Survey Registration For Long–Term Natural Environment Monitoring

Shane Griffith
GeorgiaTech Lorraine - UMI 2958 GT-CNRS
sgriffith7@gatech.edu

Cédric Pradalier
GeorgiaTech Lorraine - UMI 2958 GT-CNRS
cedric.pradalier@gatech.edu

Abstract

This paper presents a survey registration framework to assist in the recurrent inspection of a natural environment. Our framework coarsely aligns surveys at the image–level using visual SLAM, and it registers images at the pixel–level using SIFT Flow, which enables rapid manual inspection. The variation in appearance of natural environments make data association a primary challenge of this work. We discuss this and other challenges, including: 1) alternative approaches for coarsely aligning surveys of a natural environment; 2) how to select which images to compare between two surveys; and 3) strategies to boost image registration accuracy.

We evaluate each stage of our approach, emphasizing alignment accuracy and stability with respect to large seasonal variations. Our domain is lakeshore monitoring, in which an autonomous surface vessel surveyed a 1 km lakeshore 33 times in 14 months. Our results show that our framework precisely aligns a significant number of images between surveys captured up to roughly three months apart, often across marked variation in appearance. Using these results, a human was able to spot several changes between surveys that would have otherwise gone unnoticed.

1 Introduction

This paper presents a survey registration framework to assist in the recurrent inspection of a natural environment[1]. Robots may soon automate many tasks in natural environments that require repeated observation and management. Strides have been made towards precision agriculture [Carlone et al., 2015; Bargoti et al., 2015], search and rescue along forest trails [Giusti et al., 2015], and a number of different types of environment monitoring [Kularatne and Hsieh, 2015; Hitz et al., 2014a; Cabrol et al., 2012]. In these scenarios, a robot that gathers and analyzes data may be tasked with spraying certain crops, notifying the authorities, or compiling a record of changes. Before a robot can make these decisions, it may first associate data from different surveys in order to compare it. Natural environments pose, however, a significant challenge to visual data association, due to the variation in appearance that occurs there.

An assortment of techniques have been proposed to overcome different types of variation in appearance in outdoor and natural environments. The choice of point–based feature descriptors can be important, as some are better suited to natural environments than others [Krajnik et al., 2015], while patches and whole images gain robustness because they capture scene structures [Liu et al., 2011; McManus et al., 2014]. Different strategies have increased robustness to illumination/shadows [Corke et al., 2013], night [Nelson et al., 2015], and noise [Gu et al., 2009]. As these and other types of noise create ambiguous cases for data association, sequences of images have been shown to improve coherence [Churchill and Newman, 2013].

[1]This paper is an extension of a paper first presented at the 2015 Conference on Field and Service Robotics [Griffith and Pradalier, 2015]. It also builds on two previous papers [Griffith et al., 2014, 2015].

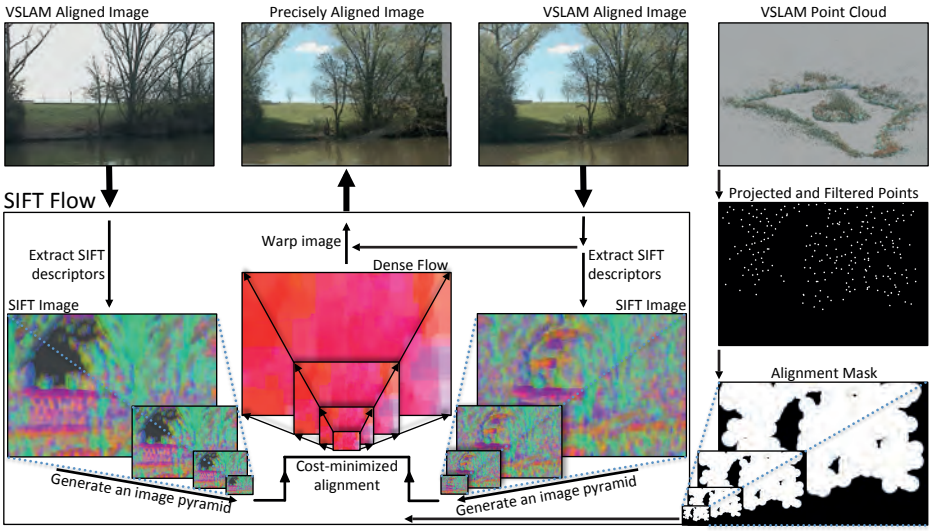


Figure 1: The registration of two images, which visual SLAM found to capture the same scene. For each image, SIFT descriptors are computed at each pixel to form a SIFT image, which is down–sampled into an image pyramid. An image mask representing the lakeshore (derived from the 3D information in the feature tracks of visual SLAM) is used to bias where the SIFT images are aligned, which helps avoid aligning noise due to the sky or the water. The output flow aligns one of the input images to the other, which enables quick change detection for manual inspection tasks.

Rather than use a different technique to address each type of variation, recent work has shown that it may be possible to overcome them all using a single approach. One deep learning method for matching image patches is robust to variation in appearance and environment condition [Sunderhauf et al., 2015]. Yet, only portions of images are matched, the matches can be imprecise, and the technique does not extend well to natural environments (see Section 5.3). Image registration can also overcome different types of variation in appearance, but it sacrifices robustness to large changes in viewpoint [Liu et al., 2011]. Thus, for applications in natural environment monitoring in which a robot is likely to repeatedly take a similar path, image registration techniques may be most suitable. Yet, few papers have addressed survey registration in the context of long–term natural environment monitoring.

This paper presents a framework for achieving pixel–level image alignment between fortnightly surveys of a lakeshore. Our framework divides survey inspection into three steps: 1) coarse alignment, in which images are paired; 2) precise alignment, in which paired images are registered; and 3) change detection, in which registered image pairs are flickered back-and-forth to enable rapid human inspection. We compare state-of-the-art techniques for both coarse and precise alignment and find that a framework based on Visual SLAM and SIFT Flow performs best, as shown in Fig. 1. We also discuss how to select which images are compared between two surveys and strategies to improve image registration.

This framework was applied to the alignment of 33 surveys of a 1 km lakeshore captured over 14 months, which represents a spatially large and a temporally long scale study using ASVs. In this application, most image pairs between surveys up to three months apart registered well. The rest mostly paired well. Upon inspecting the environment using the registered surveys, a human detected several changes, which would have otherwise gone unnoticed. This was possible because many registered image pairs achieved robustness to variation in appearance of the sky, the water, changes in objects on a lakeshore, and the seasonal changes of plants. The registration of several other image pairs failed in particular ways, however, which indicate directions for future work.

This paper is structured as follows. In Section 2, we demonstrate the difficulty of data association in a natural environment by asking humans to perform a simple labeling task. Related work is described in Section 3. Section 4 frames this problem in terms of our experimental platform and application domain. Our framework is introduced in three sections: coarse survey alignment in Section 5, precise survey alignment in Section 6, and manual change detection in Section 7. The conclusion follows.

Figure 2: Variation in appearance of a section of the lakeshore in the span of nearly a year, captured in six images. There is significant variation in the vegetation, the lighting, the sun glare, and the water level. This makes survey registration difficult.

2 The Difficulty of Data Association in a Natural Environment

A high degree of variation in appearance must be overcome in the data association of images of a natural environment (see Fig. 2). To characterize the difficulty of this task, we asked humans to perform manual feature recognition. Humans searched for matching features between pre-aligned image pairs while we measured the time it took them. All our surveys were first coarsely aligned using our framework. Given a random pair from this set, subjects either found and clicked a matching feature in both images or marked the alignment a failure. The latter label was also used if the subject could not recognize a co-occurring feature in both images. For this analysis, we collected a set of 937 data points, which was truncated at three minutes to remove cases of experiment interruptions (13 of these).

Figure 3 shows the results. Subjects marked 902 (98%) of the image pairs as successfully aligned. Most image pairs provided enough context for the comparison—99% were shifted by less than half the image width and 90% by less than one fourth. Yet, 111 of them (12%) required more than 30 seconds to recognize and then select a single matching feature in both frames. Label time had little correlation with the percent of image overlap, and surprisingly, little with the amount of elapsed time between surveys.

The difficulty of manual image alignment was primarily due to the lack of persistent structural features at particular locations along the shore. Lone trees and buildings were easiest to match; dense foliage was hardest. Although the foliage dropped away in some months, the uniformity of the remaining features also made those difficult to match. Examples of hard-to-match image pairs are shown in Fig. 4, which highlights the novelty of our dataset and the challenges of survey alignment for long–term environment monitoring.

Note that humans almost always found one matching feature point among hard-to-match cases. Some detail of the scene had to remain perceptible in both images in order to do this. However, specific details were not always matched. For example, in some cases matches were placed on the center of a bush, which was clearly the same, but whose foliage obscured a precise match. The uniformity of features and the fact that humans performed time–consuming spatial searches to find one matching point suggests that point–based feature matching is inapplicable to natural environments. Instead, correspondence of scene structure may be better suited to the recognition and alignment task.

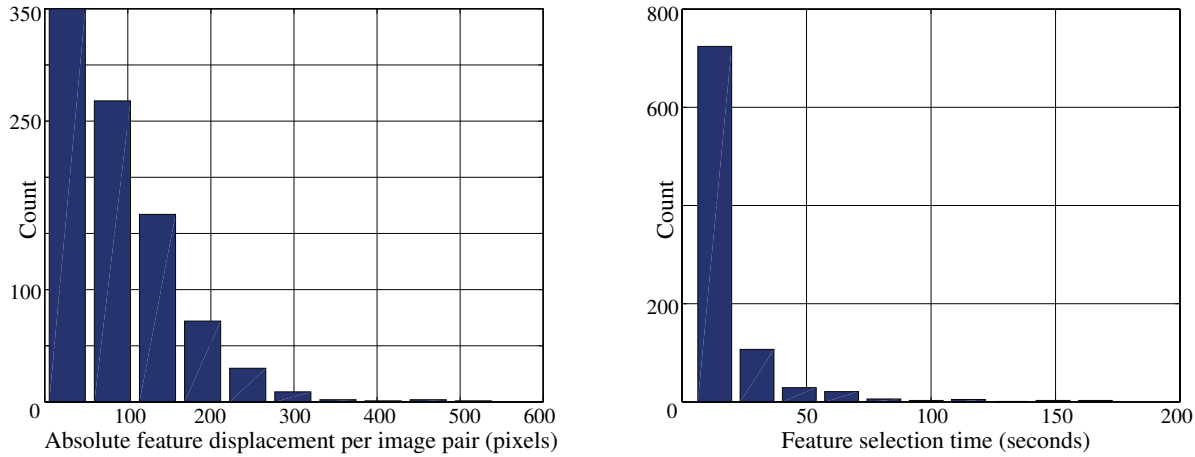


Figure 3: **left)** The distribution of feature displacements on user–marked image pairs, which indicates most image pairs had significant overlap. **right)** The distribution of time spent labeling the images. Humans took longer than 30 seconds to find a single matching feature in 12% of the image pairs.

3 Related Work

A map of the shore is one of the primary components of lakeshore monitoring. For this, the field of Simultaneous Localization and Mapping (SLAM) provides the foundation. However, a few different systems have been established specifically for lakeshores. Heidarsson and Sukhatme [2011] and Subramanian et al. [2006] are among the first to demonstrate SLAM on a lake. In case a robot is repeatedly deployed on the same lake, Hitz et al. [2014b] show that 3D laser scans of a shoreline can be used to identify some types of changes. Their system distinguished the dynamic leaves from the static trunk of a willow tree in two different surveys collected in the fall and the spring. Dense laser scans can also be used to ascertain crop growth [Carlone et al., 2015]. Rather than use an ASV, Jain et al. [2013] demonstrated the advantages of using a drone for mapping a shoreline, which flies above debris and below dense tree cover.

Natural environments still, however, pose several challenges to SLAM systems. A few examples include: 1) their large spatial scale; 2) their dynamic features (e.g., moving trees, changing water levels); and 3) their high level of visual similarity (e.g., branches and leaves of different trees may appear to be from the same one). The third, i.e., place recognition, is a central focus of this paper. Work on data association and sequence alignment investigate how this can be performed, albeit usually for more structured domains. We discuss recent work in these two areas below.

3.1 Visual Data Association in Outdoor Settings

In many applications point–based data association can find the most similar image in a database. A set of features are extracted from an image and used to search for a previous observation of the same scene. SIFT features are typically used, being the state-of-the-art method (e.g., Košecka [2013]; Beall and Dellaert [2014]; He et al. [2006]). Unfortunately, SIFT matching (using OpenCV [Bradski, 2000]) in natural environments often fails or returns too few features (less than 10) due to the intra-image similarity of the descriptors [Griffith et al., 2014]. This has led to studies in which other point–based features were tested for their suitability for data association in outdoor settings Krajnik et al. [2015]. In these environments, the BRIEF feature descriptor (and the variant GRIEF) has been shown to outperform SIFT [Krajnik et al., 2015]. In our own evaluation, BRIEF, ORB, SIFT, and SURF performed very similarly, with a slight qualitative advantage for ORB. The default implementation of ORB had the fastest execution time, which also favored it for the analysis.

A dictionary of representative features are used in some approaches (e.g., FabMAP [Cummins and Newman, 2008]). Places are identified using a histogram of occurring features. Such features could be image patches or standard feature

May 2 and May 28 | Feb. 5 and June 6 | June 6 and Apr. 9

Figure 4: Three image pairs (one per column) that are hard for a human to align. A human spent over 30 seconds on each image pair, both validating that the images capture the same place and then selecting the same physical feature in them.

descriptors. In testing the default implementation of OpenFabMAP, FabMAP had the same shortcoming as SIFT–based image retrieval since many of our scenes contain repetitive features. Although the method can retrieve similar looking images, it does not achieve a globally consistent match set.

Due to the shortcomings of keypoint descriptors, some work has focused on directly using whole or parts of images. Low–resolution image thumbnails are used (or sequence of image thumbnails) as input to the matching system of SeqSLAM [Milford et al., 2004, 2014]. Milford et al. [2014] showed that SeqSLAM could be augmented with point–based features and homography estimation to closely align images from different surveys of a natural environment. Although their goals are similar to the work presented here, the images were captured by a human, the image alignment approach relies on the accuracy of point–based feature matching, and a homography provides the warping function, which only precisely aligns images of planar scenes.

SeqSLAM is, additionally, highly sensitive to image viewpoint [Sünderhauf et al., 2013]. An image in the training set needs to have the same viewpoint as the sought–after image. It is not always feasible for an autonomous system in the field to achieve this level of repetition. On a lakeshore, for example, varying water levels, debris, and other factors yield a unique trajectory every survey. These factors contribute to its poor performance on our dataset (using the OpenSeqSlam implementation [Milford and Wyeth, 2012]).

Neubert et al. [2014] deals with seasonal changes by introducing a prediction step in which whole images are modified to look more like the current season. McManus et al. [2014] utilize patches of images, called ‘scene signatures’, which are matched using classifiers and capture information about the structure of each scene. In case a particular location is stubborn to data association, ‘multiple experiences’ of the location can be accumulated until new observations are associated well [Churchill and Newman, 2013].

A desirable aspect of long–term natural environment monitoring is the identification of groups of images of the same scene, which are captured at different times of the year (see Fig. 2). So far, this is perhaps best achieved in Martin-Brualla et al. [2015], in which time–lapse videos are created after mining the internet for thousands of images of the same place. Unfortunately, their approach discards many images whose SIFT features do not match. In monitoring tasks, images from every survey have to be part of the comprehensive result.

This paper promotes image registration as the mechanism for both viewpoint selection and precise alignment of images

of a natural environment. Rather than use an assortment of techniques to overcome different types of variation in appearance, the data association of whole images is a single technique that provides robustness to most of them (at the expense of robustness to variation in viewpoint). Note that image registration is popular in SLAM techniques for aligning consecutive images because a more dense set of depth estimates can be obtained for a scene compared to, e.g., sparse features tracking using Kanade-Lucas-Tomasi Feature Tracker (KLT) [Lucas and Kanade, 1981; Shi and Tomasi, 1994]. In LSD-SLAM, for example, intensity images are aligned, and the depth estimates of the scene are used to boost alignment performance [Engel et al., 2014]. In contrast, in this work, image registration is investigated for aligning images from different surveys, which undergo large variations in appearance.

Several image registration techniques in computer vision have been made to overcome significant variation in appearance [Liu et al., 2011; Kim et al., 2013; Yang et al., 2014]. This paper builds on SIFT Flow [Liu et al., 2011], which is designed to find dense correspondences among whole images worth of point–based features. It combines the precision of point–based feature matching with the robustness of whole–image matching. Yet, it aligns whole images indiscriminately of their contents. Its run time also constrains its use as a general data association strategy. In tailoring it to monitoring tasks, however, we have found ways to overcome these limitations.

3.2 Video Sequence Alignment

Video sequence alignment is the problem of time-synchronizing multiple videos of the same scene, typically with different viewpoints. In contrast to the approaches above, sequence alignment establishes coherence at the level of the complete video sequence instead of image-to-image. Thus, the matching solution is built consistently over the full sequence, which takes advantage of the frames with salient features to constrain the matches on less discriminative frames. As shown in Section 2, these types of approaches may be especially relevant for time–aligning surveys of natural environments, which have many uniform features.

Starting from single image retrieval, the simplest solution to the video sequence alignment problem is to find one point of alignment between the two sequences and then assume that they are recorded synchronously, i.e., every frame $(i+t)$ of the first sequence is matched to frame $(j+t)$ in the second sequence, where frame i from the first sequence has been identified as a match for frame j in the second sequence. A more complex approach would perform linear scaling between the two sequences, i.e., frame $(i+t)$ corresponds to $(j+a\cdot t)$. The scaling parameter can be computed given two matching pairs.

Wang et al. [2014] proposed a method based on time warping to find an alignment between two video sequences. The approach computes a matching cost for all potential pairwise matches using feature descriptors. The cost of matching frame i from the first sequence with frame j from the second sequence is inversely proportional to the number of matched feature descriptors between the two frames. False positive feature matches and outliers are not filtered out because they have a relatively weak effect on the cost matrix. The sequence alignment algorithm uses Dijkstra's shortest path algorithm to extract the ridge line of best matching frames.

Naseer et al. [2015] have extended the work of Wang et al. [2014] to address video sequence alignment in the midst of seasonal variation. They explored several approaches for building the cost matrix including features descriptors such as Histogram of Gaussians (HoG) and the node activations from a layer of a Convolutional Neural Network.

We tailor this approach to large–scale video sequences and then evaluate it on our dataset. Using manually labeled alignments as the ground truth, we compared pose–based and appearance–based approaches, and also included SIFT Flow as one type of descriptor of appearance. As will be demonstrated, the average performance of all the appearance–based approaches over large seasonal variations are always worse than a geometric, pose–based alignment of the sequences. Although it works well in structured environments (e.g., urban environments) in the presence of seasonal changes, we show that the cost matrix loses discriminative power in a natural environment.

Figure 5: The Kingfisher as it traversed the perimeter of Lake Symphony.

4 Experimental Setup

We used Clearpath's Kingfisher ASV for our experiments (see Fig. 5). It is about 1 meter long and 0.7 meters wide, with two pontoons, a water–tight compartment to house electronics, and an area on top for sensors and the battery. It is propelled by a water jet in each of its pontoons and turns during power differentials. It can reach a top speed of about 2 m/s, but we mostly operated it at lower speeds to maximize battery life, which is about an hour with our current payload.

Our Kingfisher is equipped with a suite of exteroceptive and interoceptive sensors. A 704x480 color pan-tilt camera (Axis P5512e) captures images at 10 frames per second. Beneath it sits a single scan line laser–range finder with a field of view of about 270 degrees. It is pointed just above the surface of the water and provides a distance estimate for everything less than 20 m away. The watertight compartment houses a GPS, a compass, and an IMU.

The ASV was deployed on Lake Symphony in Metz, France, which is about 400 meters long and 200 meters wide with an 80–meter–wide island in the middle. The nature of the lakeshore varied, with shrubs, trees, boulders, grass, sand, buildings, birds, and people in the immediate surroundings. People mostly kept to the walking trail and a bike path a few meters from the shore, and fishermen occasionally sat along the shore.

We used a simple planner to autonomously steer the robot around the perimeter of the lake and the island. As the boat moves at a constant velocity of about 0.4 m/s, a local planner chooses among a set of state lattice motion primitives to keep the boat 10 m away from the lakeshore on its starboard side (or port side for the island). Because our boat has a strong inertia and low rotational control authority a simple reactive behavior is insufficient. It needs to plan its maneuvers around tight turns and during trajectory following. Our planner is implemented as a model–predictive controller with a roughly identified model. With this configuration, the robot is capable of performing an entire survey autonomously; however, we occasionally took control using a remote control in order to avoid fishing lines, debris, to swap batteries, etc.

We have regularly deployed the robot up to once per week since August 18, 2013. At the time of this writing we are approaching 100 km of autonomous navigation and shore monitoring. This paper analyzes data from 33 different surveys (of roughly 80 total), which span fourteen months. A survey was part of this set if it was captured between January 2014 and February 2015 and consisted of a mostly complete run around the lakeshore and the island. Each survey was captured in the daytime on a weekday in sunny or cloudy weather, at various times of the day. Rainy weather was avoided because water droplets on the camera dome diffract the images of the shore. Note that the surveys were not always started and stopped in the same locations.

One survey represents a collection of image sequences, measurements of the camera pose, and other information

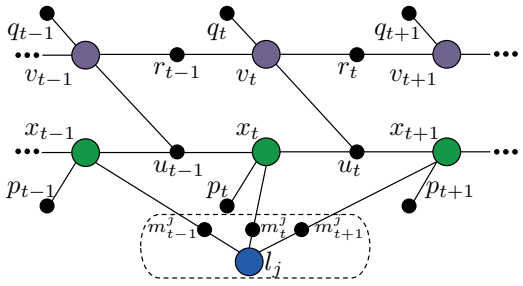


Figure 6: Factor graph of our visual SLAM approach. The colored nodes are the variables to be estimated. The black nodes depict factors, which constrain the variables with which they share an edge. The dotted line represents a smart factor [Carlone et al., 2014], which eliminates a landmark and the measurements of it from the factor graph. Each variable is described in Section 5.1.

about the robot's movement. During a survey, k, the robot acquires the tuple $\mathcal{A}^k$=$\{\mathcal{T}_i^k, \mathcal{I}_i^k, \hat{C}_i^k, \hat{\omega}_i^k\}_{i=1}^{|\mathcal{A}^k|}$ every 10^{th} of a second, where $\mathcal{T}$ is the current time, $\mathcal{I}$ is the image from the pan-tilt camera, $\hat{C} \in$ SE(3) is the estimated camera pose, and $\hat{\omega}$ is the estimated angular velocity of the boat. The estimated camera pose is derived from the boat's GPS position, the boat's compass heading, and the pan and tilt positions of the camera. The IMU provides $\hat{\omega}$. Each survey is down–sampled by a factor of five to ten to reduce data redundancy and speed up computation time.

5 Coarse Survey Alignment

The objective of coarse survey alignment is to find a sequence of pairs, (i, j), between two surveys, $\mathcal{A}^1$ and $\mathcal{A}^2$, such that for every pair of images $(\mathcal{I}_i^1, \mathcal{I}_j^2)$, $\mathcal{I}_i^1$ and $\mathcal{I}_j^2$ correspond to the same place. More formally, we define an alignment $\mathcal{H}_m^n$ between survey $\mathcal{A}^m$ and $\mathcal{A}^n$ as the sequence of pairs

$$\mathcal{H}_m^n = \left\{ (i,j) \text{ such that } j = \arg\min_j d(C_i^m, C_j^n) \right\}, \tag{1}$$

where d is a distance metric that indicates how well the two images correspond to the same scene. Applied to surveys, the notion of "correspondence" is somewhat ambiguous. Given that all our images have the same reference frame (due to GPS and the compass), the corresponding image in another survey would be the one from the closest viewpoint because it is likely to have the most overlap. However, this choice disregards visual information, which may favor a different image. The location of salient features, occluders, different camera angles, and noise (e.g., sun glare) may render another image best. We therefore implement both pose– and appearance–based coarse alignment methods and then evaluate them to identify the best approach for our application.

5.1 Pose–Based Coarse Alignment

Pose–based coarse alignment identifies similar viewpoints between two surveys using the estimated boat pose. A simple way to do this is to calculate d as a function of the boat poses, i.e., enforce that the position and the heading are similar. A slight variation on this approach is to calculate d as a function of the boat positions and the observed shore points, which is what we use. Either the raw sensor measurements (standard GPS and compass) or the pose estimated by visual SLAM allow us to compute this type of coarse alignment.

We use visual SLAM to compensate for the fact that the boat's sensors are noisy. All sensors have limited precision and sometimes output incorrect values. Variation in a measurement depends on the sensor it came from and environment factors. Up to 2.5m fluctuations in the boat's position from GPS are due to typical atmospheric variation, but environment interference (e.g., overhead trees) can sometimes skew it farther. Up to 10 degree deviations can occur in the boat's orientation from the compass during peak motor currents. The IMU captures the boat's angular velocity to within 1 deg/s after bias estimation and removal, which helps estimate the distance traveled. Distance traveled is

also a function of the positions of tracked visual features in the image sequence. To measure it, we extract up to 300 Harris points from a roughly uniform 12x20 grid over each image and then track them as long as possible. The accuracy of visual tracking is limited by the image resolution (704x480) and tracking errors caused by occlusions between branches.

Each survey represents a sequence of sensor data ordered in time, which we fuse together using a factor graph. A factor graph is standard for graph–based approaches to SLAM. Figure 6 depicts how we construct it. The colored nodes represent the variables to be estimated, including the boat's 6D pose, x_t, the 3D position of each tracked visual landmark, l_j, and the boat's 6D velocity, v_t. Our inclusion of the boat's velocity makes this factor graph different from that typically used for mobile robots. Unlike a wheeled robot, the jet thrusters on our ASV do not measure odometry. To accommodate that, the boat's relatively constant velocity provides an alternative way to constrain the distance traveled between poses.

The values of the variables are constrained by the factors in the graph, which are depicted as black nodes in the figure. A factor that corresponds to a sensor reading constrains the values of its neighbor nodes to be a function of the sensor's precision. These are the pose constraints, p_t, from the GPS and the compass, the velocity constraints, q_t and r_t, from the IMU and the boat's relatively constant speed, and the landmark constraints, m_t^j, from the visual feature tracker. An additional factor, u_t, uses the boat's previous velocity v_{t-1} and the boat's previous pose x_{t-1} to constrain the boat's pose at x_t.

Any particular assignment to the variables in the factor graph corresponds to some error, which we want to minimize. Without other information, a single variable is best estimated using the value of the sensor that measures it (e.g., set x_t to p_t). Indeed, this is how we estimate the initial value of each variable. As measurements from many different sensors are added over time, however, the initial estimates may no longer best fit all the constraints. The most accurate estimate of the boat's position and the landmarks maximizes the agreement among all the constraints in the factor graph, rather than that of a single local constraint.

We achieve a globally optimized estimate of the boat poses and the landmark positions using iSAM2 [Kaess et al., 2012], which is part of the GTSAM bundle adjustment framework [Dellaert, 2012]. Kaess et al. [2012] introduced the Bayes tree data structure and then used it to create iSAM2, which performs incremental smoothing and mapping as the factor graph is constructed. New measurements only affect a subset of the Bayes tree. This subtree is reinterpreted as a factor graph, appended with the new factors, reordered, eliminated into a new Bayes tree, and then inserted back into the existing Bayes tree. Cholesky factorization is used to solve for an update to the value of each variable.

Our factor graph also includes *smart factors* to speed up optimization [Carlone et al., 2014]. The dotted–line in Fig. 6 depicts one smart factor. Each one eliminates a landmark and its supporting measurements from the factor graph. This is a significant reduction: up to 300 landmarks are observed per pose, each of which is observed by up to roughly 50 poses. The Schur complement enables this reduction. In addition to faster optimization, degenerate landmarks are easily detected because the optimization of each landmark is isolated from the rest of the optimization.

5.2 Trajectory and Map Accuracy

SLAM produces a trajectory and a map, which are more accurate than when computed from raw sensor data. This is crucial because our framework is in several ways based on the assumption that these are accurate. The trajectory provides the coarse alignment between surveys. It is also used to maximize coverage for survey comparisons. The map allows us to mask out the sky and the water for precisely aligning lakeshore images. The function of optimization is to significantly improve the accuracy of these values.

The improvement in accuracy due to optimization is measured using the reprojection error. The reprojection error of a landmark is the L2 distance between its reprojected 2D position and its original 2D pixel location. Before optimization, the reprojected 2D position is only a function of the measurements from its own feature track. After optimization, however, the boat's trajectory is pulled in line with all the feature tracks, and the position of each 3D landmark is made consistent with the rest of the variables in the graph.

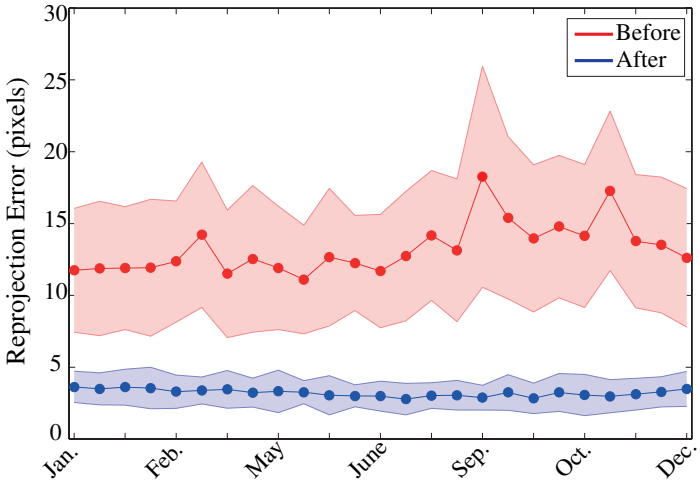


Figure 7: Average reprojection error before and after applying visual SLAM to surveys from 2014.

The comparison is shown in Fig. 7. The reprojection error consistently averages 12-15 pixels before optimization and 3 pixels after. The standard deviation is also much higher before optimization. Because these accuracies are tied to GPS coordinates, after optimization we can more reliably identify images with the most overlap from different surveys. Before optimization, in contrast, the accuracy is low enough that two images from similar camera poses sometimes have little overlap.

5.3 Appearance–Based Coarse Alignment

An alternative to pose–based coarse alignment is to rely primarily on the lakeshore's appearance. The challenge here is scene matching, which is related to our goal of correspondence, but is simpler because full pixel–level alignment is unnecessary at this stage. There are multiple ways to do this. Point–based features such as SIFT, SURF, and ORB have long proven effective at recognition, but have been shown to have limited application outdoors. The features from some convolutional neural networks have proven to work well outdoors, but have been shown to utilize man–made rather than natural structures. SIFT Flow is often robust to the extreme variation in appearance of natural environments, which makes it worth considering here. We thus evaluate all three approaches in the VideoSnapping sequence alignment framework [Wang et al., 2014].

The property of surveys that they are captured along approximately the same path makes them suitable for appearance–based sequence alignment. Following Wang et al. [2014] and Naseer et al. [2015], this involves constructing a cost (or similarity) matrix for every pair of images using some function to measure difference (or similarity) in appearance. Several optimizations are made to tailor sequence alignment to the environment monitoring scenario.

We can avoid computing the full cost matrix due to the fact that our image sequences are captured by a robot. Each image is associated with a known pose, which has a globally consistent reference frame due to GPS and the compass. Only images from a similar boat pose (within 20 m and 20 degrees) need to be compared.

We also subsample our surveys to reduce the time to compute a cost matrix. Our robot travels at roughly 0.4 m/s while capturing images at 10 fps, which leads to many redundant images. There are about 30x too many. Thus, surveys were down-sampled to one image every three seconds.

Finally, somewhat low resolution images (704x480) were used in each individual image comparison. This limited the required storage space during data collection. As a secondary consequence, however, it may have affected the number of quality matches in the cost matrix. To counter this, the performance of all feature detectors in OpenCV (e.g., SIFT, SURF, and BRIEF) were evaluated, and ORB was used because it performed best.

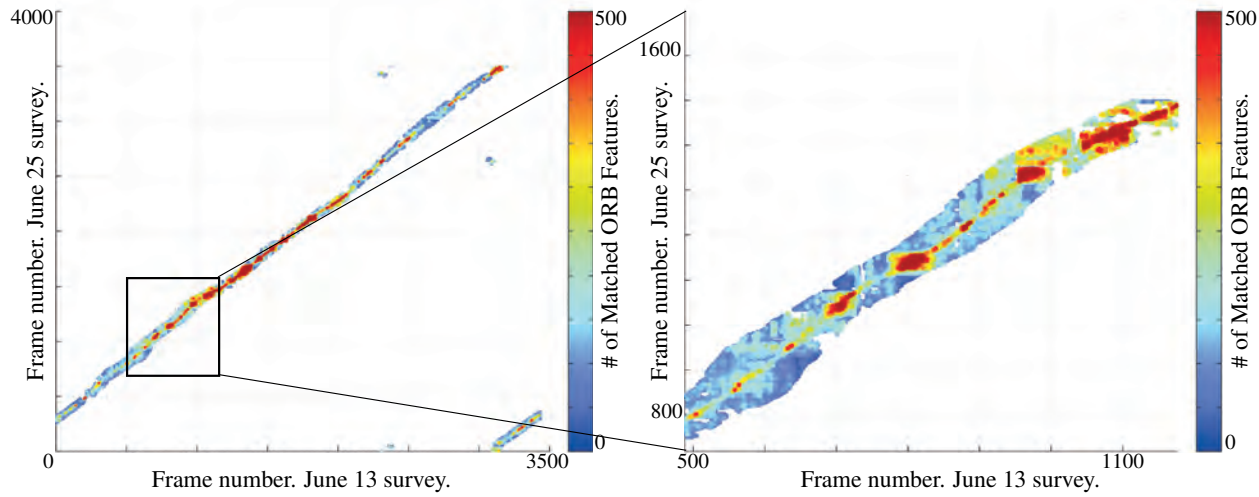


Figure 8: Similarity matrix of the number of matched ORB features between the June 13th and the June 25th surveys. Only the images at roughly the same locations in both are compared, which are colored dark–red (highest–matching) to dark–blue (lowest–matching). Areas with excessive pose differences are colored white. A close up of a matched region is shown on the right.

The above optimizations reduce the time required to create the cost matrix, which is useful for aligning many long surveys. Full sequence alignment for 27x26 = 702 cost matrices completed in five days on approximately 10 Intel Xeon PCs with 8 cores each. One matrix is shown in Fig. 8. There, because the number of matched ORB features is used as the metric, which we want to maximize, it is a similarity matrix. Only a small portion of images had to be compared. The discernible ridge of high match scores indicates where coarse alignment may be best. The comparison of all our coarse alignment approaches on all of the 702 survey alignments is presented in the next section.

In addition to ORB features, following Naseer et al. [2015] and Sünderhauf et al. [2015], we also used the output from a layer of a convolutional neural network (CNN) that was trained for recognition tasks. The output from a single layer of the deep network is used as a whole image descriptor. The distance between two descriptors is computed using the normalized dot–product. This method is referred to as Conv3 in our evaluation.

Prior to evaluating Conv3, we searched for the convolutional neural network layer that would be most discriminative on our dataset. Following Sünderhauf et al. [2015] and Naseer et al. [2015], we looked at various layers of pre-trained convolutional neural networks provided by the Caffe library [Jia et al., 2014]. Several models use the network topology of the BVLC Reference CaffeNet trained on various datasets. We compared all five convolutional layers in the Caffe reference topology using several cost matrices generated on our dataset, and like Sünderhauf et al. [2015], found that layers three and four were the most discriminative. We also tested a network pre-trained on a database of places [Zhou et al., 2014], and a hybrid network pre-trained on both the place database and a collection of objects. We ultimately used a layer from the hybrid network due to its slightly better performance.

Although SIFT Flow performs image registration (see Section 6.2), it can be used for scene matching and included in the set of coarse–alignment methods to be compared. Given two input images, SIFT Flow outputs both an alignment and a matching energy. A low energy indicates the image pair was successfully aligned. For this procedure, low resolution images (44x30) are used because the alignment is fast to compute, yet is still indicative of a matched scene.

5.4 Comparison of Coarse Alignment Methods

We evaluated the five different coarse alignment methods using human–labeled alignments as the baseline. Unfortunately, our dataset is too large for humans to manually align all the surveys. We therefore created two reference sets: 1) complete coarse alignment of the surveys from June 13 and June 25; and 2) partial coarse alignment of the survey

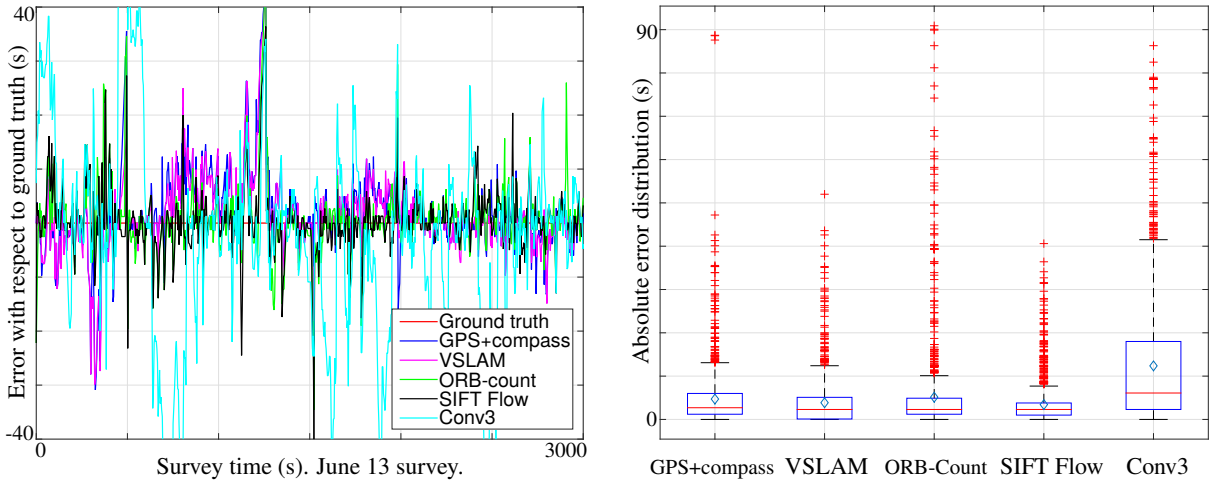


Figure 9: Alignment accuracy of five different methods on the coarse alignment of surveys from June 13 and June 25, using manually labeled images as the ground truth. Accuracy is measured in seconds offset from the ground truth, which corresponds to the number of frames in the 1 Hz video sequence. **left)** Accuracy per method over time. **right)** Distribution of the absolute error over the full survey for each method. The red line indicates the median of the absolute error and the diamond the mean.

from June 25 to the other 27. The second set consisted of two separate sections of the lakeshore (the first with more vegetation, the second with a building in the background). Each required 200 labels, whereas the complete survey required about 4000 labels. To speed up the labeling task, users started off at the closest image as determined by the GPS and the compass, and then scrolled forward or backward through the survey until they found the image pair that matched best.

The accuracy of all five approaches in coarsely aligning two surveys to one another (i.e., reference set one) is shown in Fig. 9. Somewhat surprisingly, the Conv3 method performed much worse than the others. Among the rest, this analysis shows no discernible improvement in accuracy in using one approach instead of another. Although all the approaches have outliers, they all deviate from the ground truth fairly consistently over the complete survey.

There is, however, significant deviation in all the methods at about image 1380. Outliers occurred there because the operator took control of the boat to avoid fishing lines. During the maneuver, the boat was moved somewhat perpendicularly to the shore, which is inconsistent with how the boat moved on its own in the other surveys. Subsequently, the hand–picked image pairs in this range are more subjective.

We expected the appearance–based approaches to perform well because few changes could have accumulated between the two surveys in two weeks. Indeed, two of the three appearance–based approaches performed well, on average. Some areas were, however, easier to match than others due to the availability of discriminative features. This particularly affected the Conv3 method, whose performance is consistent with the observations of Naseer et al. [2015]. In their work, the Conv3 layer decreased the saliency of vegetation in order to boost the recognition of streets and buildings in the midst of seasonal variations.

Although two appearance–based solutions worked well in the case of a short time interval between two surveys, this trend did not hold for the coarse alignment of all the surveys, as shown in Fig. 10 and in Fig. 11. Figure 10 is the evaluation for the sequence of shore with mostly vegetation, Fig. 11 the sequence with a building in the background. On both sequences, GPS+compass and visual SLAM performed comparably, SIFT Flow a bit worse, ORB-Count still worse, and Conv3 unreliably. SIFT Flow performed better on the sequence with vegetation than on the sequence with the building in the background. As expected, the performance of ORB-count decreased in proportion to the increasing interval of time between surveys. This effect was also somewhat visible with SIFT Flow and held true for most surveys, except those around the end of July.

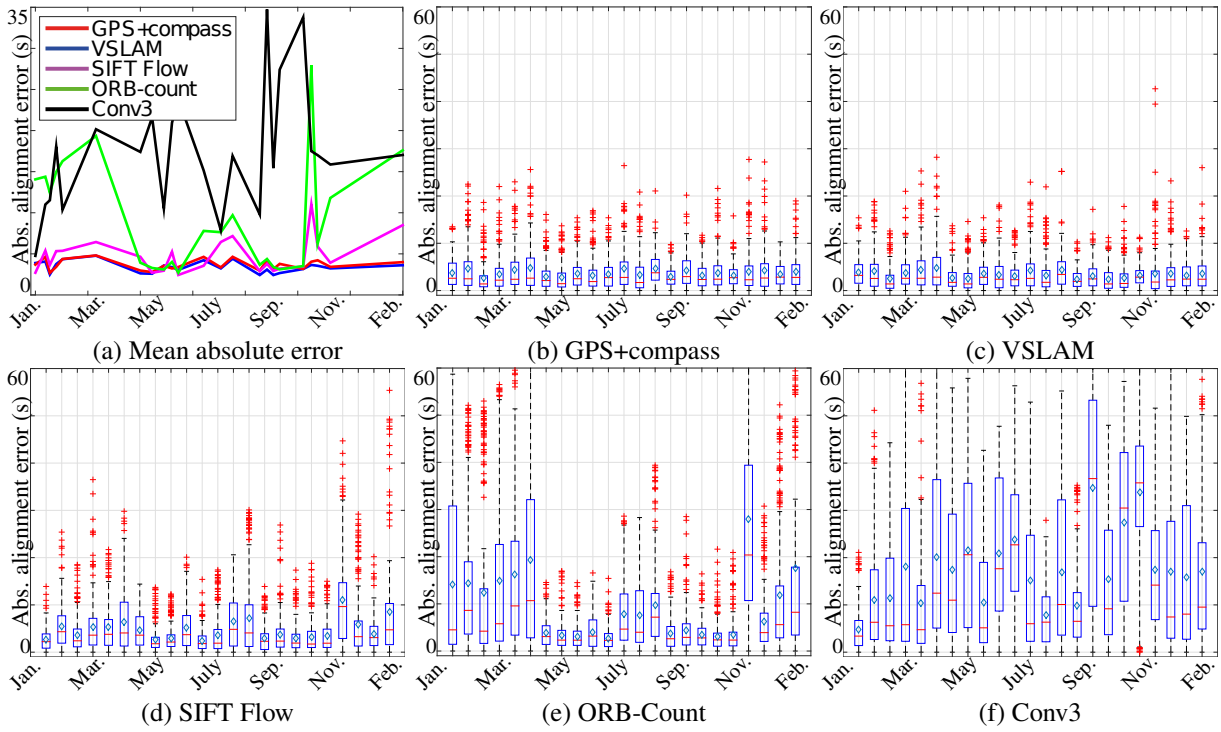


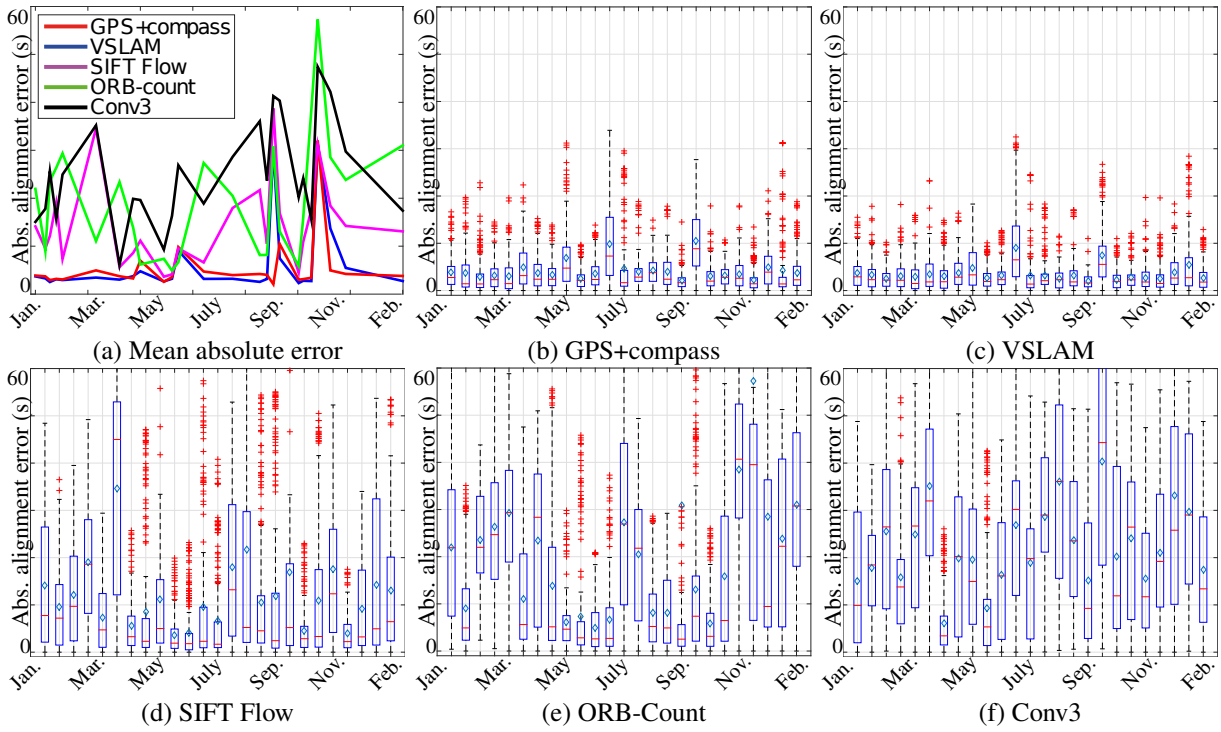
Figure 10: Quantitative comparison of the five different coarse–alignment methods for sequence one, on which the June 25th survey was aligned with 24 other surveys. The accuracy of each method was measured as the offset to the human–labeled matches.

The inaccurate appearance–based alignment of the July and August surveys was due to the lakeshore's changing appearance. In July, the higher water level occluded part of the shore and resulted in the rest being captured from a different viewing angle. In August, significant sun glare undermined the methods. These sources of noise appeared in a few other lower performing alignments as well. Compared to ORB-count, SIFT Flow's performance on sequence 1 shows that it retained a bit more robustness to variation in appearance for scene matching.

There are several instances where SIFT Flow appeared to perform somewhat inconsistently. In sequence two, for example, the alignment of the Feb. 5 survey was much better than the Mar. 14 survey. Although the vegetation changed across both survey comparisons, the lighting conditions were similar on Feb. 5 and on June 25 (overcast). In contrast, the bright sky on Mar. 14 added shadows to the largely changed vegetation, which made the scene appear very different from a computer vision perspective.

Some of the variation in alignment performance between the two sequences is due to the subjectivity of the human labeling. The first sequence has strong features in the foreground with an ascending hill and weaker features in the background. The second sequence has, in contrast, strong features in the background (buildings) and weak features in the foreground (reeds, grass shores). While the weak foreground features were mostly discernible, the labelers tended to favor images that aligned with the background features. Many other outliers correspond to visually subjective situations in which the boat detoured a tree and observed it from different viewpoints.

The fact that ORB–count performed well across intervals of a couple months was surprising because it contradicted some of our preliminary analyses, which thus warranted a more comprehensive evaluation than those shown in Fig 9-11. Although we had the coarse alignments for all 702 survey comparisons, we did not have a human–labeled reference set of this magnitude. Instead, because visual SLAM's performance proved accurate and stable, we measured the performance of ORB-count and SIFT Flow against that (ORB again outperformed other point–based features, e.g., BRIEF, in this evaluation). The accuracy of both is calculated as a function of the number of weeks between surveys to capture how robust each method is to variation in appearance.



(a) Mean absolute error (b) GPS+compass (c) VSLAM

(d) SIFT Flow (e) ORB-Count (f) Conv3

Figure 11: Quantitative comparison of the five different coarse–alignment methods for sequence two, on which the June 25th survey was aligned with 25 other surveys. The human–labeled images provided the ground truth against which the accuracy of each method was measured.

Figure 12 shows the results. Both approaches lose data association power over time, up to six months later. The SIFT Flow coarse alignment approach is, on average, more robust than ORB–count to the variation in appearance that accumulates between surveys over time. Whole image alignment using SIFT Flow captures scene structure, which allows it to match difficult scenes more reliably (see Section 2).

The parabolic shape of the curves indicates that aligning the previous year's surveys to new surveys may be possible. More precise alignments are found between images from the same season than across seasons, even if the images were captured from different years. For example, many images from the March 2014 survey align well with images from the Feb 2015 survey. Because our evaluation was limited to a year of data, there is less data toward 52 weeks, which caused the variation in performance near the end of the graph. This is also visible in the standard deviation, which shrinks over time.

6 Precise Survey Alignment

The objective of precise survey alignment is to construct a flow field that warps one image to its coarsely aligned pair such that scene elements match all the way down to their pixels. Even for relatively low resolution images (e.g., 320x240) this can be a computationally expensive process (~30 seconds). However, it is unnecessary to perform this step for all the image pairs between two surveys. In Section 6.1 we show how to select a subset of image pairs while retaining maximum coverage of the shore, which reduces the computation time. Images are precisely aligned using SIFT Flow, which is described in Section 6.2. We evaluate the alignment quality in the subsequent sections.

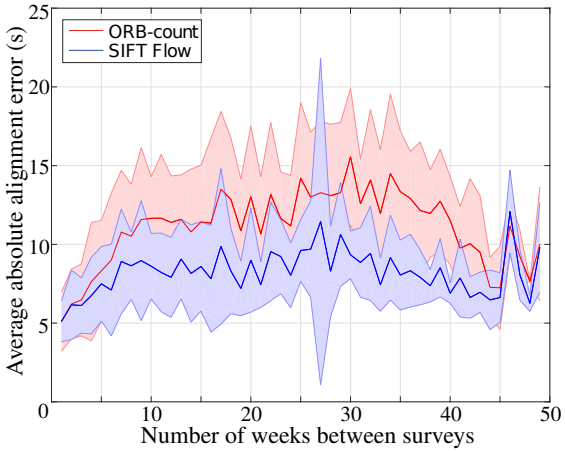


Figure 12: Accuracy of appearance–based coarse survey alignment as the number of weeks between surveys is increased. Visual SLAM provided the reference against which we measured alignment error.

6.1 Selecting a Minimum View Set

To reduce the computational overhead of dense image registration (Section 6.2) and to enable a manual comparison between two surveys (Section 7), we select a minimum view set from among the roughly 4,000 image pairs of coarse survey alignment. Initially, a large set of images in each survey is desirable for the feature tracking step of visual SLAM and to reduce motion blur. Yet, the redundancy in the images makes survey comparison cumbersome. Ideally, a person comparing two surveys would only see a subset of these images, where each corresponds to a unique section of the shore. It is desirable to find a minimal subset of images that covers as much lakeshore as is seen in both surveys.

Another name for this is the "Set Cover Problem" (SCP) [Chvatal, 1979], which can be expressed as follows. Let $\mathcal{S}$ be the set of all the observable positions in a survey of a lakeshore. Each camera pose, i, of the survey observes a subset $\mathcal{I}_i$ of these shore points, where $\mathcal{S} = \bigcup_{i \in I} \mathcal{I}_i$. The goal is to find a set of poses J for which $\mathcal{S} = \bigcup_{j \in J} \mathcal{I}_j$ and $|J|$ is as small as possible. This Set Cover Problem is known to be NP-Hard in general. It can be approximated using linear programming or a simple greedy approach, which gives sufficient performance for our application.

The set of shore points that compose $\mathcal{S}$ is identified using the optimized poses from visual SLAM. Because the robot is controlled to move at a constant distance d from the shore, every point $d \pm \epsilon$ away in the camera frustum is considered part of it (in practice, we chose $d = 10$m and $\epsilon = 1$m). To get a discrete set of shore points, the shore map is rasterized into a grid, in which each non-zero cell represents the shore. An arc centered on a pose is drawn with radius d and thickness ϵ with an angle consistent with the camera intrinsic parameters. The shore points from two different surveys and their overlap are shown in Fig. 13.

Several camera poses are eliminated due to practical constraints before the minimal cover set can be identified. Poses with an invalid camera configuration[2] or with a high likelihood of motion blur[3] are rejected. Poses from two different surveys without a similar view of the lakeshore are also rejected. In this paper, we found that two poses have a similar view if their 3D positions are close and the intersections of both their camera axes with the shore are close as well. The distance between the camera angles is expressed in this way to keep comparable values with the distance between the 3D positions.

The Set Cover Problem is solved using the greedy algorithm shown in Algorithm 1. The method provided the results illustrated in Fig. 14 in less than 30 seconds on a laptop PC. For a coarse survey alignment with 4,000 image pairs, about 200 of those will be selected for the cover set of the shore, which corresponds to over an order of magnitude fewer image pairs to register. Note that the set of images does not view the entire shore; only all the shore points seen

[2] Some images were recorded before the camera pan and tilt were set, and some during the transition between monitoring the lakeshore and the island.

[3] predicted using the IMU rotation speed

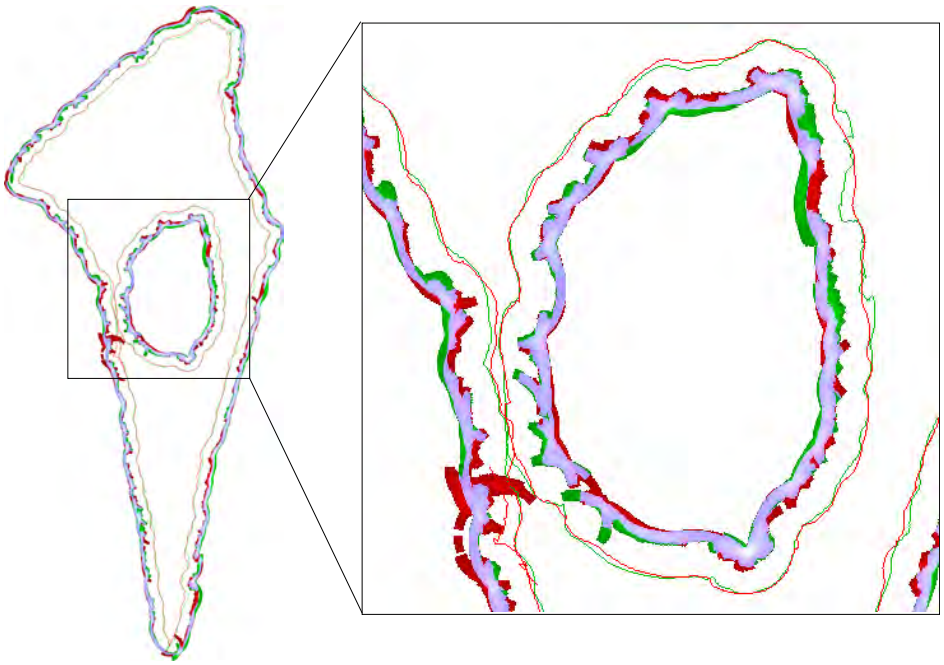


Figure 13: Optimized trajectories of the boat and the extrapolated shore points in view from each pose for two surveys. The boat's trajectory is represented with thin lines. The shore points are the larger swaths. Red and green delineate two different surveys while mauve indicates viewpoint overlap. A close–up of the island is shown on the right.

from both surveys with similar views can be covered (in our dataset, this is approximately $88\% \pm 14\%$ of the shore line).

Let L be the list of selected viewpoints, initially empty;
while *there are shore points to observe* **do**
    Select the valid shore point P which is the least observed;
    Let V be a viewpoint such that
        V observes P and;
        V observes the largest number of unobserved shore points;
    Remove P and all shore points observed in V from the list of points to observe;
    Append V to L;
end
return L

Algorithm 1: Greedy algorithm for maximizing the coverage of the shore with a minimal number of poses.

6.2 Image Registration

Given two poses that view approximately the same scene from two different surveys, we run image registration in a local search of several nearby images, and output the image pair with the best alignment score we find (the registration process for a single image pair is shown in Fig. 1). Image registration is performed using a tailored version of the SIFT Flow scene alignment algorithm [Liu et al., 2011], which can find dense correspondence between images that have significant amounts of variation between them.

The algorithm constructs a Markov Random Field (MRF) to represent matching information and constraints. A dense image of SIFT descriptors (see Lowe [2004]) define the matching pattern (the data terms of the MRF) to be optimized between two images. The flow at a pixel is spatially constrained by adjacent flows (a smoothness criterion), and lower

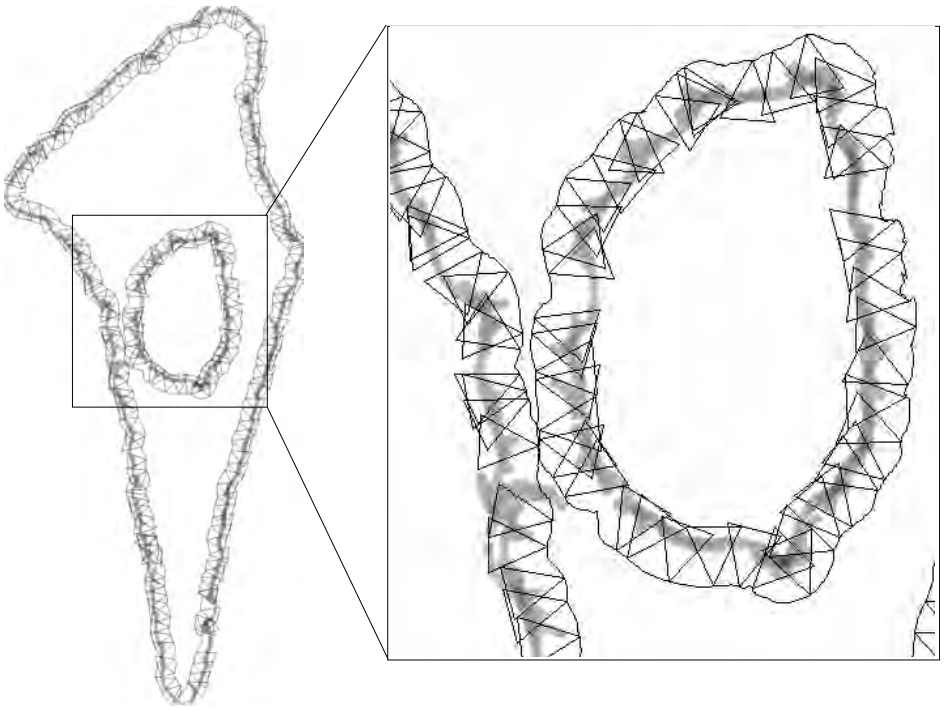


Figure 14: Cover set for the surveys in Fig. 13, shown as a collection of camera frustums. The poses shown are a subset from the red survey, which are selected to account for co-visibility with the green survey. A close-up of the island is shown on the right.

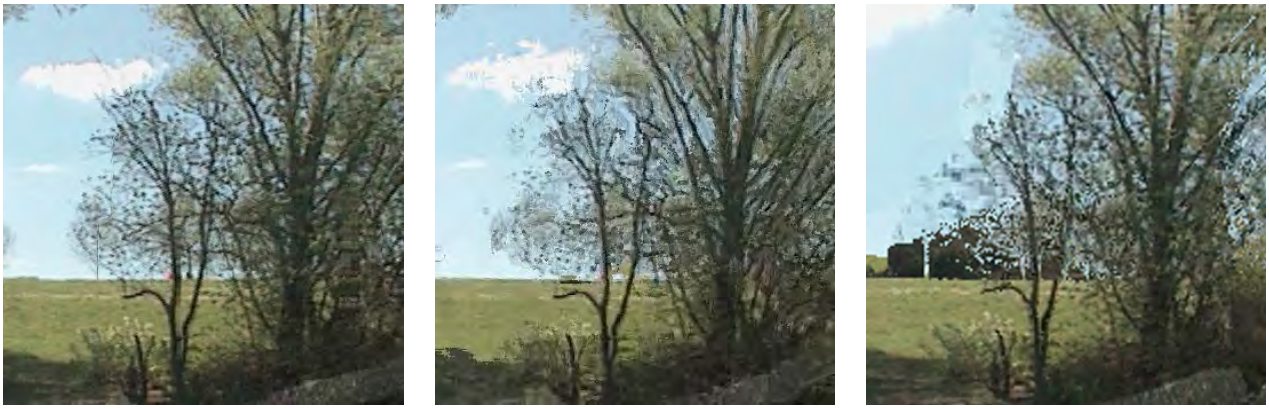
magnitudes of flow are favored (regularization). For two images of approximately the same scene, the alignment score is minimized when the flow lines up salient structures between them.

We add a bias to the image alignment process in favor of aligning the lakeshore rather than the sky or the water. As is, SIFT Flow's cost function is designed to align the contents of a scene indiscriminately. Unfortunately, most of our images are a majority sky and water, which add noise because they retain little consistent structure between surveys. Salient structure can appear in the water if it is reflective, which reduces the likelihood of a good alignment because the reflectivity of the water changes between surveys. The varied appearance of the sky also affects the alignment.

The bias to SIFT Flow's cost function is derived from the optimized map from visual SLAM. The location of the sky and the water in each image is estimated using the landmark positions. Although most points are on the shore because most corner features occur there, some are occasionally identified in the sky and the water, which this process filters out. Points with a negative elevation usually correspond to the water. Points far away usually correspond to the sky. The rest are interpreted as part of the lakeshore. Given an image and the set of 3D landmarks it observes, an image mask is created by projecting the points onto the image as a circle (with radius r= 28, which gave the best performance). For each pixel in the non-zero regions, the data terms of SIFT Flow's objective function are biased (by a factor of 1.5) to align the contents there compared to the other areas of the image.

Image pyramids speed up the process of finding the best alignment between two images. The MRF defined by SIFT Flow is solved using belief propagation, which can require a significant amount of computation time to converge in large graphical models. An image pyramid halves the size of both images for several layers (four in this paper, as in [Liu et al., 2011]). The search for the best alignment starts at the top and proceeds down the image pyramid, with the output at each layer bootstrapping the optimization at the next higher resolution. A search window defines the correspondence hypothesis space considered for each pixel, which is reduced in size with each successive layer.

A local search is performed around the two candidate poses to find the two images that align best before computing



SIFT Flow | Deformable Spatial Pyramids | Daisy Filter Flow

Figure 15: Image registration using SIFT Flow, Deformable Spatial Pyramids, and Daisy Filter Flow. The pixel-level alignment quality is often higher using SIFT Flow in our application.

the full resolution output alignment. SIFT Flow seldom finds a dense correspondence between the first two coarsely aligned images we give it. The perspective difference between the two images is often different enough that an incorrect, high score alignment is found. A better, low score alignment is usually possible between nearby images, which have a slightly different perspective. Thus, images at 0, $\pm$ 1.5 , and $\pm$ 3.0 second offsets from the two image candidates are considered, for a total of 25 different alignments. To speed up the search, only images at the top layer of the image pyramid are aligned.

6.3 Comparison of Image Registration Methods

Several image registration techniques are among the state-of-the-art, which motivates their comparison to show why we used SIFT Flow. We compare SIFT Flow to Deformable Spatial Pyramids (DSP) [Kim et al., 2013] and Daisy Filter Flow (DFF) [Yang et al., 2014]. In our analyses, SIFT Flow produced many high quality alignments, but it can be slow for full–image alignment. One of the primary contributions of DSP is to address the slow runtime of SIFT Flow. It is also robust to differences in scale. DFF has the advantage that it is robust to changes in rotation. This comparison applies the three methods to the same image pair to show their typical alignment quality on images from our dataset.

The results are shown in Fig. 15. Although all three methods find an alignment between the images, the quality of the SIFT Flow alignment is higher than the other two. The fine details retained little spatial coherence in the DSP result, and several artifacts were added in the DFF result. DSP finished in under a second, SIFT Flow in about 30 seconds, and DFF in about two hours (without parallelization).

SIFT Flow may be the best current method for our framework because some of the problems the newer image registration algorithms were designed to fix are relatively infrequent in our monitoring application. Our framework achieves fast alignments (≈ 0.8 seconds) using SIFT Flow by restricting alignment to the top layer of the image pyramid until high–resolution alignments are needed. Over half of the full runtime of SIFT Flow is spent aligning the bottom–most layer, which searches a tiny 3x3 hypothesis space around each pixel. DSP eliminates the spatial constraint at this granularity to gain speed, but as a result it loses spatial coherence in the fine details. Moreover, although DSP is robust to scale, our boat is consistently 10 m away from the shore. Similarly, although DFF is robust to rotations, those are relatively infrequent in our dataset.

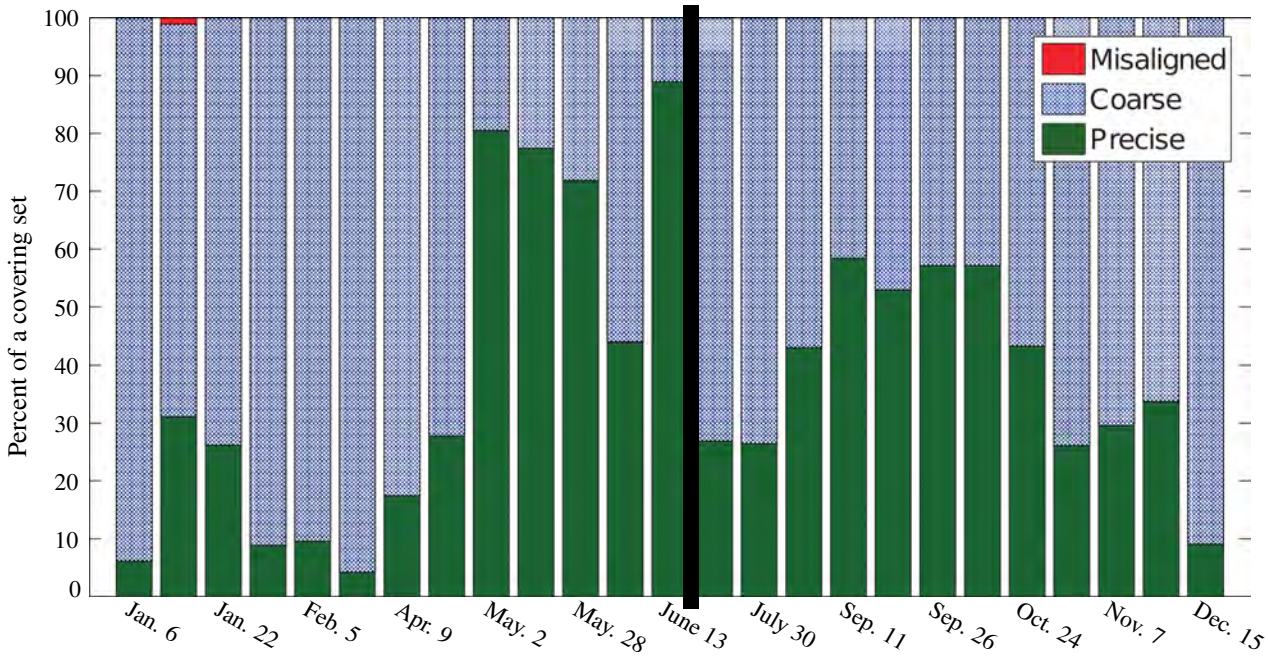


Figure 16: The alignment quality of the survey from June 25 with other surveys from 2014. The vertical bar denotes where the June 25 survey falls among these surveys.

6.4 Alignment Quality Over Time

One of the primary questions of this framework is how well it can align lakeshore surveys, in general, across large spans of time and in the midst of significant environment variation, which we evaluate here. In this analysis we compared the June 25, 2014 survey with all the other surveys from 2014. For each survey, each image in its cover set and the aligned image from the following survey were flickered back–and–forth in a display. The quality of each alignment was manually identified according to three criteria: 1) *precise*—almost the entire image is aligned well with little noise; 2) *coarse*—the images correspond to the same scene and some objects may be precisely aligned; and 3) *misaligned*—the images correspond to different scenes or it is hard to tell they come from the same scene.

The results are shown in Fig. 16. The graph shows that fewer images are precisely aligned if surveys from larger time gaps are compared. This is the trend we expected, in addition to specific cases that are consistent with the results in Fig. 17. The same dips in performance are observed for the two July surveys with high water levels. Thus, the number of precisely aligned images decreased in proportion to the accumulated variation between surveys. Given the mid-summer survey, this meant that surveys between mid-spring and early fall mostly aligned well.

Several of the winter surveys had few precise alignments with the mid-summer survey. The coarsely aligned images always captured the same view of the lakeshore, but image registration failed. Large artifacts added the biggest source of error. Patches of the images often aligned well while other areas aligned with replicated patches. In many of these cases, however, a human would also have difficulty identifying a precise alignment. For example, barren trees with visible background corresponds as well as noise to the fully leafed trees.

6.5 Alignment Quality Across Consecutive Surveys

Because some monitoring applications may only require survey alignment between consecutive surveys, the alignment quality across consecutive surveys is evaluated. We chose ten surveys for this analysis and aligned them in consecutive order using our framework. They span a total time of 30 weeks, with one week the shortest interval of time between compared surveys and nine weeks the longest. A human evaluated the alignment quality.

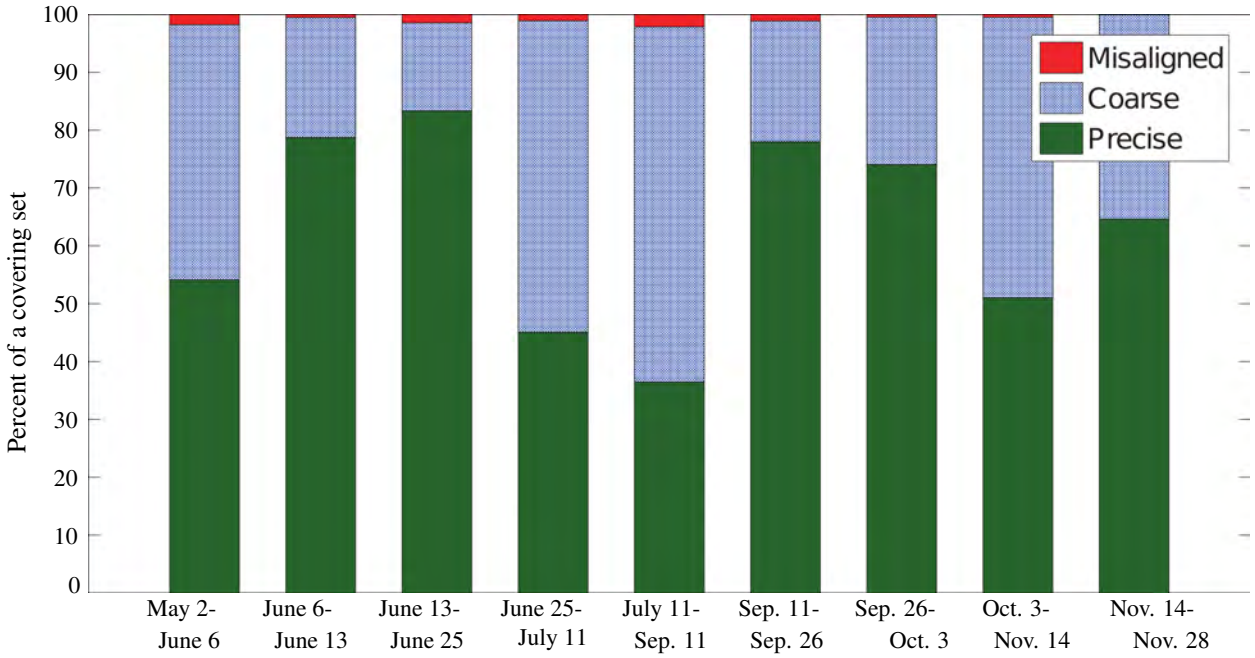


Figure 17: Alignment quality for comparisons of 10 different surveys. All 10 were captured in 2014.

The results are shown in Fig. 17. A large number of precise alignments are found in all the comparisons, although some have more than others. The two cases with the fewest precise alignments involve a comparison with the July 11 survey, which captured a high water level. The upper half of many images in these two comparisons were precisely aligned. Yet, because the perspective and the shoreline appearance significantly changed between surveys, SIFT Flow inaccurately extended the shore downward to try to compensate for the large differences in appearance. In the other comparisons, fewer precise alignments are due to sun glare and larger intervals of time between surveys (i.e., the seasonal variation of plants). The few misalignments indicates an end user was almost always shown images of the same scene.

7 Manual Change Detection

Although we endeavor to create a system for fully autonomous lakeshore monitoring, including detecting changes autonomously, in this work change detection was left to an end user. Our user interface exploits human skill at detecting changes in flickering images of a scene. If an image pair from two different surveys is aligned at the pixel level, changes flash on and off when the images are flickered back and forth. If the precise alignment is not possible, a user can always revert to a side-by-side comparison of images. This approach enables a human to readily detect changes (often requiring only a single flicker) for a survey comparison of a large spatial environment consisting of hundreds of images. The following three sections describe what our users saw when comparing surveys, including changes, variation in appearance, and alignment artifacts.

7.1 Detected Changes

While labeling the alignment quality of each comparison, several changes between surveys were also found, which represent uses of our approach for change detection. Six examples are shown in Fig. 18. Five were found in precisely aligned images; the removed treetop was identified in coarsely aligned images. Although the image pairs in the figure are hard to delineate as precisely or coarsely aligned, this distinction is readily apparent in the flickering display. Flickering the images made it possible to detect small changes, like the cut branch. This and the other changes were unknown to us before the analysis. In fact, although we saw a tree floating in the water after some heavy rains (as can be seen protruding the water in the Sky and Water example of Fig. 19), we did not know where it came from.

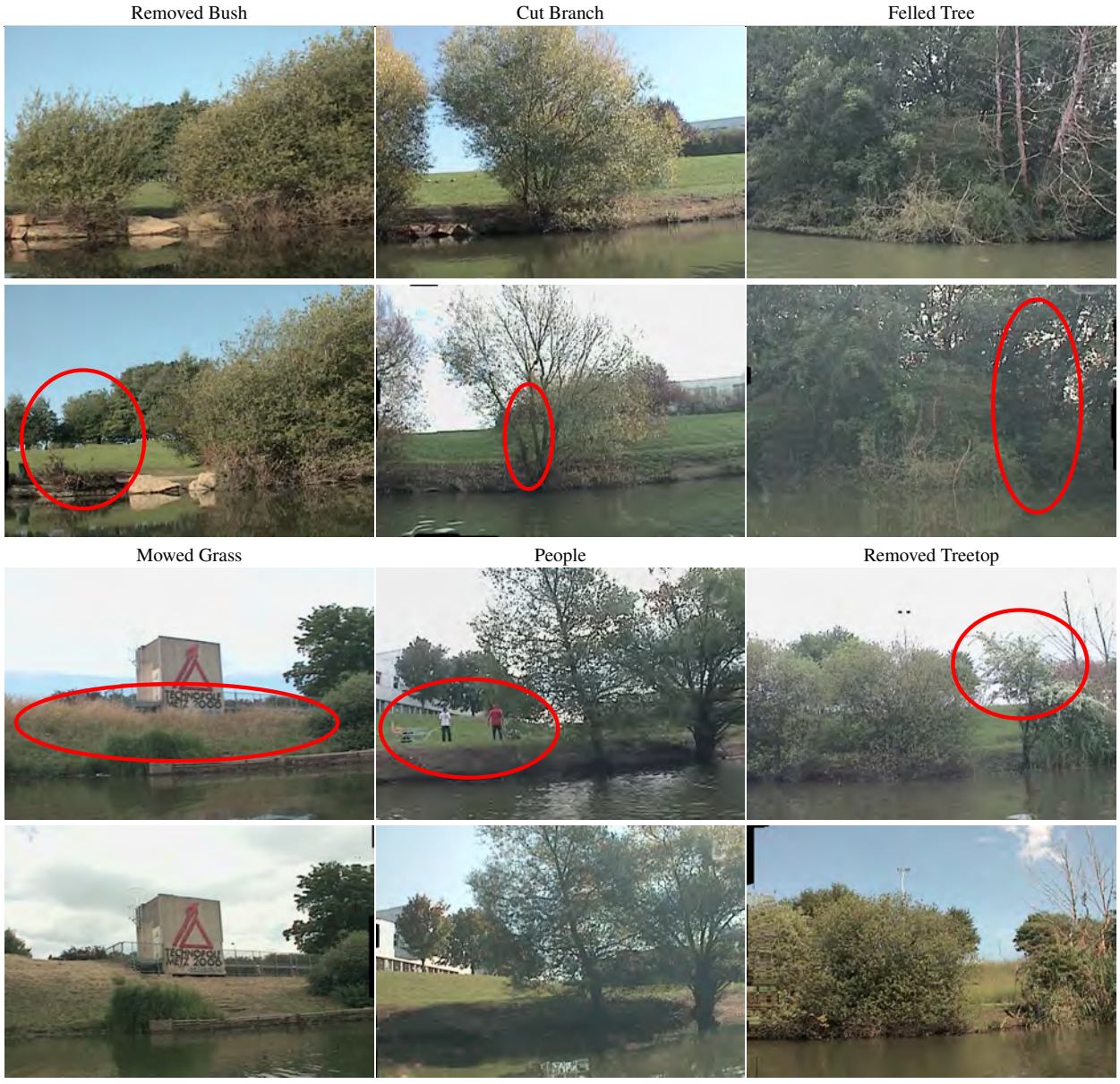


Figure 18: Six notable changes a human found while comparing the 10 different lakeshore surveys.

7.2 Robustness to Different Sources of Variation

Our framework finds many scene correspondences across months of variation, yet the extent of the variation to which our framework is robust has not yet been shown. Clearly, natural environments like lakeshores have a significant amount more variation than most indoor environments. In our dataset, for example, all four seasons are captured, albeit a mild winter. Presumably, our framework achieves precise alignments due to its robustness to many different, combined sources of variation in appearance.

Six prototypical examples of robustness to a particular source of variation are shown in Fig. 19. Before two images are precisely aligned the variation in appearance between them is often significant. Perhaps the example with the most variation is the one labeled 'seasonal'. Foliage was lost before the second image was captured, and the images have different illumination, sky, water, shadows, and a globe reflection. Results from Section 6.4 indicate that a precise alignment may not have been possible if the time interval was larger or there was also sun glare.

7.3 Image Registration Errors

Each coarsely aligned image pair indicates a failure case of image registration, but the blanket "coarse" term masks the magnitude of the error, which we show here. Given that coarse survey alignment almost always identifies image pairs of the same scene, the majority of alignment errors are due to mismatches of image registration. Mismatches are unavoidable: the shoreline may become flooded in subsequent surveys, the background may become occluded, plants may bloom, and changes may render some scenes differently. Whole–image alignment can often minimize the effect of this noise. Yet, this technique can also produce odd warping artifacts when it fails, which requires reverting to the unregistered image pair for performing a comparison.

Six common ways image registration failed are shown in Fig. 20. Image registration does not comply with the physics of structures in each warped image, which is apparent in all the cases. This effect has been observed in other image processing work as well (e.g., texture synthesis [Kwatra et al., 2003]). Because each pixel is potentially warped differently than nearby pixels, the warp may be inconsistent across the image. Additionally, SIFT Flow may try to align scene structures to noise (e.g., sun glare) and changes (e.g., a high-water level water), obfuscating the scene. Out of all failure cases, most alignments are labeled "coarse" because they are translated versions of the same scenes.

8 Conclusion and Future Work

This paper presented a survey registration framework for long–term natural environment monitoring tasks, which was found to be more reliable than other state-of-the-art techniques. In comparing pose– and appearance–based techniques for aligning surveys on a coarse image-to-image level, we found that a pose–based approach using visual SLAM was, on average, closest to what a human would choose. In comparing techniques for aligning images on a precise pixel-to-pixel level, we found that SIFT Flow produced higher quality alignments. After using these two methods to register surveys, a human was able to readily spot several changes that would have otherwise gone unnoticed.

Our evaluations support the conclusion that visual data association is more robust to variation in appearance the more it relies on the structure of the environment. This was apparent in the way humans found feature correspondences between scenes with large appearance variations. It was also apparent in the comparison of image registration and keypoints for appearance–based sequence alignment. Furthermore, the use of geometric information of scenes led to improved image registrations, in that the water and the sky could be masked out. These results are due to the fact that scene structures (e.g., the landscape, trees, and rocks) are among its most consistent features.

In making extensive use of scene structure for data association, our framework obtained a significant number of precise alignments, particularly between surveys captured within three months of one another. In these cases, our framework often demonstrated robustness to the marked variation in appearance that is typical of lakeshore environments. For larger durations, however, the proportion of precise alignments decreased significantly, which indicates that our dataset

Figure 19: Six different sources of noise and precisely aligned image pairs, which show that our approach is often robust to ‘extreme’ sources of variation in appearance.

Figure 20: Six different alignment errors made during image registration.

captures a representative real–world environment that is still a formidable challenge to be overcome in future work.

In future work we plan to improve upon our method's robustness to the variation in appearance between surveys. There are still many coarsely aligned image pairs, which are in reach of becoming precisely aligned. The results in this paper support a transition from a process based mostly on aligning visual features to one that also places significant weight on aligning the 3D structure of the lakeshore. For example, visual SLAM captures the structure of the environment in the map of the shore, but that is underutilized in the image registration process. It may be possible to directly use the map values to anchor image alignment.

Acknowledgments

Funding for this project is provided by the Lorraine Region, France.

References

Bargoti, S., Underwood, J. P., Nieto, J. I., and Sukkarieh, S. (2015). A pipeline for trunk detection in trellis structured apple orchards. *Journal of Field Robotics*, 32(8):1075–1094.

Beall, C. and Dellaert, F. (2014). Appearance-based localization across seasons in a Metric Map. In *6th PPNIV*, Chicago, USA.

Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.

Cabrol, N., Grin, E., Haberle, C., Moersch, J., Jacobsen, R., Sommaruga, R., Fleming, E., Detweiler, A., Echeverria, A., Blanco, Y., et al. (2012). Planetary lake lander: Using technology relevant to titan's exploration to investigate the impact of deglaciation on past and present planetary lakes.

Carlone, L., Dong, J., Fenu, S., Rains, G. C., and Dellaert, F. (2015). Towards 4d crop analysis in precision agriculture: Estimating plant height and crown radius over time via expectation-maximization. In *ICRA Workshop on Robotics in Agriculture*.

Carlone, L., Kira, Z., Beall, C., Indelman, V., and Dellaert, F. (2014). Eliminating Conditionally Independent Sets in

Factor Graphs: A Unifying Perspective based on Smart Factors. In *IEEE International Conference on Robotics and Automation (ICRA)*.

Churchill, W. and Newman, P. (2013). Experience-based navigation for long-term localisation. *IJRR*, 32(14):1645–1661.

Chvatal, V. (1979). A greedy heuristic for the set-covering problem. *Mathematics of operations research*, 4(3):233–235.

Corke, P., Paul, R., Churchill, W., and Newman, P. (2013). Dealing with shadows: Capturing intrinsic scene appearance for image-based outdoor localization. In *IROS*, pages 2085–2092. IEEE.

Cummins, M. and Newman, P. (2008). Fab-map: Probabilistic localization and mapping in the space of appearance. *The International Journal of Robotics Research*, 27(6):647–665.

Dellaert, F. (2012). Factor Graphs and GTSAM: A Hands-on Introduction. Technical Report GT-RIM-CP&R-2012-002, GT RIM.

Engel, J., Schöps, T., and Cremers, D. (2014). Lsd-slam: Large-scale direct monocular slam. In *Computer Vision–ECCV 2014*, pages 834–849. Springer.

Giusti, A., Guzzi, J., Ciresan, D., He, F.-L., Rodriguez, J. P., Fontana, F., Faessler, M., Forster, C., Schmidhuber, J., Di Caro, G., et al. (2015). A machine learning approach to visual perception of forest trails for mobile robots. *IEEE Robotics and Automation Letters*.

Griffith, S., Dellaert, F., and Pradalier, C. (2015). Robot-Enabled Lakeshore Monitoring Using Visual SLAM and SIFT Flow. In *RSS Workshop on Multi-View Geometry in Robotics*.

Griffith, S., Drews, P., and Pradalier, C. (2014). Towards autonomous lakeshore monitoring. In *International Symposium on Experimental Robotics (ISER)*.

Griffith, S. and Pradalier, C. (2015). A spatially and temporally scalable approach for long-term lakeshore monitoring. In *International Conf. On Field And Service Robotics*.

Gu, J., Ramamoorthi, R., Belhumeur, P., and Nayar, S. (2009). Removing image artifacts due to dirty camera lenses and thin occluders. *ACM Transactions on Graphics (TOG)*, 28(5):144.

He, X., Zemel, R., and Mnih, V. (2006). Topological map learning from outdoor image sequences. *JFR*, 23(11-12):1091–1104.

Heidarsson, H. and Sukhatme, G. (2011). Obstacle detection from overhead imagery using self-supervised learning for autonomous surface vehicles. In *IROS*, pages 3160–3165. IEEE.

Hitz, G., Gotovos, A., Pomerleau, F., Garneau, M.-E., Pradalier, C., Krause, A., and Siegwart, R. Y. (2014a). Fully autonomous focused exploration for robotic environmental monitoring. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2658–2664. IEEE.

Hitz, G., Pomerleau, F., Colas, F., and Siegwart, R. (2014b). State estimation for shore monitoring using an autonomous surface vessel. In *International Symposium on Experimental Robotics (ISER)*.

Jain, S., Nuske, S. T., Chambers, A. D., Yoder, L., Cover, H., Chamberlain, L. J., Scherer, S., and Singh, S. (2013). Autonomous river exploration. In *FSR*.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*.

Kaess, M., Johannsson, H., Roberts, R., Ila, V., Leonard, J. J., and Dellaert, F. (2012). iSAM2: Incremental smoothing and mapping using the Bayes tree. *IJRR*, 31(2):216–235.

Kim, J., Liu, C., Sha, F., and Grauman, K. (2013). Deformable spatial pyramid matching for fast dense correspondences. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2307–2314. IEEE.

Košecka, J. (2013). Detecting changes in images of street scenes. In *Computer Vision–ACCV 2012*, volume 7727 of *LNCS*, pages 590–601. Springer.

Krajnik, T., Santos, J. M., and Duckett, T. (2015). Life-long spatio-temporal exploration of dynamic environments. In *Mobile Robots (ECMR), 2015 European Conference on*, pages 1–8. IEEE.

Kularatne, D. and Hsieh, A. (2015). Tracking attracting lagrangian coherent structures in flows. In *Robotics: Science and Systems*.

Kwatra, V., Schödl, A., Essa, I., Turk, G., and Bobick, A. (2003). Graphcut textures: Image and video synthesis using graph cuts. In *ACM Transactions on Graphics (ToG)*, volume 22, pages 277–286. ACM.

Liu, C., Yuen, J., and Torralba, A. (2011). SIFT Flow: Dense correspondence across scenes and its applications. *PAMI*, 33(5):978–994.

Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110.

Lucas, B. D. and Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. In *IJCAI*, volume 81, pages 674–679.

Martin-Brualla, R., Gallup, D., and Seitz, S. M. (2015). Time-lapse mining from internet photos. *ACM Transactions on Graphics (TOG)*, 34(4):62.

McManus, C., Upcroft, B., and Newman, P. (2014). Scene signatures: Localized and point-less features for localization. In *RSS*, Berkeley, USA.

Milford, M., Firn, J., Beattie, J., Jacobson, A., Pepperell, E., Mason, E., Kimlin, M., and Dunbabin, M. (2014). Automated sensory data alignment for environmental and epidermal change monitoring. In *Australasian Conference on Robotics and Automation 2014*, pages 1–10. Australian Robotic and Automation Association.

Milford, M. J. and Wyeth, G. F. (2012). Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 1643–1649. IEEE.

Milford, M. J., Wyeth, G. F., and Rasser, D. (2004). Ratslam: a hippocampal model for simultaneous localization and mapping. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 1, pages 403–408. IEEE.

Naseer, T., Ruhnke, M., Stachniss, C., Spinello, L., and Burgard, W. (2015). Robust visual slam across seasons. In *IROS*.

Nelson, P., Churchill, W., Posner, I., and Newman, P. (2015). From Dusk till Dawn: Localisation at Night using Artificial Light Sources. In *ICRA*.

Neubert, P., Sünderhauf, N., and Protzel, P. (2014). Superpixel-based appearance change prediction for long-term navigation across seasons. *RAS*.

Shi, J. and Tomasi, C. (1994). Good features to track. In *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on*, pages 593–600. IEEE.

Subramanian, A., Gong, X., Riggins, J., Stilwell, D., and Wyatt, C. (2006). Shoreline mapping using an omni-directional camera for autonomous surface vehicle applications. In *OCEANS*, pages 1–6. IEEE.

Sünderhauf, N., Dayoub, F., Shirazi, S., Upcroft, B., and Milford, M. (2015). On the performance of convnet features for place recognition. *arXiv preprint arXiv:1501.04158*.

Sünderhauf, N., Neubert, P., and Protzel, P. (2013). Are we there yet? challenging seqslam on a 3000 km journey across all four seasons. In *Proc. of Workshop on Long-Term Autonomy, IEEE International Conference on Robotics and Automation (ICRA)*, page 2013. Citeseer.

Sunderhauf, N., Shirazi, S., Jacobson, A., Dayoub, F., Pepperell, E., Upcroft, B., and Milford, M. (2015). Place recognition with convnet landmarks: Viewpoint-robust, condition-robust, training-free. *Proceedings of Robotics: Science and Systems XII*.

Wang, O., Schroers, C., Zimmer, H., Gross, M., and Sorkine-Hornung, A. (2014). VideoSnapping: Interactive Synchronization of Multiple Videos. *ACM Trans. Graph.*, 33(4):77:1–77:10.

Yang, H., Lin, W.-Y., and Lu, J. (2014). Daisy Filter Flow: A generalized discrete approach to dense correspondences. In *Computer Vision and Pattern Recognition (CVPR)*, pages 3406–3413. IEEE.

Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., and Oliva, A. (2014). Learning deep features for scene recognition using places database. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K., editors, *Advances in Neural Information Processing Systems 27*, pages 487–495. Curran Associates, Inc.