remote sensing

MDPI

Article

Dense In Situ Underwater 3D Reconstruction by Aggregation of Successive Partial Local Clouds

Loïca Avanthey *,† **and Laurent Beaudoin** †

Epita Research Laboratory, EPITA Computer Engineering School, 94270 Le Kremlin-Bicêtre, France; laurent.beaudoin@epita.fr

* Correspondence: loica.avanthey@epita.fr

† These authors contributed equally to this work.

Abstract: Assessing the completeness of an underwater 3D reconstruction on-site is crucial as it allows for rescheduling acquisitions, which capture missing data during a mission, avoiding additional costs of a subsequent mission. This assessment needs to rely on a dense point cloud since a sparse cloud lacks detail and a triangulated model can hide gaps. The challenge is to generate a dense cloud with field-deployable tools. Traditional dense reconstruction methods can take several dozen hours on low-capacity systems like laptops or embedded units. To speed up this process, we propose building the dense cloud incrementally within an SfM framework while incorporating data redundancy management to eliminate recalculations and filtering already-processed data. The method evaluates overlap area limits and computes depths by propagating the matching around SeaPoints—the keypoints we design for identifying reliable areas regardless of the quality of the processed underwater images. This produces local partial dense clouds, which are aggregated into a common frame via the SfM pipeline to produce the global dense cloud. Compared to the production of complete dense local clouds, this approach reduces the computation time by about 70% while maintaining a comparable final density. The underlying prospect of this work is to enable real-time completeness estimation directly on board, allowing for the dynamic re-planning of the acquisition trajectory.

Keywords: underwater 3D reconstruction; partial reconstruction; completeness; SfM; seabed digital twin

Citation: Avanthey, L.; Beaudoin, L. Dense In Situ Underwater 3D Reconstruction by Aggregation of Successive Partial Local Clouds. *Remote Sens.* **2024**, *16*, 4737. https://doi.org/10.3390/rs16244737

Academic Editors: Fabio Menna, Erica Nocerino, Dimitrios Skarlatos, Caterina Balletti, Panagiotis Agrafiotis, Gottfried Mandlburger

Received: 15 October 2024
Revised: 13 December 2024
Accepted: 16 December 2024
Published: 19 December 2024

1. Introduction

High-precision dense 3D reconstructions of underwater areas covering several hundred square meters, which use images captured in the visible spectrum, are highly valuable, particularly as a core component in the creation of digital twins [1]. This concept, although well-established in industry [2–5], is still emerging in the natural sciences [6], with significant potential for ocean sciences [7]. By providing a faithful digital representation of underwater environments, digital twins can enhance our understanding and allow us to better predict the behavior of these complex systems.

For a range of applications—such as marine resource management, ecosystem preservation, underwater industrial operations, or advanced scientific research—a high-resolution 3D model is critical for capturing the full complexity and dynamics of these environments. This involves not only modeling macro-structures but also capturing finer details, like the distribution of marine flora and fauna on the seabed, which requires precision down to the centimeter scale and the acquisition of distinguishable textures. Therefore, using vision-based methods to generate these 3D models is particularly advantageous.

The image data used to construct these 3D models are acquired either automatically by robots or manually by divers. A key challenge in these acquisition missions is the ability to evaluate "on-site" the completeness of the data collected, which means this must be performed within a short timeframe and within the computational constraints available

in the field. The idea is to perform these assessments either directly on board a robot in near real-time or between dives on a standard laptop. This capability is critical from an operational standpoint as it enables the reprogramming of a coverage trajectory over the same area, if necessary, to complete the dataset during a mission.

In practice, holes in the 3D reconstruction are quite common, particularly in underwater environments where the challenges in achieving precise localization are compounded by reduced observation range (short swath). These gaps often result from trajectory drifts caused by currents or positioning inaccuracies, visual obstructions due to relief, or unusable images (blurred or poorly exposed images due to rapid movements and poorly controlled changing lighting conditions, for example). Factors such as these, including the dynamic and unpredictable nature of underwater lighting, make achieving complete reconstructions especially difficult.

Methods for evaluating the completeness of a 3D point cloud reconstructed from images rely on the dense cloud itself as the sparse cloud lacks the necessary details for this purpose, and a triangulated model may conceal the potential gaps that need to be identified. Once reconstructed, the completeness of the dense point cloud is typically assessed by comparing it to a ground truth such as another model or cloud generated from a different type of sensor. In the absence of ground truth data, the dense cloud can be evaluated on its own, using metrics such as point density, or by leveraging data from the original images that contributed to its generation [8–13].

The generation of the dense point cloud is the primary bottleneck in the process of assessing the completeness on the field as it is a highly time-consuming task. Therefore, this paper explores methods to rapidly generate the dense cloud using limited resources to address this challenge.

The main source of computational cost for dense cloud generation lies in the very high number of points to be evaluated and reprojected in 3D. This phenomenon is further accentuated by the often high acquisition frequencies, varying on average from 120 to 18,000 images over a 10-min period [14,15]. However, a large part of these points are redundant due to the high overlap rate required for 3D reconstruction, typically between 60% and 80% [16,17]. Thus, for a new image, several tens of thousands of points in small resolutions and up to several millions in high-definition resolutions are computed to complete the global cloud [18], while at least 60% of these points have already been evaluated with previous images.

There are three main approaches in the literature for producing dense 3D point clouds: Multi-View Stereo (MVS), incremental dense Structure-from-Motion (dense SfM), and AI-based methods.

MVS approaches evaluate all correspondences across clusters of images, reprojecting matched points from multiple views into a single 3D point. This results in a globally consistent dense reconstruction with minimal redundancy [19–24]. This approach is typically preceded by a sparse SfM framework that accurately estimates camera poses and forms coherent clusters for dense correspondence search. Some software packages or processing pipelines, such as COLMAP [25], visualSFM [26,27], or Metashape [28], integrate both steps, while others require the integration of sparse SfM modules (e.g., Theia [29], OpenMVG [30]) with dense reconstruction modules (e.g., PMVS/CMVS [31], OpenMVS [32]).

Because all viewpoints of the same object are cross-validated to calculate a single point in the final cloud, no redundant 3D points are created. However, the time required to identify multi-view correspondences and produce an optimized global result makes these techniques impractical under operational constraints (on average, dense cloud computation from around 1000 images on a standard laptop with a GPU can take one to two days). Moreover, regardless of the computation time, and although these pipelines can be used between successive robot launches during a mission, they are unsuitable for real-time replanning applications. This is because, in such cases, the images are not available all at once but are instead captured progressively as the area coverage unfolds.

Incremental dense SfM approaches, on the other hand, are built within sparse incremental SfM frameworks [33–42]. In these approaches, only two successive viewpoints of an object are used to produce local dense clouds, which are then aggregated using camera poses calculated by the framework. Thus, unlike MVS, each new viewpoint not only contributes new points but also adds redundant ones to the global cloud, thereby increasing both memory consumption and computational load.

Depending on the frequency and range of bundle adjustment, these approaches produce reconstructions with varying degrees of precision and computational efficiency. Applications prioritizing speed use bundle adjustment sparingly (mainly on loop closures) with limited scope (often defined by a back-propagation window), as seen in SfM frameworks within simultaneous localization and mapping (SLAM) algorithms (e.g., ORB-SLAM3 [43]). For applications where timing is less critical, bundle adjustment is applied more frequently (at each newly processed image, for example), both locally and globally, to maximize accuracy, as in COLMAP. Thus, each new viewpoint adds points (both unique and redundant) to the global dense cloud, increasing memory consumption and computational load.

Finally, in the realm of deep learning, image-based 3D reconstruction techniques primarily employ neural radiance fields (NeRFs) [44,45], 3D Gaussian splatting [46], or large feed-forward networks [47–49]. These methods show considerable promise for generating 3D models with quality comparable to traditional photogrammetric techniques [50,51]. However, they currently require substantial computational power and memory, making them resource-intensive and, hence, incompatible with operational constraints in real-time applications.

The approach that best aligns with the field constraints outlined in this paper is the dense Structure-from-Motion (SfM) approach. However, in comparison to the other two approaches, this approach does not effectively manage the redundancy of data generated by the processing of each new image during the pipeline. In the literature, this redundancy is typically addressed either as a posteriori or upstream.

A posteriori techniques generally involve filtering or meshing. Filtering entails either not reprojecting depth map points into 3D space or decimating the dense 3D points cloud after the reprojection (whether in its final form or during intermediate stages) based on various criteria. These criteria often include removing points that are too close or too distant from others in the 3D cloud or in the 2D reprojection of the cloud onto a new frame or eliminating points that lack spatial or colorimetric homogeneity within their neighborhood. However, applying these criteria can be particularly challenging in natural scenes with poorly defined structures.

Meshing, an alternative to filtering, generates a surface from the dense cloud points, simplifying its geometric complexity. However, this method requires defining a level of detail so as not to hide holes, a task that becomes particularly difficult when dealing with complex elements such as vegetation or wildlife. While filtering and meshing are often used together to manage data volume and reduce noise, they are not well-suited to the operational constraints described in this paper. Instead of reducing computational time, they often increase it as redundant data must first be generated and later removed.

To address redundancy upstream, some methods propose frequency-based or spatial sampling techniques. In these approaches, dense reconstruction is not performed on all consecutive images (which are primarily used for pose estimation) but rather on selected key frames [52,53]. These key frames are chosen to minimize overlap between local clouds, either based on temporal frequency (e.g., selecting one frame every five) or spatial criteria derived from trajectory estimation. However, these approaches often rely on a priori assumptions, which can introduce artificial gaps in the reconstruction, potentially distorting the completeness of the final point cloud.

The approach proposed in this paper aims to accelerate dense point cloud construction on resource-constrained units by incorporating redundancy management upstream within a Structure-from-Motion (SfM) framework. This approach is conceivable, provided that the

framework relies on points of interest, as is common in many existing SfM frameworks, including those previously discussed.

Our method leverages a robust point detector designed to accommodate the variable quality of underwater images, identifying reliable points representative of the textured areas, referred to as SeaPoints, in place of the standard detectors (such as SIFT [54] or ORB [55] or AKAZE [56]) typically used in SfM pipelines. We estimate the overlap rate and the boundaries of the overlap area by analyzing these SeaPoints across four selected frames from the data stream.

This estimation serves two main purposes: first, it enables an automatic spatial sampling tailored to the actual data characteristics, and second, it allows the generation of partial local dense clouds by producing disparity maps and 3D reprojections only in regions providing new information. This selective densification is achieved by propagating matches within a restricted neighborhood around a subset of chosen SeaPoints. The resulting partial local clouds are progressively aggregated during the acquisition process, ultimately forming a comprehensive global dense cloud, similar to the way complete local dense clouds are combined in a conventional dense SfM pipeline.

Section 2 presents the data used and details the proposed method of our approach. The results are presented in Section 3 and are discussed and put into perspective in Section 4, before concluding on the work in Section 5.

2. Materials and Methods

2.1. Overview of the Data Used

The data used to test the proposed algorithm primarily come from four underwater image datasets (see Figure 1), some of which are already publicly available, which we acquired in the Mediterranean Sea:

- The Mermaid Dataset, comprising approximately 1000 images, represents an area of around 150 m^2 at a depth of about 20 m near the Lion-de-Mer islet in Saint-Raphaël, Var, France. The site includes a statue of a mermaid, a sandy seabed scattered with fragments of Roman tiles dating back to the 1st century (remnants of a shipwreck whose wooden structure has long since disappeared), and a rocky outcrop.
- The Lost Freediver Rock Dataset, consisting of approximately 3000 images, covers an area of about 250 m^2 at a depth of 10 m near Cap de Nice in the Alpes-Maritimes, France. The site features rocky formations covered with diverse marine life.
- The Flying Fortress Dataset, composed of around 2000 images, represents the wreckage of a B17 aircraft (which sank in February 1944) located at the base of the citadel of Calvi in Corsica, France, at a depth of about 30 m. Since the acquisition, covering an area of approximately 350 m^2 was completed in a short time frame and with a single pass; the resulting model contains numerous gaps.
- The Landingship Wreck Dataset, consisting of around 3000 images, depicts the wreck of a landing ship tank from Operation Dragoon (USS LST-282, which sank in August 1944) near Cap Dramont in Boulouris, Var, France. The site covers an area of approximately 400 m^2 and is situated at a depth of around 10 m. The wreck mainly consists of an engine, fragments of the frame, and metal plates scattered amidst seagrass beds.

These datasets were selected for this study to encompass a variety of conditions, including different depths, lighting environments, textures, and object types. All images were captured using low-cost cameras (GoPro – GoPro, Inc., San Mateo, CA, USA—and PiCam HQ—Raspberry Pi Ltd, London, UK) under natural lighting conditions. The data acquisitions were performed as still captures rather than frames extracted from a video stream, with no image stabilization activated during the acquisition process. The image resolutions are either 3840 × 2880 px (GoPro) or 4056 × 3040 px (PiCam HQ). The Ground Sampling Distance (GSD) of the images across the four datasets varies between 0.5 mm and 5 mm, with an average GSD of approximately 1.5 mm per dataset. The mean overlap rate between consecutive images is approximately 85% for the Lost Freediver Rock and

Landingship Wreck datasets, 70% for the Mermaid dataset, and 65% for the Flying Fortress dataset. For the GoPro camera, a standard diving housing with a 3 mm flat porthole was employed, whereas for the PiCam system, custom-built housings featuring 10 mm thick flat ports were utilized.

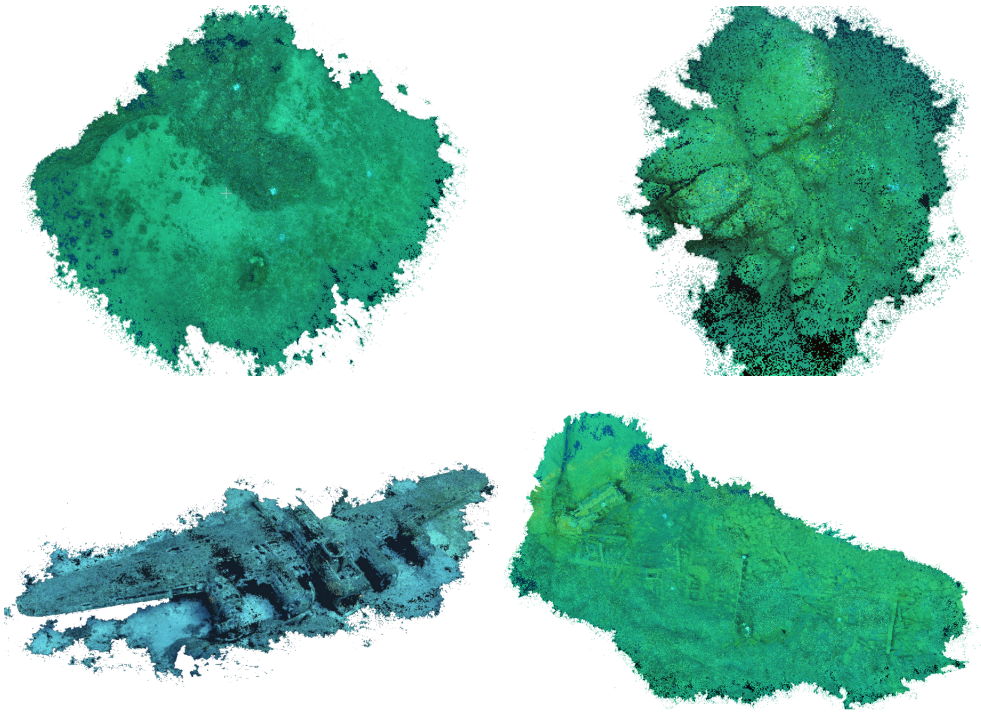

Figure 1. Overview of the global dense point clouds constructed from images of the four datasets: the Mermaid Dataset (**top left**), the Lost Freediver Rock Dataset (**top right**), the Flying Fortress Dataset (**bottom left**), and the Landingship Wreck dataset (**bottom right**).

2.2. Overview of the Proposed Method

The images used for 3D reconstruction are captured by a camera mounted on a platform (robot or diver) that moves systematically across the study area to ensure complete coverage.

The workflow of a standard incremental SfM framework is illustrated in Figure 2 and includes the following main steps: feature detection in each new view, feature matching with the previous view, and robust estimation of the camera pose of the new view relative to the last. Matched points are then reprojected in 3D (sparse cloud). For frames beyond the initialization phase, a loop closure search and/or a bundle adjustment can be applied—either locally (on poses and reprojections within a recent set of images) or globally (on poses and reprojections of all previously processed images). At this stage, in a dense SfM scenario, a dense local point cloud can be generated through depth map estimation and 3D reprojection of matched points. The process then loops, incorporating each new image incrementally.

For onboard applications or scenarios requiring accelerated computations, pre-calibration of the camera is often favored over self-calibration in the pose estimation process. In underwater environments, such calibration data must incorporate a refractive model to account for light propagation through the underwater interfaces (water-glass-air). One approach involves a modified pinhole model that approximates these interfaces as lenses, often referred to as the "UW pinhole" model. However, this model is only accurate at a given distance, limiting its applicability in scenes with substantial depth variation. A common alternative is the use of a dome port centered on the camera optics instead of a planar port, which theoretically eliminates refraction effects and permits the application

of a standard pinhole model [57,58]. For improved accuracy, other approaches explicitly simulate refraction effects, as demonstrated in recent studies [59].

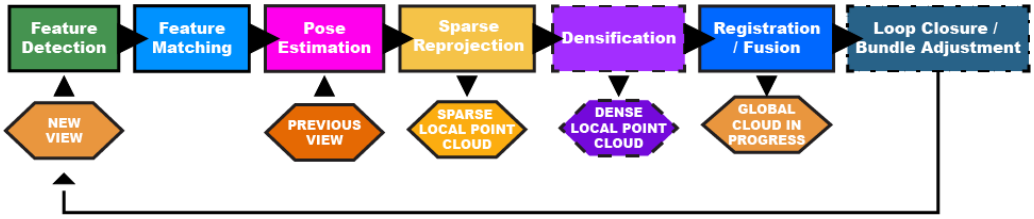

Figure 2. Workflow diagram of a standard incremental SfM framework, with options for incremental dense cloud generation and flexible application of loop closure detection and bundle adjustment based on criteria such as sparsity or exhaustiveness.

The method presented in this article specifically targets two components of the standard SfM framework: feature detection and matching along with the densification process (generation of the local dense cloud). Our algorithm operates independently of the other stages, utilizing their outputs without altering them. Consequently, it can be integrated into various pipelines, provided they are feature-based, allowing users flexibility in their choice of framework. This other stage will thus not be discussed in this paper.

The overall workflow of the proposed algorithm is illustrated in Figure 3. The algorithm starts by forming a new image pair, selecting the next image that optimizes spatial sampling relative to the prior pair, followed by a second image that provides sufficient viewpoint variation to enable depth estimation.

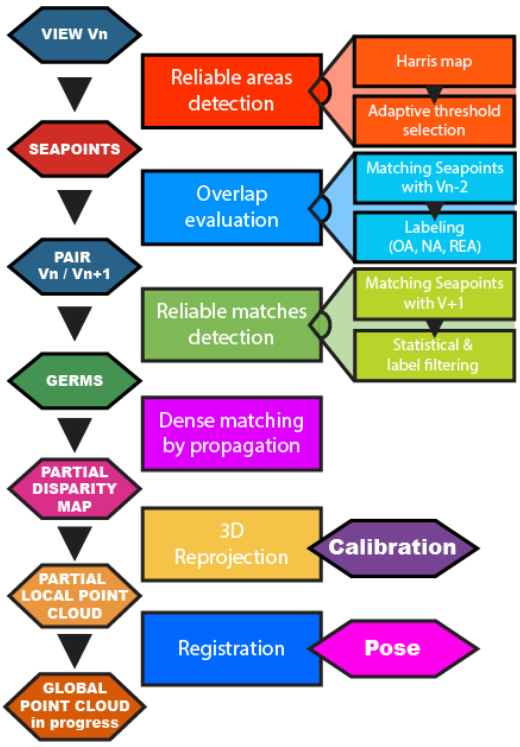

Figure 3. Diagram of the algorithm for generating a partial local dense point cloud from an image pair selected in the incremental flow to optimize spatial sampling and depth resolution. Its main steps include detecting reliable areas using SeaPoints, assessing the overlap rate and identifying the overlap area based on prior information, and performing dense matching by propagating matches in the vicinity of SeaPoints outside the overlap area to obtain a partial disparity map. The resulting dense points can then be reprojected into the 3D frame, as with sparse points within the SfM framework, to form a partial local dense cloud that is subsequently aligned with previous local clouds.

To achieve this, the algorithm matches SeaPoints detected in the newly acquired view V_n with those in the first image of the previous pair, assessing overlap by applying influence areas. SeaPoints are key features designed to reliably identify textured area in underwater environments regardless of the variability of image quality, as discussed in Section 2.3. The matching process for SeaPoints is described in Section 2.4. If an excessive overlap with previously analyzed areas is detected, the algorithm moves directly to the next acquired view for a new overlap assessment. The method for overlap evaluation using influence areas and masking is detailed in Section 2.5. The next image, V_{n+1}, is then tested as a candidate for pairing with V_n, continuing until depth measurability is deemed feasible.

After incrementally selecting a candidate pair for densification, the algorithm proceeds to generate a partial local dense cloud. At this stage, SeaPoints in the selected view V_n are labeled to indicate whether they belong to the overlap area (OA), a new area (NA), or an area requiring reevaluation due to insufficient information (REA). This labeling process, based on data obtained during overlap assessment, is described in Section 2.5.

Matched SeaPoints in the new pair (V_n, V_{n+1}) that are labeled as part of new areas (NA) or reevaluation areas (REA) are used as seed points. Densification around these seeds is achieved through spatial propagation within a constrained neighborhood, enabling the construction of a partial disparity map with automatic exclusion of the overlap area. The partial disparity map reconstruction process is covered in Section 2.6, while the handling of occlusion areas is discussed in Section 2.7.

2.3. Reliable Texture Area Identification with SeaPoint Detector

Many steps of our algorithm are based on the key features detected in the evaluated views. They rely on a large number of points that must effectively identify the most reliable areas and provide an accurate representation of them. Specifically, too few points would fail to provide a meaningful overlap rate, while on the other hand, dense matching relies on spatial propagation and requires a sufficient number of reliable seeds to cover all new areas without encroaching on the overlap regions.

However, in underwater environments, the quality of images can vary significantly within a single acquisition sequence due to factors such as visibility, lighting, and turbidity. As a result, conventional feature detectors often produce a highly variable number of points depending on the view processed, which are not necessarily located in the most reliable areas. Two main categories of strategies have emerged to address the challenges of feature detection in underwater imagery. The first category attempts to recreate conditions typical of aerial imagery, relying on color correction techniques, often powered by neural networks, to significantly improve image quality [60–62]. The main idea is that once the images are corrected, conventional detectors are no longer affected by the specific effects caused by the underwater environment. The second category of approaches seeks to mitigate any particularities of input images by using AI-based detectors and matchers [63–66].

However, both categories have significant limitations in the context of our operational constraints. Firstly, as AI-based methods, they are not easily deployable on field equipment due to their high computational requirements and dependence on specialized hardware. For the first category, the addition of a preprocessing phase (color correction) further increases computational costs, making it all the more difficult to use in resource-constrained environments. Meanwhile, the second category often involves models trained predominantly on aerial datasets, particularly urban scenes, which do not align well with the unique challenges posed by underwater imagery. Moreover, in all these configurations, the detected keypoints do not provide a sufficient distribution across the image to enable fine segmentation into reliable and unreliable regions. This shortfall limits their utility for our specific needs.

To address these challenges, we introduce the SeaPoints. The SeaPoint detector employed in this study is an enhanced version of the algorithm presented in [67]. Designed to highlight the most textured and, therefore, the most reliable regions of an underwater image, the SeaPoint detector offers an optimal balance between efficiency, robustness, and

speed, making it particularly suited to scenarios with limited computational resources. The updated structure of this detector is illustrated in Figure 4.

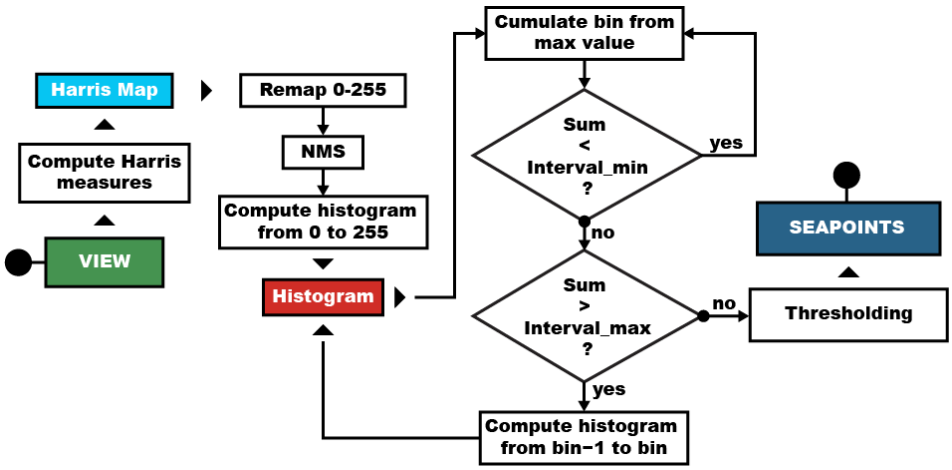

Figure 4. Diagram of the SeaPoint detector algorithm. To begin, we construct a map containing the Harris measurements for each pixel. Next, non-maximum suppression (NMS) is applied on the map to retain only the local maxima within a specified radius. The threshold to select the SeaPoints among these values is then determined through an analysis of the cumulative histogram of the map, aiming to achieve a given range of points. If the currently analyzed histogram bin lacks sufficient granularity (too many values in one bin), the range of values is expanded, generating a new histogram, and the analysis is recursively continued until convergence is achieved. The target interval, which indicates the desired minimum and maximum number of points, must be sufficiently wide to ensure convergence. A target interval with a range of a few hundred points typically guarantees convergence across a wide range of image types. Usually we look for several thousand points on 10 MP images.

The detection process begins with the calculation of Harris measures [68] for each pixel, an approach well-suited to identifying highly textured regions. It then remaps the Harris measure values to a fixed interval from 0.0 to 255.0, using the minimum and maximum values of the calculated map. This normalization guarantees that all values are positive and uniformly scaled across all views. Following this, it applies non-maximum suppression (NMS) within a defined neighborhood radius to filter out points that are either not local maxima or insufficiently dispersed across the view.

To finalize the keypoint selection, the SeaPoint detector applies an adaptive threshold that automatically adjusts to the image quality. This threshold is determined through a recursive search to yield a consistent number of feature points within a target range across all images in the study area. In practice, this target range corresponds to a few thousand points for images with a resolution of around 10 MP.

The adaptive threshold is determined by analyzing the histogram of the measured map. Specifically, we compute the histogram and progressively accumulate values, starting from the highest bin downwards, similar to a cumulative distribution function but proceeding from the maximum value toward zero. This accumulation continues until the cumulative sum reaches the minimum of the target interval. The measure at the currently analyzed bin is then taken as the threshold under consideration, with the cumulative sum representing the number of points that meet this threshold.

When the number of points exceeds the minimum of the target interval, we check to ensure that this count does not exceed the maximum value of the target interval. Since Harris values are not distributed linearly, they tend to appear in clusters of varying sizes, depending on the number of bins we have established for building the histogram. In such cases, we recalculate the histogram for the values located between the current bin's bounds to achieve a finer distribution of this cluster of measures, and we begin summing the bins again to refine the threshold. This recursive process can be repeated as many times as

necessary to effectively separate the measure. An example of a histogram is provided in Figure 5.

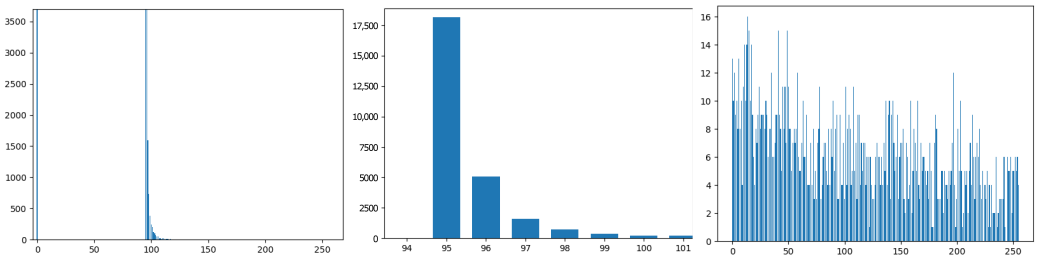

Figure 5. Example of the histogram during the SeaPoint Detector process for the image of the Lost Freediver Rock dataset in Figure 6. (**Left**) is the first histogram and (**center**), a zoom on this histogram. There are not enough points accumulated when arriving at bin 98 (2202 points) of the first histogram to be consistent with the minimum of the target interval (2500 points minimum), but we would exceed the maximum of the target interval (3000 points maximum) by taking bin 97 (3793 points). We, therefore, re-explode the contents of bin 97 into a new histogram by a recursive call (**right**). The algorithms finally converge on 2502 points at bin 186 of the second histogram. Here, the histograms were calculated on 256 bins.

At the end of this recursive search, we obtain a threshold that guarantees a number of feature points contained in the target interval, which we only have to apply on the measure map to obtain the SeaPoints and retrieve their coordinates (see Figure 6).

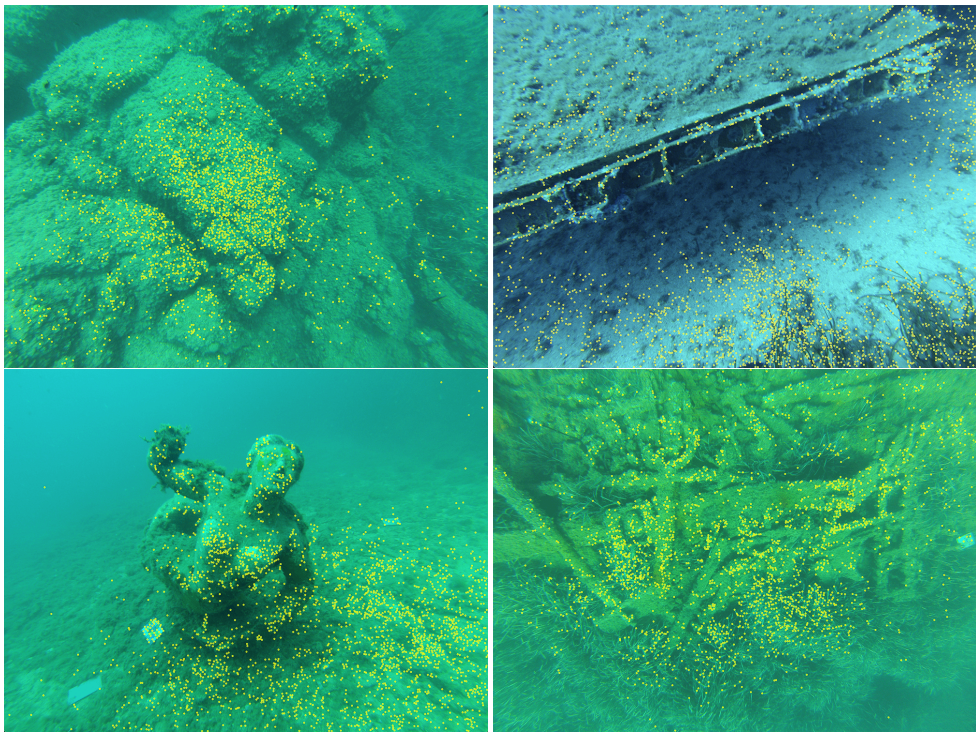

Figure 6. SeaPoint Detector examples with a target interval of [2500, 3000]. (**Top left**): 2502 SeaPoints found on an image of the Lost Freediver Rock dataset in two recursive rounds (threshold adjusted to 38,32% of the max value). (**Top right**): 2930 SeaPoints found on an image of the Flying Fortress dataset in one round (threshold adjusted to 90,20% of the max value). (**Bottom left**): 2500 SeaPoints found on an image of the Mermaid statue in two recursive rounds (threshold adjusted to 62.31% of the max value). (**Bottom right**): 2504 SeaPoints found on an image of the LandingShip Wreck dataset in one round (threshold adjusted to 47.45% of the max value).

2.4. Consolidation of Paired SeaPoints for Overlap Evaluation and Seeds Formation

As explained in Section 2.2, our algorithm starts by selecting an initial pair of views to generate a local dense cloud. It then identifies the next pair to produce the next local cloud that will be registered into a global reference frame alongside all other local clouds. In this context, SeaPoints detected on a given view V serve four main purposes when matched with those from other views: (1) To assess the overlap rate, helping to select the next pair of views; (2) To estimate the relative pose between successive pairs of views for the local clouds registration; (3) To identify and delimit the overlap area between two successive pairs of views, thereby constraining the dense reconstruction only to areas that provide new information; (4) To form seeds within a pair of views and propagate the correspondence in their neighborhood to generate a local partial dense cloud. These steps are respectively detailed in Sections 2.5 (1 and 2) and 2.6 (3 and 4). As illustrated in Figure 7, we refer to the matching within a pair of views as intrapair matching (4) and those between successive pairs of views as interpair matching (1, 2, and 3).

For matching SeaPoints, we employ the rBRIEF descriptor [55], an adapted version of the BRIEF descriptor [69] with added rotation invariance. This approach that has been developed to match ORB points is highly resource-efficient, making it particularly suitable for applications with limited computational resources.

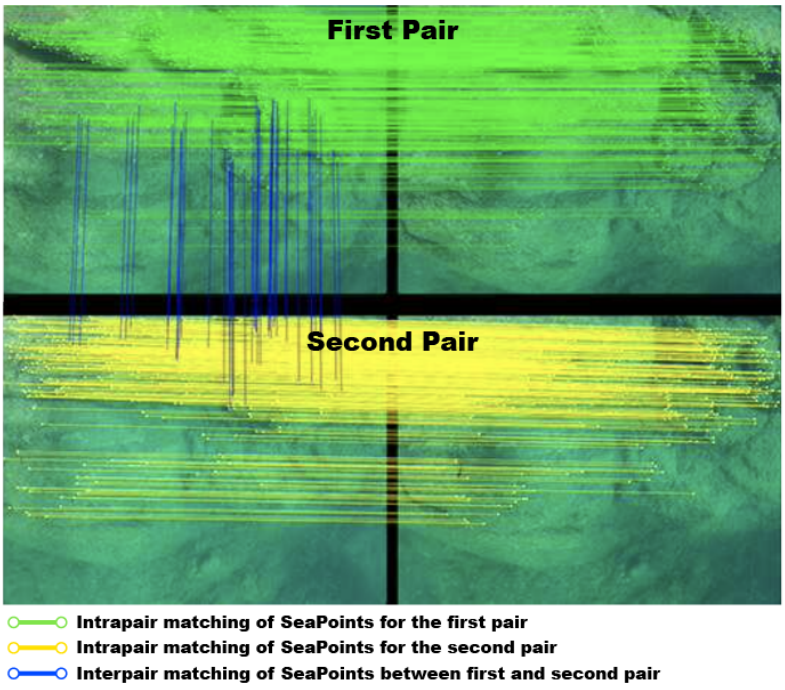

Figure 7. In green and yellow: visualization of the intrapair matchings used to form the seeds for densifying the matching through propagation and generating the partial local dense clouds (4). In blue: visualization of the interpair matching used to evaluate the overlap rate for selecting the next pair (1), to estimate the relative pose for registering the new local cloud (2), and to automatically exclude the overlap area from the 3D reconstruction of the new local cloud (3).

Ensuring the reliability of matching is crucial for both dense reconstruction and registration. Due to the challenging properties of the underwater environment, the rate of erroneous matches is often higher than in aerial contexts. As a result, robust filtering of these matches is an essential step to improve the quality of the reconstruction.

Our filtering algorithm is inspired by the principles of optical flow. Given that our matching densification approach relies on spatial neighborhood criteria and propagation, we opted for a statistical filtering technique applied to the flow of matching vectors to detect and remove incorrect matches from the initial sparse set. The diagram of this filtering algorithm is illustrated in Figure 8.

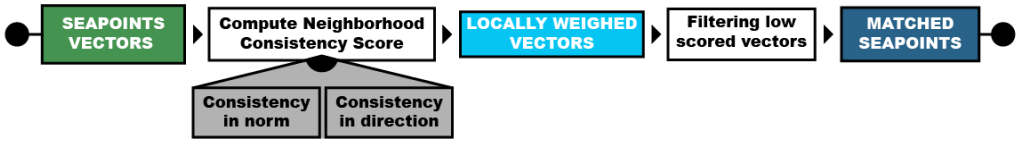

Figure 8. Flowchart illustrating the local statistical filtering applied to SeaPoint directional vectors for distinguishing inliers from outliers: the consistency score of each vector is incremented for each neighboring vector with a similar norm and direction. Vectors with low consistency scores are classified as outliers and are removed, resulting in a refined, robust list of matched SeaPoints.

Let us define a matched SeaPoint directional vector as the vector originating from a matched SeaPoint in view V and extending to its corresponding matched SeaPoint in view V + 1; both are positioned in V + 1. When plotting all the matching directional vectors in V + 1, incorrect matches become clearly visible: they appear as vectors that deviate in direction compared to their neighbors and/or have significantly different lengths.

To filter the outliers, we evaluate the consistency of each candidate directional vector with neighboring vectors by comparing the candidate vector to each nearby vector. If the differences in both direction and length are small, the consistency score of the candidate vector is increased. After all comparisons are completed, the candidate vector is deemed reliable if its consistency score reaches a sufficient threshold; otherwise, it is removed from the list of matched SeaPoints. An illustration of this concept is provided in Figure 9. This filtering step ensures the reliability of the paired Seapoints for their different uses, whether intrapair or interpair.

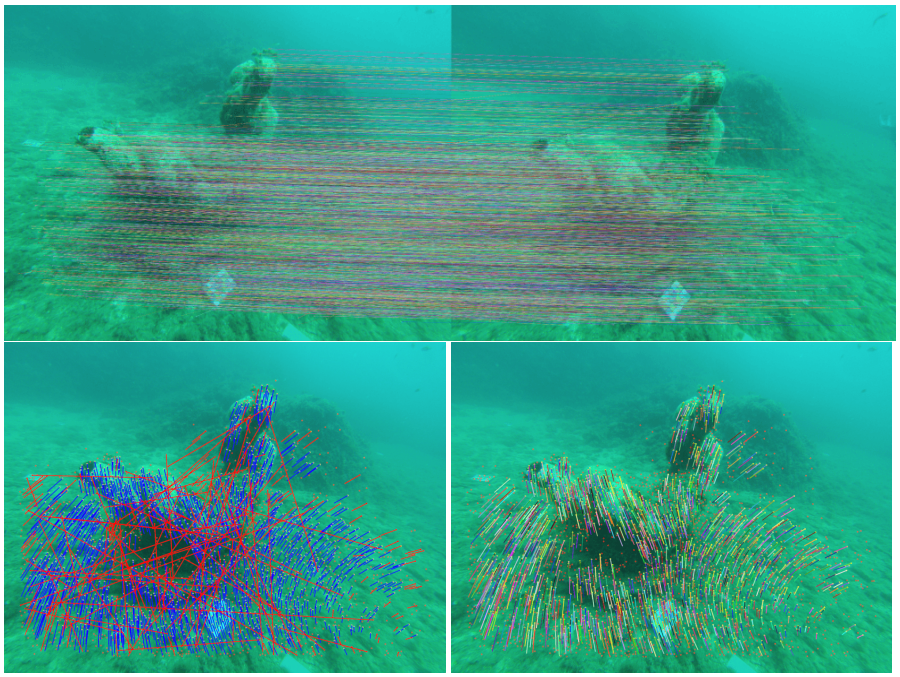

Figure 9. Directional vector flow is a representation of the matching within a single view. A local statistical filtering process, based on neighborhood coherence, is applied: neighboring vectors exhibiting similar norms and directions contribute to the assessment of the studied vector. The greater the number of votes, the more coherent the vector is deemed. The most locally coherent vectors are kept as inliers. In this image, the resulting inliers are represented in blue, while those identified as outliers are marked in red. The latter have a different direction and/or norm from their neighbors (or not enough neighbors to ensure this).

2.5. Interpair Matching to Assess the Overlap Rate, Select the Next Pair of Views, and Estimate the Pose Between Local Successive Clouds

Let us assume that we have an initial pair of views, (V, V + 1), which has allowed us to generate a first dense local cloud. The next step in our algorithm is to select the following pair of views to produce the next local cloud. The main selection criterion is that the new pair should contribute a significant amount of new information compared to the previous one. Therefore, we aim to minimize the overlap rate, but not excessively, as the second objective is to estimate the relative poses of the two local clouds generated from successive pairs. Indeed, for this, we need to rely on reliable matches that are at least moderately distributed. Estimating these successive poses enables us to register the local clouds into a common reference frame, which is a standard procedure that will not be elaborated on in this article.

We start with the assumption that successive views in the acquisition sequence will have a higher overlap than views that are more distant from one another. A view separated by one step (e.g., V and V + 1) will generally have a higher overlap than those separated by two steps (V, V + 2), and so forth. Thus, for our selection process, we first evaluate the view with the highest expected overlap rate and then proceed to analyze subsequent views if the overlap rate is deemed too high. We stop as soon as the overlap rate falls below the desired maximum, ensuring that it does not become too low. With this assumption, the next successive candidate of the pair (V, V + 1) is V + 2.

To assess the overlap rate, we perform an interpair matching between the SeaPoints of V (the first image of the initial pair) and those of V + 2 (the first image of the second potential pair). The overlap rate is then approximated as follows. By definition, the SeaPoints matched between V and V + 2 inherently belong to the overlap area. Since SeaPoints are distributed over the most reliable regions of the views, we treat them as a representative sample of the overlap area. To approximate the boundaries of this area, we extend the sample using the concept of influence areas, assuming that all points near these interpair SeaPoints have a high likelihood of also belonging to the overlap area. An illustration of this process is shown in Figure 10.

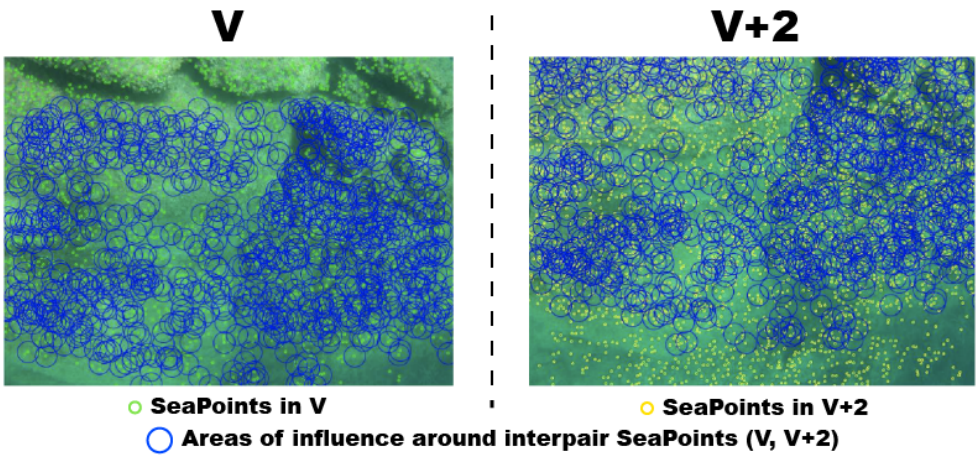

Figure 10. Identification of the overlapping area thanks to the establishment of an area of influence around the interpair SeaPoints. The blue circles indicate the influence areas around the interpair matches on a view V (**left**) and on its subsequent view V + 2 (**right**), delimitating the overlap area between the two views.

A mask is then applied to mark the pixels of V + 2 belonging to the overlap area, and by counting these pixels and comparing their number to the total number of pixels in the view V + 2, we obtain an estimated overlap rate between V and V + 2 (see Figure 11).

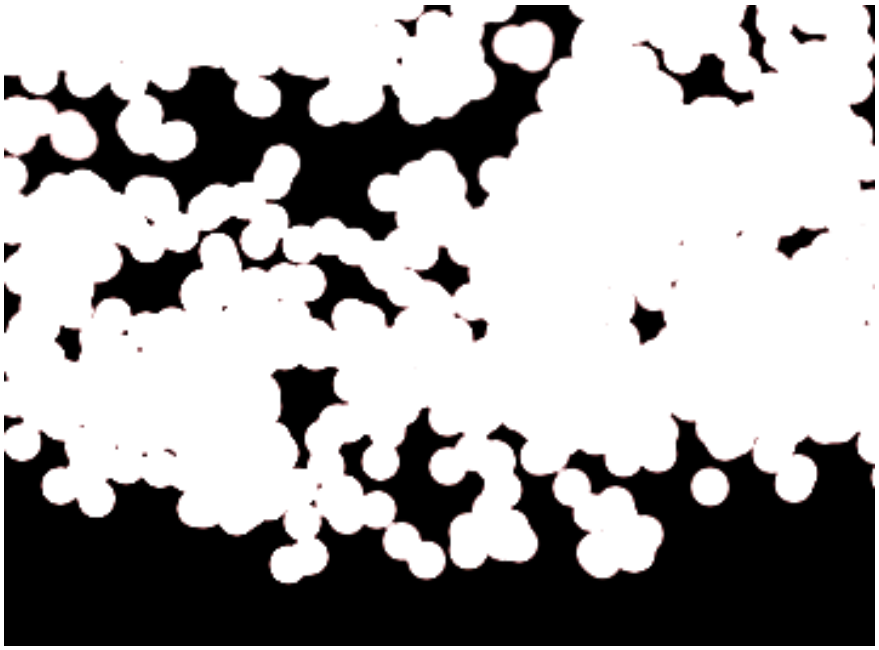

Figure 11. Mask on V + 2 given the areas of influence (see Figure 10) calculated between V and V + 2 using the interpair SeaPoints. The sum of all white pixels in the mask is used to estimate the overlap rate between V and V + 2 with regard to the total number of pixels in V + 2.

If the overlap rate exceeds 60%, it is considered too high, and V + 2 is skipped. The algorithm then proceeds to evaluate the next view in the sequence, V + 3, using the same approach. Conversely, if the overlap rate falls within acceptable limits, V + 2 is selected as the first image of the next pair, which will be used to generate the subsequent local cloud.

As discussed in Section 2.7, the interpair SeaPoints labeled as REA (Re-Evaluation Area) in V will be used on the mask to erase the markings present in their area of influence. As a result, they will not be counted as part of the overlap area, even if they might physically belong to it, due to insufficient information in their vicinity. These areas they are representing, therefore, remain to be explored in the same way as the new zones.

If V + 2 is selected, all its SeaPoints, which are in the active areas of the mask, will be labeled OA (Overlap Area). All the remaining SeaPoints of V + 2 are labeled as NA (New Area) by default. This labeling, based on the analysis with the previous pair, is essential for processing the new pair (see Section 2.6).

2.6. Intrapair Matching for Forming the Seeds Using the Labeled SeaPoints, Propagating the Matching in Their Neighborhood, and Building a Partial Dense Local Cloud

We have established how to select the first image, V + 2, for the new pair; now, we need to choose a second image to complete the pair. To achieve the densest possible reconstruction, a high overlap rate between the two views is essential, typically between 80% and 90%. Following the same principle stated previously, in which successive views in the acquisition flow have a higher overlap rate than those that are further apart, we select the directly successive view, V + 3, to pair with V + 2.

However, it is crucial to have a sufficient viewpoint difference between the two images to accurately estimate disparity and reproject the points in 3D. We can verify this criterion after performing intrapair matching (V + 2, V + 3) of the SeaPoints: if the mean norm of the matching vectors (see Section 2.4) is too low, it indicates that the viewpoint of V + 3 is nearly identical to that of V + 2 (which may occur in cases such as a stationary carrier, whether a robot or diver). In such instances, we opt for the next view, V + 4, to complete the pair.

To reconstruct the local dense cloud, we begin with sparse matching between the two views of the pair (intrapair matching), followed by densification of this matching. Our densification criterion is based on spatial neighborhood principles [70–73]: if a point is close to another, their respective correspondents are likely to be as well, which helps limit the search area. The matched SeaPoints that serve as starting points for propagating the matching are referred to as seeds. The diagram of our propagation algorithm is shown in Figure 12.

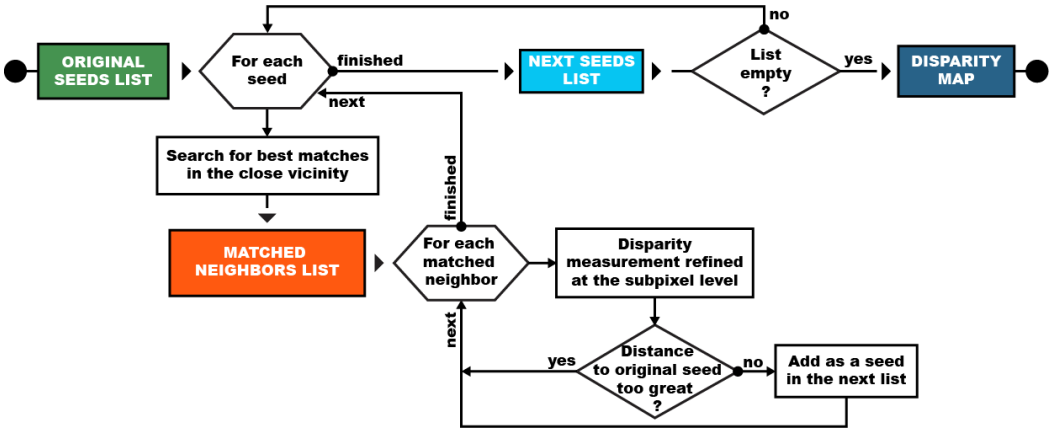

Figure 12. Diagram of the algorithm used to densify the matching by propagation around the seeds. In the first iteration, we analyze the neighborhood of a list of seeds (the initial seeds are all SeaPoints). After having selected all the best possible matches, and if they are not too far from their initial seed (this distance can be approximated by the number of iterations, for example), they are added to a new list of seeds. This new list will be studied in a second iteration until there are no more seeds added to the next list (the points did not match or are all already matched with the best score or are too far from the initial SeaPoint).

SeaPoints from V + 2 labeled as OA (Overlap Area) are excluded from the intrapair matching, ensuring they do not contribute to the initial set of seeds as we do not want to reconstruct information in these areas. We then aim to pair the pixels surrounding the SeaPoints of V + 2, which serve as seeds, with pixels near the corresponding SeaPoints in V + 3. After analyzing the vicinity of an initial seed, we retain the best matchings, which then become new seeds for further pairing of neighboring pixels. The growth of the matching will reach its limits when two regions, originating from different initial seeds, collide and their boundaries stabilize. In the absence of such collisions, the propagation from nearby to distant areas will continue as long as new matchings are identified in the vicinity of the current seeds being studied.

To prevent propagation from extending into the overlap area that we do not wish to reconstruct, we impose a maximum limit on the expansion around the initial seeds. This constraint ensures that propagation is confined to previously unexplored regions. Additionally, this limit helps us remain close to the SeaPoints that represent the most reliable features in the view, allowing us to automatically exclude areas lacking reliable information, such as open water regions.

The disparity map generated at the conclusion of the propagation is a partial map (see Figure 13), containing information solely from the newly explored areas with reliable information. The points are then reprojected into 3D to create the partial local dense point cloud. Finally, Section 2.7 addresses how our algorithm manages the specific case of occlusion areas. The diagram of this overall process to partially reconstruct the local cloud is shown in Figure 14.

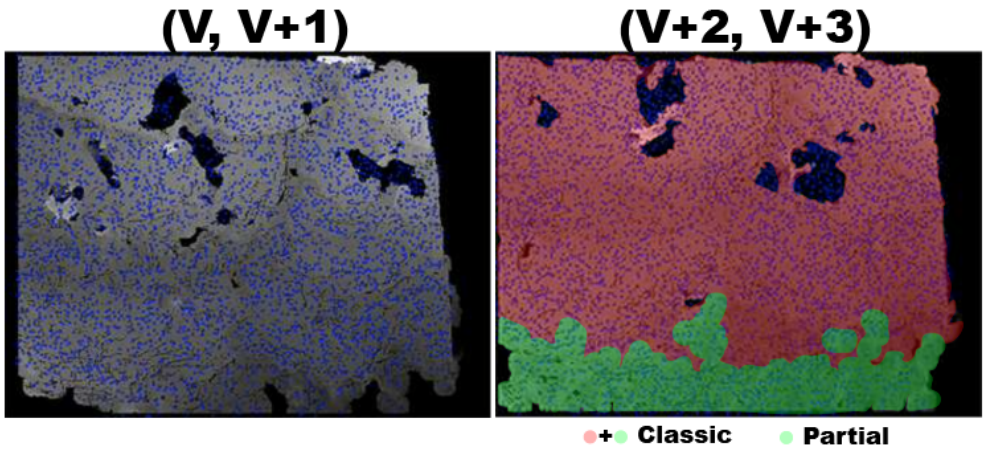

Figure 13. Partial reconstruction: on the (**left**), the disparity map of the first pair with the SeaPoints in blue, and on the (**right**), the disparity map obtained for a normal propagation of the second pair (red + green) as well as the partial disparity map (green only) taking into account the exclusion of the overlap area.

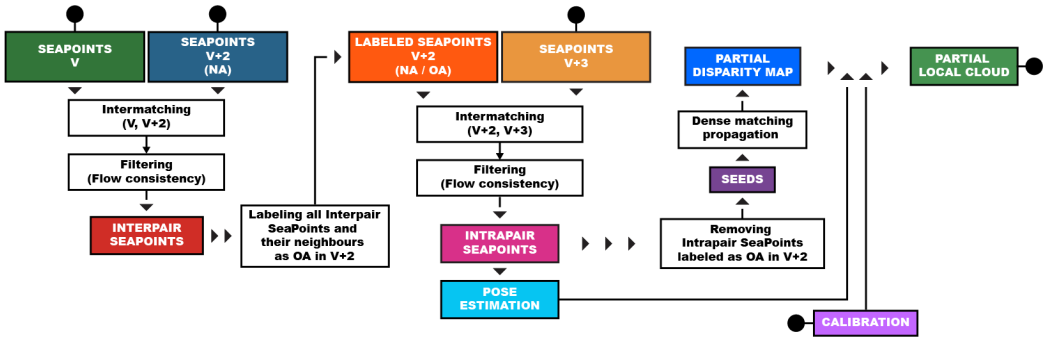

Figure 14. Diagram of the algorithm that reconstructs a partial local cloud by propagating the matching around the seeds while automatically excluding the overlapping area (and areas without reliable information).

2.7. Management of Occlusion Areas

We have described how we approximate the delimitation of overlap areas between two successive pairs, allowing us to focus the reconstruction on parts of the study area that have not yet been explored. However, the change in viewpoint between successive pairs can reveal previously hidden objects within the overlap zone itself. In such cases, we distinguish between two types of occlusions, as illustrated in Figure 15: intrapair occlusion areas and interpair occlusion areas.

Intrapair occlusions occur within a single pair. For example, in the pair (V, V + 1), a seed may have been selected after the intrapair matching but was unable to propagate due to significant differences in the local neighborhood between the two views (e.g., when observing a scene relief from very close proximity). When defining the overlap area for the next pair (V + 2, V + 3), we aim to give these areas a second chance to contribute to the 3D reconstruction.

We can spot these areas by analyzing the disparity map generated by the dense propagation algorithm for (V, V + 1). All SeaPoints of V that have a neighborhood lacking sufficient detail in the disparity map, as well as those in their close vicinity, will be labeled as REA (Re-Evaluation Area). All of the others will be conversely labeled as PA (Processed Area).

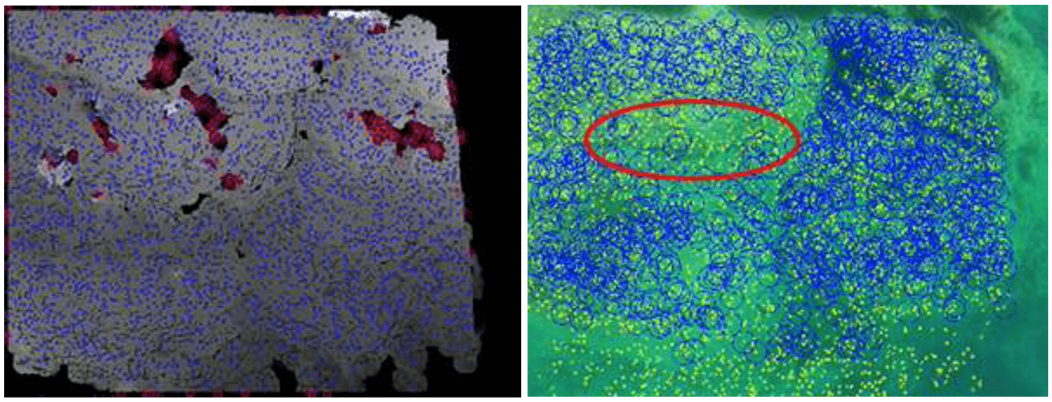

Figure 15. Illustration of two types of occlusion: on the (**left**), intrapair occlusion areas for which local seeds (circled in red in the black areas) have not spread, on the (**right**) an interpair occlusion area (circled in red) for which there has an absence of SeaPoints matched during the interpair matching (no blue circles).

During the interpair matching process seen in Section 2.6 between V and V + 2, all matched SeaPoints in V + 2 and their surrounding SeaPoints are labeled as OA for evaluating the boundaries of the overlap area. Subsequently, we add a new step to identify the interpair SeaPoints in V that have been labeled as REA and remove the OA label from their counterparts and their immediate neighbors in V + 2. This change of label ensures that they are not excluded from the intrapair matching of (V + 2, V + 3) and can participate in the new propagation process.

Interpair occlusion areas arise when a previously hidden part of the scene within the overlap zone becomes visible in the new pair due to the change in viewpoint. If these newly visible areas are not too small, they are automatically handled by our algorithm. Since these regions were not visible in the first pair (V, V + 1), they would not have generated any SeaPoints and thus cannot be used in the interpair matching. Thus, if these areas are large enough not to be classified as part of the overlap area, they can contribute seeds during the new intrapair matching of (V + 2, V + 3), allowing them to participate in the propagation process and fill in the missing information for those parts of the scene. The occurrence of such areas depends greatly on the relief of the scene and the observation distance.

The diagram of this part of the algorithm is shown in Figure 16. A partial propagation result taking into account intrapair and interpair occlusions is available in Figure 17.

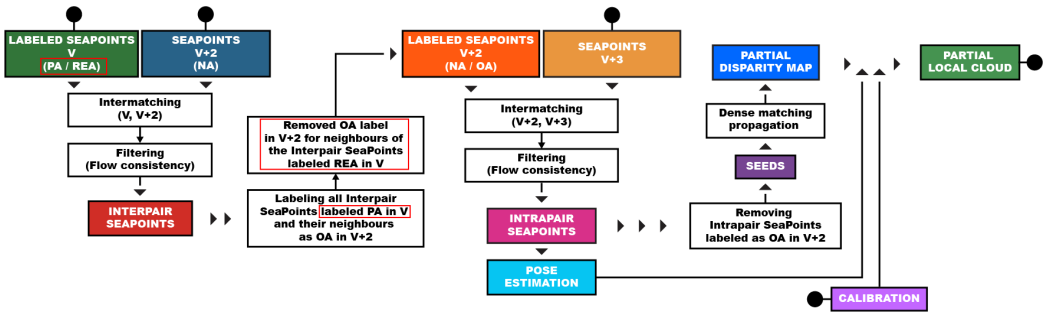

Figure 16. Modified diagram of the algorithm that reconstructs a partial local cloud by propagating the matching around the seeds while automatically excluding the overlapping area to take into account occlusion problems (compared to Figure 14, the changes are framed in red).

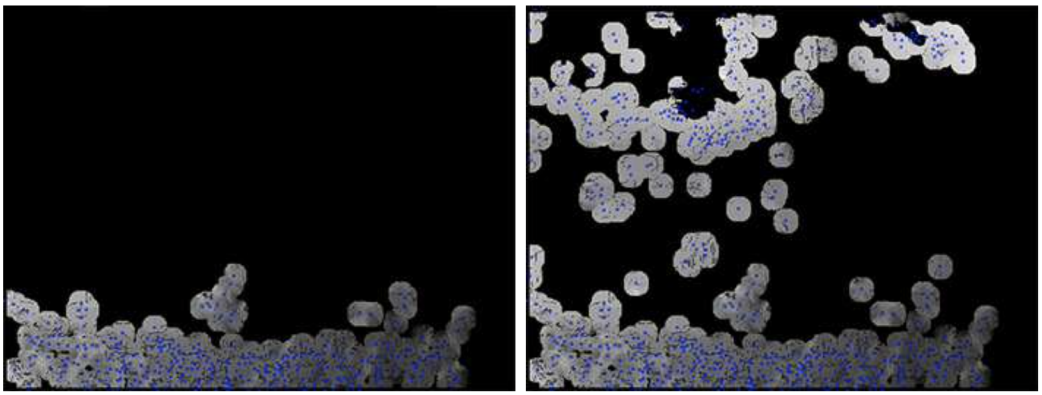

Figure 17. On the (**left**), the disparity map obtained after a partial propagation excluding entirely the overlap area, and on the (**right**), the disparity map obtained after a partial propagation taking into account intrapair and interpair occlusions.

3. Results

In this section, we present the results from two perspectives, corresponding to the two components outlined in Section 2.2: first, the contribution of SeaPoints to the algorithm compared to other types of keypoint commonly used in SfM frameworks, and second, the contribution of the partial dense local cloud computation compared to its complete counterpart. All tests presented in this section were performed on a standard laptop with a 5-core CPU and 16 GB of RAM. The images were processed at their full resolution. For computations on dense clouds by SfM-MVS techniques the use of a basic GPU (Intel Iris Plus Graphics) was enabled.

Regarding SeaPoints, the effectiveness of its adaptive thresholding in consistently producing a large number of reliable keypoints, irrespective of image quality, was demonstrated in [67]. The convergence method proposed in this study, based on a distribution function, does not significantly alter the threshold value (and thus the number of detected keypoints). However, it is more efficient and computationally economical than the divide-and-conquer approaches employed in earlier versions of the detector, such as proportional-derivative or dichotomy methods.

In this work, we focus on evaluating the capability of SeaPoints to effectively represent the reliable part of the overlap area and facilitate matching densification through local propagation while minimizing overflows into undesired regions (e.g., areas lacking reliable information or already reconstructed). To achieve this, SeaPoints must satisfy several key criteria: they must be sufficiently numerous, well-distributed, and well-located to adequately cover the target areas, highly distinguishable to ensure accurate matching for overlap and reconstruction region estimation, and computationally efficient.

The evaluation of these criteria was conducted through interpair matching of detected keypoints, followed by the application of influence areas around each matched point to create associated masks. For comparison, we tested SeaPoints against ORB and SIFT keypoints under similar conditions. The SIFT and ORB detectors were adjusted to produce a comparable number of keypoints as SeaPoints, ensuring a fair evaluation across all three criteria. Figure 18 illustrates an example of the results obtained, and Table 1 provides a summary of the quantitative comparisons.

Notably, these improved performances do not come at the cost of substantial computational overhead. The processing time for obtaining good pairs of SeaPoints (detection, description, matching, and filtering) averages 0.46 s per image (with approximately 3000 SeaPoints per image), comparable to ORB points (0.42 s) and far faster than SIFT (7.44 s). A closer analysis reveals that the primary time savings occur during the detection/description phase, where SeaPoints demonstrate a significant advantage over SIFT in computational efficiency.

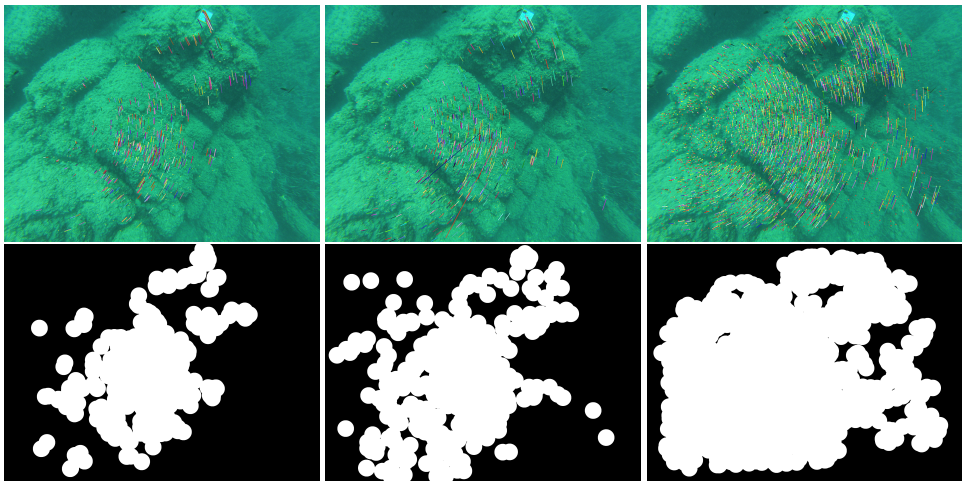

Figure 18. From left to right: example results of intermatching ORB points, SIFT points, and SeaPoints, each using approximately 3000 keypoints in both images of the interpair (**top row**), along with their corresponding masks showing influence areas applied around the matches to segment the reliable regions of the overlap area (**bottom row**).

Table 1. Median results from the tests (an example is shown in Figure 18) comparing SeaPoints, ORB, and SIFT based on matching performance, reliable overlap area assessment, and computation time. The images were processed at full resolution, and the number of detected keypoints per image pair was standardized to 3000 across all three detectors to ensure a fair comparison. The last two rows detail the distribution of total computation time between the detection/description phase and the matching/filtering phase.

(For About 3000 pts/Image)	**SeaPoints**	**SIFT**	**ORB**
Median % of matched points/pair	59.7%	23.2%	41.6%
Median % of reliable shared areas calculated/pair	63.5%	38.9%	27.6%
Total median computation time	0.46 s	7.52 s	0.42 s
Median computation time/pair: Detection/Description	0.39 s	7.44 s	0.35 s
Median computation time/pair: Matching/Filtering	0.068 s	0.081 s	0.070 s

Compared to SIFT and ORB detectors, SeaPoints stand out in their ability to efficiently produce match features and delineate the reliable part of the overlap areas with precision while also facilitating robust propagation into new areas during the densification stage. Approximately 60% of the SeaPoints successfully produce matches, a significant improvement over SIFT (23.2%, with some false positives) and ORB (41.6%).

Thanks to the enforcement of spreading through the NMS step, SeaPoints are inherently well-distributed across the image, avoiding spatial clustering. This distribution is evident in Figure 18 and is further supported by the median percentage of reliable shared overlap areas, which is significantly higher for SeaPoints (63.5%) compared to SIFT (38.9%) and ORB (27.6%) on the same pairs. This superior distribution makes the estimation of overlap areas considerably more reliable.

The second component deals with local point clouds. We proposed a method designed to accelerate the computation of local dense clouds by estimating the overlap area boundaries. This allows partial disparity maps to be generated, performing the calculations only in new areas, enabling the reconstruction of partial dense local clouds.

To assess the effectiveness and impact of this approach, we compared partial reconstructions to full reconstructions of local clouds. Figure 19 illustrates an example of two consecutive local clouds whose image pairs were selected under identical conditions. Both clouds were calculated using the same sparse SeaPoints matching to ensure consistent evaluation. The criteria for comparison included computational efficiency and completeness of the results, with quantitative findings summarized in Table 2. A global metric

evaluation relative to a datum reference (ground truth) was not conducted in this study, as the outcome primarily depends on the specific SfM pipeline employed rather than directly reflecting the performance of the proposed algorithm. Instead, the primary objective of this evaluation was to measure the time savings achieved through partial reconstruction and ensure that it did not come at the expense of introducing new gaps or failing to fill existing gaps, compared to a full reconstruction process.

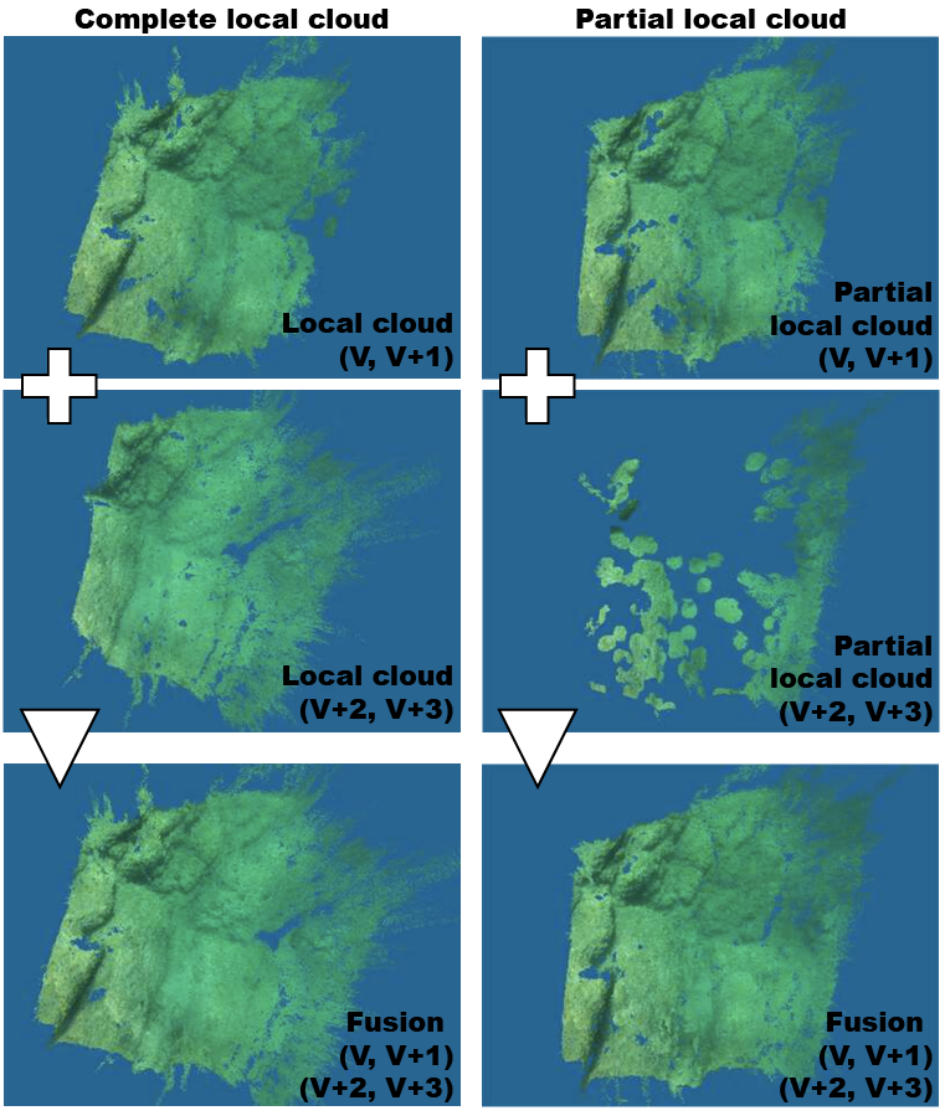

Figure 19. At the (**top**), the two successive local clouds reconstructed classically (total reconstruction), in the (**center**), the two successive local clouds, the second of which is partially reconstructed by following our method. (**Below**), the fusion of the two classic local clouds on the (**left**) and the fusion of the two partial local clouds on the (**right**).

Table 2. Median results from the tests (an example is shown in Figure 19) comparing complete and partial local point cloud aggregation based on density and computation time. The images were processed at full resolution, and the number of detected keypoints per image pair was standardized across all pairs to ensure a fair comparison.

	Complete Local Cloud	Partial Local Cloud
Median density (pt/cm^2)	3.15	1.65
Median computation time	16.27 s	4.89 s

In terms of density, aggregated clouds without redundancy management achieve approximately 3 to 4 points per square centimeter, while those formed using partial local clouds reach 1 to 2 points per square centimeter. These last results correspond to ground sample distances (GSDs) ranging from 7 mm to 5.5 mm, consistent with the GSD of the input data. The completeness of both cloud types is comparable, with minor stochastic variations in the selection of initial points. Notably, the reduced noise from redundant points in partial reconstructions enhances texture perception and the visual quality of the point cloud without having to resort to a posteriori filtering, though it remains, of course, inferior to the results obtained using MVS techniques.

The rapid and precise estimation of interpair overlap areas ensures an approximately 60% overlap between views V_n and V_{n+2}. This adaptive approach avoids redundant computations in areas of limited movement (e.g., stationary or slow-moving sequences), yielding significant time savings depending on the dataset.

On average, newly discovered areas (NA) between pairs (e.g., (V_n, V_{n+1}) and (V_{n+2}, V_{n+3}) constitute about 36%, consistent with the overlap percentage and the presence of occluded regions. Statistical analysis indicates that each interpair SeaPoint has, on average, four neighboring interpair SeaPoints within its influence area. This density significantly contributes to the quality and efficiency of the matching densification, as only isolated seeds propagate to their limits, while most seeds stop propagating upon collision with neighboring regions, remaining within areas of high reliability.

Approximately 5% of seeds fail to propagate due to local occlusions, with 40% of these (i.e., 2% of local seeds in total) being interpair seeds eligible for a second propagation attempt. Similarly, 12% of the seeds correspond to interpair occlusions, which are also given a second chance to propagate. This second opportunity would not have been feasible with simpler overlap area estimations, such as bounding boxes. All these previous percentages are indicative, as variability depends on scene characteristics. For instance, in a flat, textured area (e.g., a pebble plain), occlusions are rare, whereas they are more prevalent in cluttered environments like algae-covered rocks.

The cumulative effect of these optimizations is that, on average, only 40% of the local seeds are retained by the algorithm for each new pair, leading to a computational time saving of approximately 70% compared to full local cloud reconstruction (Table 2). This significant gain is further enhanced by controlled resource usage (memory and processing), drastically reducing memory-swapping overhead and improving overall computational performance. The computation times for the various components of the framework obtained on the different datasets are presented in Table 3.

As an indication, the production of the global dense cloud for our datasets using SfM-MVS softwares requires a significant amount of time, even with GPU acceleration (Intel Iris Plus Graphics GPU). The fastest tools, such as Metashape (v1.8.3), process the Mermaid Dataset in approximately 27.55 h, the Flying Fortress Dataset in 32.47 h, the Lost Freediver Rock Dataset in 64.06 h, and the Landingship Wreck Dataset in 53.42 h. The latter was computed on a more powerful desktop machine (i7-9700K CPU 3.60 GHz, NVIDIA GeForce GTX 1660 SUPER GPU), reducing computation times by an average factor of 3.5 compared to less powerful configurations.

The impact of the approach presented in this paper depends on the processing chain in which it is integrated. Incorporating our algorithm into these pipelines adds an additional 45 min to 2 h of computation time for full-resolution images, depending on the dataset.

For SfM pipelines prioritizing precision, such as Colmap and Metashape, our method offers a reduction in time required for sparse matching (point detection and matching), but this is offset by the added computation of partial dense local clouds, which at worst need to be generated every two frames. With GPU acceleration, the times required to compute poses (sparse clouds) are 2.32 h for the Mermaid Dataset, 2.9 h for the Flying Fortress Dataset, 37.17 h for the Lost Freediver Rock Dataset, and 19.25 h for the Landingship Wreck Dataset (on the high-performance desktop machine for the latest). Without GPU acceleration, the processing of the Mermaid Dataset alone can exceed 20 h. These pipelines

may be viable for simpler datasets with a two-day processing window during a mission (night computation, for example) but would require substantial adjustments to bundle adjustment workflows to support lightweight onboard use.

Table 3. A comparative analysis of computation times for various components of the framework was conducted across four datasets. The components evaluated include pose estimation using incremental Structure-from-Motion (SfM) pipelines, which are categorized as either precision-oriented (e.g., COLMAP, Metashape) or speed-oriented (e.g., OrbSLAM3), and densification processes, which encompass Multi-View Stereo (MVS) techniques as implemented in COLMAP and Metashape, as well as incremental methods for generating complete or partial local point clouds. All computation times were measured with full-resolution image processing on a standard laptop equipped with a 5-core Intel CPU 2 GHz and 16 GB of RAM unless otherwise specified. Times marked with * indicate the use of GPU acceleration (Intel Iris 491 Plus Graphics), those marked with ** were obtained on a more powerful system featuring GPU acceleration (Intel i7-9700K CPU at 3.60 GHz, NVIDIA GeForce GTX 1660 SUPER), and those marked † were derived from images processed at a quarter of their original resolution.

	Mermaid Dataset	**Flying Fortress Dataset**	**Landingship Wreck Dataset**	**Lost Freediver Rock Dataset**
Pose Estimation				
SFM (accuracy-oriented)	2.32 h *	2.90 h *	19.25 h **	37.17 h *
SFM (speed-oriented)	0.12 h †	0.15 h †	0.25 h †	0.15 h †
Densification				
MVS	30.15 h *	24.65 h *	34.17 h **	26.89 h *
Incremental (complete)	2.81 h	3.60 h	6.08 h	3.67 h
Incremental (partial)	0.74 h	1.02 h	1.77 h	1.02 h

For SfM pipelines optimized for speed, such as OrbSlam3, full-resolution images are not supported. By reducing image resolution by a factor of four, these pipelines can process the same datasets in just 10 to 15 min to compute poses and sparse clouds. These systems typically perform bundle adjustments and loop closures in separate threads, ensuring that pose estimation remains fast and responsive for real-time use by the platform, even if it will be imprecise. Our local dense cloud generation aligns well with this background processing framework. These pipelines make it feasible to process data between two consecutive deployments on the field, bringing us closer to achieving onboard processing capabilities for use during missions.

4. Discussion

The algorithm primarily handles consecutive views, but loop closures can be seamlessly integrated by treating them as successive views. These loop closures can either contribute directly to the dense reconstruction or only provide pose information to refine the alignment of previously generated local clouds within the global reference frame. For scenarios requiring scale invariance, the detection of SeaPoints can be extended using image pyramids, allowing keypoints to be identified at multiple scales. These Scale-Invariant SeaPoints (SI-SeaPoints) distribute the target interval across scales, with fewer keypoints assigned to coarser scales, ensuring a balanced representation while maintaining computational efficiency.

Additionally, although rBRIEF has demonstrated robustness to intensity variations [74], such as those typically observed in sequential or closely spaced images, it may struggle under significant intensity changes, as can be seen in loop closures or with artificial lightning. In such cases, substituting it with a more robust descriptor, like the SIFT descriptor, could enhance reliability in matching and reconstruction.

There are remaining challenges associated with embedding completeness evaluation directly on board. SfM frameworks designed for SLAM are likely better suited for address-

ing this need. But as they balance precision with computational efficiency, it is important to assess the impact of reconstruction inaccuracies, which will be a lot greater than those in systems employing frequent bundle adjustments, as well as the lowest working resolution that can be used without introducing gaps larger than the features being detected on the evaluation of the completeness itself. The availability of ground truth data for three of the datasets used in this study (all except the Flying Fortress dataset), derived from a micro-geodesic network, will provide a valuable basis for future research on the reliability of completeness estimation within a defined SfM framework.

Reducing the working resolution would naturally improve computation time. However, if it turns out to be detrimental to completeness evaluation, a hybrid approach might be required. This could involve using a low resolution for pose estimation and a high resolution for dense reconstruction, with corresponding camera calibrations. Keypoints detected in the low-resolution image could then be refined and rescaled for high-resolution processing. Improvements could also be achieved through enhanced parallelization and memory management. Dense global models are highly resource-intensive, so a synthesis and forgetting mechanism would be beneficial to avoid keeping the entire model loaded during field deployment.

Additionally, the quality of dense reconstruction could be improved by prioritizing the spatial position of keypoints over their temporal anteriority when resolving redundancy. Currently, the algorithm retains the oldest points, but it might be advantageous to prioritize points in optimal spatial positions. For example, points near the optical center of the image, where reprojection errors are minimal, could lead to better propagation results. Newly selected seeds from advantageous positions in the current pair could yield higher-quality reconstructions, though at a slightly increased computational cost. To ensure consistency, 3D points near these seeds in the global cloud would need to be decimated to prevent duplicates. Additionally, the merging process could consider the intrinsic quality of the images, prioritizing seeds with higher adaptive thresholds, as these correspond to more textured and higher-quality areas.

A limitation of the algorithm is its current inability to natively handle redundancy for views of the same area captured at different times (non-sequential views). Development leads could involve saving partial information from previously processed pairs to incorporate spatially close pairs, representative subsets, or metadata while remaining compatible with the resource constraints of field deployments.

An advantage of the algorithm is that the rapid overlap rate estimation during processing offers valuable navigation insights. For example, if overlap decreases to critical levels, at the risk that the SfM algorithms get lost, the system could communicate with the platform to adjust the trajectory by, for instance, slowing down or returning to a previously known area.

Also, the method described, which forms pairs between successive images, can be directly generalized to stereo systems and adapted to multi-view systems by extending the pairwise principle to simultaneous views. Although more complex, such systems—when finely synchronized—are particularly effective in dynamic natural environments, such as those involving flora and fauna [75].

5. Conclusions

The background context of this research is the operational in situ dense 3D reconstruction of the seabed. In such an environment, challenges such as the difficulty of obtaining precise positioning and the significant limitations on visibility often result in datasets that exhibit coverage gaps. Consequently, it is crucial to be able to assess the completeness of the collected data directly "on-site," either during the acquisition process or immediately afterward. This allows for the identification of potential coverage gaps, enabling additional data acquisition during the mission time to fill in these deficiencies.

To achieve this objective, it is essential to generate a dense point cloud of the studied area. However, conventional dense reconstruction methods are challenging to utilize

directly on-site due to various operational constraints such as limited memory, processing capacity, and computation time. Therefore, before attempting to assess the completeness of the collected data, a method is required that can accelerate the generation of the global dense point cloud while minimizing the consumption of resources in the field. A significant limitation of the existing methods lies in their inability to effectively manage the high redundancy inherent in stereoscopic imagery, which results in increased data volume, noise, and resource consumption during both the reconstruction and subsequent filtering stages.

This article presents a method for eliminating redundancy before the dense 3D reconstruction process (dense matching and reprojection), thereby removing the need for subsequent filtering of redundant data. The input data consist of successive views. During the creation of local point clouds from pairs of views, instead of performing dense matching across all points, the overlap area is first estimated using SeaPoints—keyfeatures optimized for underwater environments. All matched SeaPoints within this overlap area are excluded from the generation of the next local dense cloud. Consequently, only new 3D points are computed and integrated into the global point cloud. The global reconstruction is achieved by aggregating these partial local clouds within a Structure-from-Motion (SfM) framework. This method significantly reduces resource consumption, yielding an approximate 70% reduction in processing time, compared to a full reconstruction of the local dense clouds, while maintaining consistent final point density. When integrated into an SfM framework optimized for speed, such as those used in SLAM algorithms, dense clouds covering areas between 150 and 400 square meters can be generated in 1 to 2 h on a standard laptop with an Intel i7 CPU. This represents a substantial time gain compared to dense SfM-MVS production, which ranges from 28 to 70 h even with GPU acceleration, or to precision-focused SfM frameworks, which require between 3 and 40 h (also with GPU acceleration).

Several aspects still require further development before enabling embedded evaluation of completeness, including assessing the impact of imprecision and resolution on the evaluation, as well as optimizing computation and memory usage. Since the method operates on pairs of views, it can be generalized and adapted to stereo and multi-view systems, leveraging their rigid structure for enhanced performance.

Author Contributions: Conceptualization, L.A. and L.B.; formal analysis, L.A. and L.B.; methodology, L.A. and L.B.; writing—original draft, L.A. and L.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The images of the Mermaid Underwater Dataset presented in the study are openly available at https://doi.org/10.17882/97987 in the SR202204_LDM-S_D01 directory. The other three Datasets presented in the study have not yet been published but are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Grieves, M. *Virtually Perfect: Driving Innovative and Lean Products Through Product Lifecycle Management*; Space Coast Press: Merritt Island, FL, USA, 2011; p. 133.
2. Glaessgen, E.; Stargel, D. The digital twin paradigm for future NASA and U.S. air force vehicles. In Proceedings of the 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Honolulu, HI, USA, 23–26 April 2012. [CrossRef]
3. Jones, D.; Snider, C.; Nassehi, A.; Yon, J.; Hicks, B. Characterising the Digital Twin: A systematic literature review. *CIRP J. Manuf. Sci. Technol.* **2020**, *29*, 36–52. [CrossRef]
4. Fuller, A.; Fan, Z.; Day, C.; Barlow, C. Digital Twin: Enabling Technologies, Challenges and Open Research. *IEEE Access* **2020**, *8*, 108952–108971. [CrossRef]
5. Abisset-Chavanne, E.; Coupaye, T.; Golra, F.R.; Lamy, D.; Piel, A.; Scart, O.; Vicat-Blanc, P. A Digital Twin use cases classification and definition framework based on Industrial feedback. *Comput. Ind.* **2024**, *161*, 104113. [CrossRef]
6. Viazilov, E. About Creating a Digital Twins in Field of Earth Sciences. *Int. J. Appl. Sci. Dev.* **2022**, *1*, 42–51. [CrossRef]

7. Simões-Marques, M.; Água, P.; Frias, A.; Correia, A. Metaverse and Digital Twins: Contributions, opportunities and challenges to a sustainable use of the ocean. In *Human Factors and Systems Interaction*; AHFE International: New York, NY, USA, 2023. [CrossRef]
8. Merrell, P.; Mordohai, P.; Frahm, J.M.; Pollefeys, M. Evaluation of Large Scale Scene Reconstruction. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Rio De Janeiro, Brazil, 14–21 October 2007; pp. 1–8. [CrossRef]
9. Zhao, X.; Wu, R.; Zhou, Z.; Wu, W. A new metric for measuring image-based 3D reconstruction. In Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), Tsukuba, Japan, 11–15 November 2012; pp. 1030–1033.
10. Chen, Z.; Chen, W.C.; Sung, P.Y. A novel 3D dense reconstruction with high accuracy and completeness. In Proceedings of the 2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), San Jose, CA, USA, 15–19 July 2013; pp. 1–4. [CrossRef]
11. Stromer, D.; Kugler, P.; Bauer, S.; Lauritsch, G.; Maier, A. Data Completeness Estimation for 3D C-Arm Scans with Rotated Detector to Enlarge the Lateral Field-of-View. In *Bildverarbeitung für die Medizin 2016*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 164–169.
12. Waechter, M.; Beljan, M.; Fuhrmann, S.; Moehrle, N.; Kopf, J.; Goesele, M. Virtual Rephotography: Novel View Prediction Error for 3D Reconstruction. *ACM Trans. Graph.* **2017**, *36*, 8. [CrossRef]
13. Zhang, G.; Chen, Y. A Metric for Evaluating 3D Reconstruction and Mapping Performance with No Ground Truthing. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Anchorage, AK, USA, 19–22 September 2021; pp. 3178–3182. [CrossRef]
14. Alcantarilla, P.F.; Beall, C.; Dellaert, F. Large-Scale Dense 3D Reconstruction from Stereo Imagery. In Proceedings of the IROS Workshop on Planning, Perception and Navigation for Intelligent Vehicles (PPNIV), Tokyo, Japan, 3 November 2013.
15. Rothermel, M.; Haala, N.; Fritsch, D. Generating Oriented Pointsets from Redundant Depth Maps using Restricted Quadtrees. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. (ISPRS)* **2014**, *XL-3*, 281–287. [CrossRef]
16. McGlone, C.J.; Mikhail, E.M.; Bethel, J.S.; Mullen, R. *Manual of Photogrammetry*, 5th ed.; American Society for Photogrammetry and Remote Sensing (ASPRS): Baton Rouge, LA, USA, 1980 .
17. Linder, W. *Digital Photogrammetry*; Springer: Berlin/Heidelberg, Germany, 2009.
18. Jenkin, M.; Verzijlenberg, B.; Hogue, A. Progress towards underwater 3D scene recovery. In Proceedings of the Third C* Conference on Computer Science and Software Engineering (C3S2E'10), Montreal, QC, Canada, 19–20 May 2010; pp. 123–128.
19. Drap, P. Underwater Photogrammetry for Archaeology. In *Special Applications of Photogrammetry*; IntechOpen: London, UK, 2012; pp. 111–136.
20. Skarlatos, D.; Demestiha, S.; Kiparissi, S. An 'Open' Method for 3D Modelling and Mapping in Underwater Archaeological Sites. *Int. J. Herit. Digit. Era* **2012**, *1*, 1–24. [CrossRef]
21. Diamanti, E.; Vlachaki, F. 3D Recording of Underwater Antiquities in the South Euboean Gulf. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2015**, *1*, 93–98. [CrossRef]
22. Rende, S.F.; Irving, A.D.; Lagudi, A.; Bruno, F.; Scalise, S.; Cappa, P.; Montefalcone, M.; Bacci, T.; Penna, M.; Trabucco, B.; et al. Pilot Application of 3D Underwater Imaging Techniques for Mapping *Posidonia oceanica* (L.) Delile Meadows. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2015**, *XL-5/W5*, 177–181. [CrossRef]
23. Nocerino, E.; Menna, F.; Gruen, A.; Troyer, M.; Capra, A.; Castagnetti, C.; Rossi, P.; Brooks, A.J.; Schmitt, R.J.; Holbrook, S.J. Coral Reef Monitoring by Scuba Divers Using Underwater Photogrammetry and Geodetic Surveying. *Remote Sens.* **2020**, *12*, 3036. [CrossRef]
24. Rende, S.F.; Bosman, A.; Menna, F.; Lagudi, A.; Bruno, F.; Severino, U.; Montefalcone, M.; Irving, A.D.; Raimondi, V.; Calvo, S.; et al. Assessing Seagrass Restoration Actions through a Micro-Bathymetry Survey Approach (Italy, Mediterranean Sea). *Water* **2022**, *14*, 1285. [CrossRef]
25. Schönberger, J.L.; Frahm, J.M. Structure-from-Motion Revisited. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 4104–4113. [CrossRef]
26. Wu, C.; Agarwal, S.; Curless, B.; Seitz, S.M. Multicore bundle adjustment. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011, pp. 3057–3064. [CrossRef]
27. Wu, C. Towards Linear-Time Incremental Structure from Motion. In Proceedings of the 2013 International Conference on 3D Vision, Seattle, WA, USA, 29 June–1 July 2013; IEEE Computer Society: Piscataway, NJ, USA, 2013; pp. 127–134. [CrossRef]
28. Agisoft. Agisoft Metashape User Manual Standard Edition, Version 2.1. 2024. Available online: https://www.agisoft.com/ (accessed on 13 November 2024).
29. Sweeney, C. Theia Multiview Geometry Library: Tutorial & Reference. 2018. Available online: http://theia-sfm.org (accessed on 13 November 2024).
30. Moulon, P.; Monasse, P.; Perrot, R.; Marlet, R. OpenMVG: Open multiple view geometry. In Proceedings of the International Workshop on Reproducible Research in Pattern Recognition, Cancún, Mexico, 4 December 2016; Springer: Cham, Switzerland, 2016; pp. 60–74.
31. Furukawa, Y.; Ponce, J. Accurate, Dense, and Robust Multiview Stereopsis. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 1362–1376. [CrossRef] [PubMed]
32. Cernea, D. OpenMVS: Multi-View Stereo Reconstruction Library. 2020. Available online: https://github.com/cdcseacave/openMVS (accessed on 13 November 2024).

33. Pizarro, O.; Eustice, R.; Singh, H. Large Area 3D Reconstructions from Underwater Surveys. In Proceedings of the IEEE OCEANS Conference, Kobe, Japan, 9–12 November 2004; Volume 2, pp. 678–687.
34. Jasiobedzki, P.; Se, S.; Bondy, M.; Jakola, R. Underwater 3D Mapping and Pose Estimation for ROV Operations. In Proceedings of the IEEE OCEANS Conference, Québec, QC, Canada, 15–18 September 2008; pp. 1–6.
35. Nicosevici, T.; Garcia, R. Online Robust 3D Mapping Using Structure from Motion Cues. In Proceedings of the IEEE OCEANS Conference, Québec, QC, Canada, 15–18 September 2008; pp. 1–7.
36. Johnson-Roberson, M.; Pizarro, O.; Williams, S.B.; Mahon, I. Generation and Visualization of Large-Scale Three-Dimensional Reconstructions from Underwater Robotic Surveys. *J. Field Robot.* **2010**, *27*, 21–51. [CrossRef]
37. Williams, S.B.; Pizarro, O.R.; Jakuba, M.V.; Mahon, I.; Ling, S.D.; Johnson, C.R. Repeated AUV Surveying of Urchin Barrens in North Eastern Tasmania. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Anchorage, AK, USA, 3–7 May 2010; pp. 293–299.
38. Beall, C.; Dellaert, F.; Mahon, I.; Williams, S.B. Bundle Adjustment in Large-Scale 3D Reconstructions Based on Underwater Robotic Surveys. In Proceedings of the IEEE OCEANS Conference, Kona, HI, USA, 19–22 September 2011; pp. 1–6.
39. Mahon, I.; Pizarro, O.; Johnson-Roberson, M.; Friedman, A.; Williams, S.B.; Henderson, J.C. Reconstructing Pavlopetri: Mapping the World's Oldest Submerged Town Using Stereo-vision. In Proceedings of the IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 2315–2321.
40. Shkurti, F.; Rekleitis, I.; Dudek, G. Feature Tracking Evaluation for Pose Estimation in Underwater Environments. In Proceedings of the Canadian Conference on Computer and Robot Vision, St. Johns, NL, Canada, 25–27 May 2011; pp. 160–167.
41. Jeon, I.; Lee, I. 3D Reconstruction of unstable underwater environment with SFM using SLAM. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2020**, *XLIII-B2-2020*, 957–962. [CrossRef]
42. Jiang, S.; Jiang, C.; Jiang, W. Efficient structure from motion for large-scale UAV images: A review and a comparison of SfM tools. *ISPRS J. Photogramm. Remote Sens.* **2020**, *167*, 230–251. [CrossRef]
43. Campos, C.; Elvira, R.; Rodríguez, J.J.G.; Montiel, J.M.; Tardós, J.D. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Trans. Robot.* **2021**, *37*, 1874–1890. [CrossRef]
44. Mildenhall, B.; Srinivasan, P.P.; Tancik, M.; Barron, J.T.; Ramamoorthi, R.; Ng, R. Nerf: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM* **2021**, *65*, 99–106. [CrossRef]
45. Balloni, E.; Gorgoglione, L.; Paolanti, M.; Mancini, A.; Pierdicca, R. Few shot photogrametry: A comparison between NeRF and MVS-SfM for the documentation of cultural heritage. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2023**, *48*, 155–162. [CrossRef]
46. Kerbl, B.; Meuleman, A.; Kopanas, G.; Wimmer, M.; Lanvin, A.; Drettakis, G. A hierarchical 3D gaussian representation for real-time rendering of very large datasets. *ACM Trans. Graph. (TOG)* **2024**, *43*, 1–15. [CrossRef]
47. Wang, F.; Zhu, Q.; Chang, D.; Gao, Q.; Han, J.; Zhang, T.; Hartley, R.; Pollefeys, M. Learning-based Multi-View Stereo: A Survey. *arXiv* **2024**, arXiv:2408.15235.
48. Han, J.; Kokkinos, F.; Torr, P. Vfusion3d: Learning scalable 3D generative models from video diffusion models. In Proceedings of the European Conference on Computer Vision, Milan, Italy, 29 September–4 October 2024; Springer: Cham, Switzerland, 2024; pp. 333–350.
49. Tochilkin, D.; Pankratz, D.; Liu, Z.; Huang, Z.; Letts, A.; Li, Y.; Liang, D.; Laforte, C.; Jampani, V.; Cao, Y.P. Triposr: Fast 3D object reconstruction from a single image. *arXiv* **2024**, arXiv:2403.02151.
50. Remondino, F.; Karami, A.; Yan, Z.; Mazzacca, G.; Rigon, S.; Qin, R. A Critical Analysis of NeRF-Based 3D Reconstruction. *Remote Sens.* **2023**, *15*, 3585. [CrossRef]
51. Croce, V.; Billi, D.; Caroti, G.; Piemonte, A.; De Luca, L.; Véron, P. Comparative Assessment of Neural Radiance Fields and Photogrammetry in Digital Heritage: Impact of Varying Image Conditions on 3D Reconstruction. *Remote Sens.* **2024**, *16*, 301. [CrossRef]
52. Geiger, A.; Ziegler, J.; Stiller, C. Stereoscan: Dense 3D reconstruction in real-time. In Proceedings of the 2011 IEEE Intelligent Vehicles Symposium (IV), Baden-Baden, Germany, 5–9 June 2011; IEEE: Piscataway, NJ, USA, 2011, pp. 963–968.
53. Fillinger, L.; Funke, T. A New 3-D Modelling Method to Extract Subtransect Dimensions from Underwater Videos. *Ocean Sci.* **2013**, *9*, 461–476. [CrossRef]
54. Lowe, D. Object recognition from local scale-invariant features. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Corfu, Greece, 20–27 September 1999; Volume 2, pp. 1150–1157. [CrossRef]
55. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G. ORB: An efficient alternative to SIFT or SURF. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2564–2571. [CrossRef]
56. Fernández Alcantarilla, P. Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces. *IEEE Trans. Patt. Anal. Mach. Intell.* **2011**, *34*, 1281–1298. [CrossRef]
57. Menna, F.; Nocerino, E.; Fassi, F.; Remondino, F. Geometric and Optic Characterization of a Hemispherical Dome Port for Underwater Photogrammetry. *Sensors* **2016**, *16*, 48. [CrossRef]
58. Song, Y.; Nakath, D.; She, M.; Köser, K. Optical Imaging and Image Restoration Techniques for Deep Ocean Mapping: A Comprehensive Survey. *PFG–J. Photogramm. Remote Sens. Geoinf. Sci.* **2022**, *90*, 243–267. [CrossRef]
59. She, M.; Seegräber, F.; Nakath, D.; Köser, K. Refractive COLMAP: Refractive Structure-from-Motion Revisited. *arXiv* **2024**, arXiv:2403.08640.

60. Wang, H.; Zhang, W.; Bai, L.; Ren, P. Metalantis: A Comprehensive Underwater Image Enhancement Framework. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 1–19. [CrossRef]
61. Wang, H.; Sun, S.; Chang, L.; Li, H.; Zhang, W.; Frery, A.C.; Ren, P. INSPIRATION: A reinforcement learning-based human visual perception-driven image enhancement paradigm for underwater scenes. *Eng. Appl. Artif. Intell.* **2024**, *133*, 108411. [CrossRef]
62. Vlachos, M.; Skarlatos, D. Self-Adaptive Colour Calibration of Deep Underwater Images Using FNN and SfM-MVS-Generated Depth Maps. *Remote Sens.* **2024**, *16*, 1279. [CrossRef]
63. DeTone, D.; Malisiewicz, T.; Rabinovich, A. Superpoint: Self-supervised interest point detection and description. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–23 June 2018; pp. 224–236.
64. Sarlin, P.E.; DeTone, D.; Malisiewicz, T.; Rabinovich, A. Superglue: Learning feature matching with graph neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 4938–4947.
65. Lindenberger, P.; Sarlin, P.E.; Larsson, V.; Pollefeys, M. Pixel-perfect structure-from-motion with featuremetric refinement. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021, pp. 5987–5997.
66. Lindenberger, P.; Sarlin, P.E.; Pollefeys, M. Lightglue: Local feature matching at light speed. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 4–6 October 2023; pp. 17627–17638.
67. Avanthey, L.; Beaudoin, L.; Gademer, A.; Roux, M. Tools to Perform Local Dense 3D Reconstruction of Shallow Water Seabed. *Sensors* **2016**, *16*, 712–742. [CrossRef] [PubMed]
68. Harris, C.; Stephens, M. A Combined Corner and Edge Detector. In Proceedings of the Alvey Vision Conference, Manchester, UK, 31 August–1 September 1988; Volume 15, pp. 147–152.
69. Calonder, M.; Lepetit, V.; Strecha, C.; Fua, P. BRIEF: Binary Robust Independent Elementary Features. In Proceedings of the Computer Vision—ECCV 2010, Heraklion, Crete, Greece, 5–11 September 2010; Springer: Berlin/Heidelberg, Germany, 2010; pp. 778–792.
70. Lhuillier, M.; Quan, L. Robust Dense Matching using Local and Global Geometric Constraints. In Proceedings of the IEEE International Conference on Pattern Recognition (ICPR), Barcelone, Spain, 3–7 September 2000; Volume 1, pp. 968–972.
71. Tang, L.; Wu, C.; Chen, Z. Image Dense Matching Based on Region Growth with Adaptive Window. *Pattern Recognit. Lett.* **2002**, *23*, 1169–1178. [CrossRef]
72. Strecha, C.; Tuytelaars, T.; Van Gool, L. Dense Matching of Multiple Wide-Baseline views. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Nice, France, 14–17 October 2003; pp. 1194–1201.
73. Kannala, J.; Brandt, S.S. Quasi-Dense Wide Baseline Matching Using Match Propagation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Minneapolis, MN, USA, 18–23 June 2007; pp. 1–8.
74. Karami, E.; Prasad, S.; Shehata, M. Image Matching Using SIFT, SURF, BRIEF and ORB: Performance Comparison for Distorted Images. In Proceedings of the Newfoundland Electrical and Computer Engineering Conference, Halifax, NS, Canada, 3–6 May 2015.
75. Avanthey, L.; Beaudoin, L.; Villard, C.; Mellouk, S.; Treglia, R. Synchronization of PiCam Cameras for Three-Dimensional Study of Dynamic Multi-Domains Natural Scenes. In Proceedings of the International Society for Photogrammetry and Remote Sensing Annals [ISPRS], XXIV ISPRS Congress, Nice, France, 31 August–2 September 2020; Volume V-1-2020, pp. 277–284.