, pp. 1–6, 2018.

[7] D. Singh and M. Valdenegro-Toro, "The marine debris dataset for forward-looking sonar semantic segmentation," in *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, (Los Alamitos, CA, USA), pp. 3734–3742, IEEE Computer Society, oct 2021.

[8] P. Hansen, P. Corke, W. Boles, and K. Daniilidis, "Scale invariant feature matching with wide angle images," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1689–1694, IEEE, 2007.

[9] W. Zhang, T. Zhou, C. Xu, and M. Liu, "A sift-like feature detector and descriptor for multibeam sonar imaging," Jul 2021.

[10] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 224–236, 2018.

[11] J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou, "Loftr: Detector-free local feature matching with transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8922–8931, 2021.

[12] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superglue: Learning feature matching with graph neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4938–4947, 2020.

[13] Q. Wang, X. Zhou, B. Hariharan, and N. Snavely, "Learning feature descriptors using camera pose supervision," in *European Conference on Computer Vision*, pp. 757–774, Springer, 2020.

[14] N. Hurtós, D. Ribas, X. Cufí, Y. Petillot, and J. Salvi, "Fourier-based registration for robust forward-looking sonar mosaicing in low-visibility underwater environments," *J. of Field Robotics*, vol. 32, no. 1, pp. 123–151, 2014.

[15] E. Westman, *Underwater Localization and Mapping with Imaging Sonar*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, October 2019.

[16] S. Negahdaripour, "Analyzing epipolar geometry of 2-D forward-scan sonar stereo for matching and 3-D reconstruction," in *Proc. of the IEEE/MTS OCEANS Conf. and Exhibition*, pp. 1–10, 2018.

[17] E. Westman and M. Kaess, "Degeneracy-aware imaging sonar simultaneous localization and mapping," *IEEE J. of Oceanic Engineering*, 2019.