Article

Method for the Coordination of Referencing of Autonomous Underwater Vehicles to Man-Made Objects Using Stereo Images

Valery Bobkov [1], **Alexey Kudryashov** [1,*] **and Alexander Inzartsev** [2]

1 Institute of Automation and Control Processes, Far Eastern Branch of the Russian Academy of Sciences, 690041 Vladivostok, Russia; bobkov@iacp.dvo.ru
2 Institute of Marine Technology Problems, Far East Branch of the Russian Academy of Sciences, 690091 Vladivostok, Russia; inzar@marine.febras.ru
* Correspondence: alkud1981@mail.ru

Abstract: The use of an autonomous underwater vehicle (AUV) to inspect underwater industrial infrastructure requires the precise, coordinated movement of the AUV relative to subsea objects. One significant underwater infrastructure system is the subsea production system (SPS), which includes wells for oil and gas production, located on the seabed. The present paper suggests a method for the accurate navigation of AUVs in a distributed SPS to coordinate space using video information. This method is based on the object recognition and computation of the AUV coordinate references to SPS objects. Stable high accuracy during the continuous movement of the AUV in SPS space is realized through the regular updating of the coordinate references to SPS objects. Stereo images, a predefined geometric SPS model, and measurements of the absolute coordinates of a limited number of feature points of objects are used as initial data. The matrix of AUV coordinate references to the SPS object coordinate system is computed using 3D object points matched with the model. The effectiveness of the proposed method is estimated based on the results of computational experiments with virtual scenes generated in the simulator for AUV, and with real data obtained by the Karmin2 stereo camera (Nerian Vision, Stuttgart, Germany) in laboratory conditions.

1. Introduction

Advances in submerged industrial infrastructure, including subsea production systems (SPS), gas and petroleum pipeline systems, etc., require that regular checks of their state are made. Until recently, inspections were carried out by divers and/or tethered remotely operated underwater vehicles (ROV). However, in many cases, it is advisable to use autonomous unmanned underwater vehicles/robots (AUVs) instead of ROVs. The use of AUVs rather than ROVs is less time-consuming and less expensive when carrying out a number of operations, particularly in the case of siting SPS objects in the polar regions with complete ice cover. In [1–4], a review of the subsea infrastructure inspection problem is given, and the importance of developing new AUV-based technologies along with ROV-based ones is shown.

One of the tasks when using AUVs to inspect SPS objects is the thorough photography of its specified fragments (in particular, dashboards), for which navigation references of the AUV to the SPS with sub-meter accuracy must be provided (it is assumed, of course, that the water's transparency allows photographing). Ensuring this accuracy is a challenge. Utilizing standard on-board autonomous navigation devices along with hydroacoustic AUV navigation equipment for these purposes makes it impossible to provide the required sub-meter accuracy. These devices can be used only to arrange the passage of the AUV

to the object of inspection. A possible solution is to perform a sonar or laser scan for accurate navigation referencing to the object, and then to work with the internal INS (inertial navigation system)-based navigation system and Doppler log [5,6]. However, the accumulated navigation error and the "noisy" data from the Doppler log near the SPS object, as a result, also do not allow the required accuracy.

The high-precision navigation of an AUV relative to an SPS can be ensured by way of processing the video images obtained aboard the vehicle on a real-time basis. A key challenge here is to recognize the underwater object. Experiments on the use of video information for accurate navigation referencing are currently reduced to the problem of positioning (hovering) the AUV near a fragment with the given pattern [7]. In a more general statement, the problem of the navigation referencing of an AUV to the inspected object based on video information is not considered. Another difficulty is to overcome the errors accumulated by the prolonged movement of the AUV when using visual odometry, or any other incremental navigation method. In recent studies, the emphasis has been on various aspects of the problem of object recognition and the problem of inspecting underwater structures in general.

In [8], two algorithms for visual odometry based on stereo vision are proposed for cases of the close movement of the AUV to the seabed. In [9], the authors introduce the navigation problem in detail and the methods used for the inspection. In [10], a method for localizing AUVs using visual measurements of underwater structures and artificial landmarks is described. In a number of works, for example, in [11,12], methods of tracking the desired trajectory using visual measurements of points features and adaptive control, including neural networks, are considered. In [13], an approach based on the combined use of an extended Kalman filter and a vision system for the underwater docking of an autonomous underwater vehicle is proposed. In [14], algorithms for navigation, obstacle avoidance and AUV control are proposed to solve the problem of underwater port inspection using AUVs. In [15], a method of monocular visual odometry with optical flow tracking is proposed, which, according to the authors, is more suitable for underwater imaging than the classical approaches based on descriptors. In [16], the authors present a study testing various visual odometry solutions in relation to AUVs. In particular, the SIFT (scale-invariant feature transform) and SURF (speed up robust feature) detectors were compared for calculating vehicle movement. Testing was performed using a set of real data. The article argues that the proposed strategy could support and improve navigation using the DVL (Doppler velocity log) or could provide an alternative (without using DVL). In [17], the problem of landing on the seabed is solved using physical models that take the geometry of the vehicle, the slope of the seabed, roughness, friction and currents into account. In [18], the authors present a survey and comparison of global descriptors for 3D object recognition purposes when a 3D model of the object is available a priori. The area of interest is underwater IMR (inspection, maintenance and repair) applications. The recognition approach uses both images collected with a stereo camera and 3D depth data from a range scanner. In [19], the problem of determining the distance to an underwater object and its orientation relative to the AUV is solved. To solve this, two new architectures based on convolutional neural networks are proposed. In [20], a study is presented that was conducted within the framework of the Seventh EU Framework Program "CADDY—Cognitive Autonomous Diving Buddy" (University of Zagreb. Faculty of Electrical Engineering and Computing, Zagreb, Croatia). The approach aims to take advantage of the complementary traits of a human diver and an AUV by making their synergy a potential solution to the mitigation of state-of-the-art diving challenges. The proposed algorithms use measurements from a stereo camera, sonar, and ultra-short baseline acoustic localization to ensure the vehicle constantly follows and observes the diver. In [21], a large overview of modern technologies for solving the problems of the communication, localization and navigation of AUVs in underwater environments, which take into consideration the impossibility of relying on radio communications and global positioning systems, is presented.

In most of the known works, the problem of developing accurate visual navigation methods that are resistant to the accumulation of errors during long movements of the AUV is solved without recognizing underwater objects and subsequent coordinate references to those objects.

Increasing the efficiency of navigation in these cases is achieved through various modifications of the ICP (iterative closest point) algorithm, the use of the long-term tracking of features in images, situations of closed contours, combinations with other sensory measurements, etc. In these works, the accuracy of the proposed methods is also assessed in comparison with other methods.

In the examples of work of this kind considered below, estimates are given for the accuracy of navigation, both for virtual scenes and in conditions of underwater sea scenes. These estimates give an idea of the level of navigation accuracy of the proposed methods, including in comparison with the classical visual odometry scheme.

In [22], the authors compared three pose estimation methods for unmanned ground vehicles in GPS (global positioning system)-denied environments (RANSAC (RANdom SAmple Consensus) EKF (extended Kalman filter), GICP (generalized ICP) and iSAM (incremental smoothing and mapping)) using visual data on a real-world dataset (for an urban environment). Regarding the absolute final error (m) for a trajectory with a length of 1.443 km, the error in navigation accuracy varied from 16 to 29 m.

In [23], a practical approach to performing underwater visual localization was proposed, which improves upon the traditional EKF-SLAM (simultaneous localization and mapping). According to the authors, thanks to the realized ability to reliably solve the "closed loops" problem, as shown in the experiments, the presented approach provides accurate pose estimates, using both a simulated robot and a real robot, in controlled and real underwater scenarios. In experiments with a virtual scene, the error of the visual method was 4.4 cm. In experiments with the real robot in a pool (7 m long, 4 m wide and 1.5 m depth), the accumulated localization error of the robot when moving along a closed trajectory reached ≈40 cm. In an experiment in real undersea conditions, two implementations were compared: classical visual odometry and SLAM with loop closings established during the mission execution. It was noticed that visual odometry showed a significant location error of up to 7 m due to drift. On the contrary, according to the authors, the SLAM estimates were much closer to the real trajectory thanks to several loop closings established during the mission's execution.

In [24], which is devoted to the 3D reconstruction of objects, estimates of errors were obtained when restoring the shape of an underwater object in experiments with a real underwater scene, with errors of 2–2.8%.

In [25], the authors present the results of a comparative analysis of the effectiveness of eight known software packages that are based on the use of visual odometry and are designed to calculate the trajectories of the AUV and 3D reconstruction of underwater objects from images. Estimates of the errors in calculating the AUV trajectories are given in this article for underwater scenes with a trajectory length of ≈400 m. The best result was achieved by the ORB (Oriented FAST and Rotated BRIEF)-SLAM package [26], with an error of 11.2 m. The COLMAP package (Swiss Federal Institute of Technology Zurich, Zurich, Switzerland and University of North Carolina, Chapel Hill, North Carolina, USA, license https://colmap.github.io/license.html) (accessed on 15 September 2021) [27] showed a better result of 9.2 m, although the time to obtain the estimated trajectory can be very long, e.g., for 700 images, 7–8 h. For other software packages, these errors are much larger (as indicated in the article), ranging from 20 to 112 m. [27] proposes a structure-from-motion algorithm that improve the state-of-the-art in terms of completeness, robustness, accuracy, and efficiency. In [28], a good overview of the SLAM issue is presented.

Some new works related to underwater vehicles (ROV and AUV) and new applications of underwater robots are presented in [29–33]. In [29], the problem of the negative impact of ocean currents and various unmodeled disturbances on the UV control system is considered. The authors carried out a study based on nonlinear dynamics to implement the reliable

positioning control of an over-actuated autonomous underwater vehicle under the influence of ocean currents and model uncertainties.

In [30], the authors analyze the movement of a paired unmanned surface vehicle (USV)–umbilical cable (UC)–unmanned underwater vehicle system to investigate the interaction behavior between the vehicles and the UC in the ocean environment. For this, a new dynamic modeling method is used to study the multi-body dynamic system of this communication system.

In [31,32], the physical aspects of the functioning of an underwater construction robot for underwater use are considered.

In [33], the authors present a new algorithm for docking a torpedo-shaped autonomous underwater vehicle (AUV). The algorithm comprises three phases: depth tracking, docking feasibility region analysis and docking success probability evaluation. This article proposes an approach to ensure accurate AUV navigation in the SPS coordinate space by performing regular referencing to the object coordinate system based on processing stereo images. The emphasis is on the object recognition algorithm using a predetermined point model of the object, in which there are a limited number of characteristic points with known absolute coordinates.

Problem Statement

SPS inspection using AUV implies that the vehicle passes along the trajectory that is most suitable for accomplishing the tasks on a working mission. These tasks include but are not limited to taking photos of individual elements and units, maintaining communication lines and surveying for cathodic protection. To generate the trajectory, accurate coordinate referencing of the AUV to the inspected SPS objects is needed. To do this, one should formulate a method to accurately coordinate the referencing of the AUV to the SPS coordinate system. In this work, we base the method of referencing on video information that is received from the stereo camera mounted aboard the vehicle.

The SPS includes the individual parts that are isolated from one another, such as the drilling stations and manifold. The SPS integrates these parts into a network via pipelines and flexible drill strings. The SPS structure is schematically shown in Figure 1.
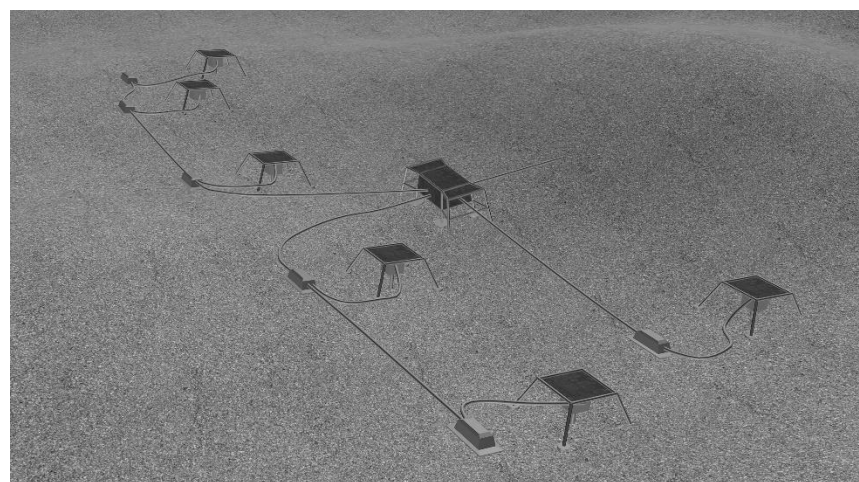

Figure 1. Subsea production system (SPS).

The video information-based navigation method used on the AUV [34] facilitates the calculation of the trajectory of movement and the construction of a set of 3D points observed by the camera (3D cloud) in each position of the trajectory. The bottom relief points and the points belonging to SPS objects are some of the observed points.

It is assumed that the geometric model of the SPS has a two-level structure with a set of constituent objects, and each object is represented by a set of 3D feature points (FPs)

coordinated in the coordinate system (CS) of the object. Firstly, such a structure makes it possible to plan the expected trajectory across the entire SPS space. Simultaneously, this model would ensure more accurate coordinate referencing to the object CS when performing necessary measurements or manipulations. Secondly, referencing to the object allows for the elimination of some navigation errors. These errors are linked to the time-related variations in the FPs, such as silting and fouling, and minor changes in the mutual arrangement of SPS objects relative to one another, which is a result of soil movement or subsidence. That is, when performing actions in relation to any object of the SPS, it is better to navigate in relation to the FPs belonging to this object.

It is also anticipated that each object has at least three FPs. The absolute coordinates are measured within the external CS when mounting the object on the bottom. For FP images on pictures, corner points are usually utilized. Corner points can be accurately determined with a greater degree of certainty by detectors and trackers.

To confirm the specified referencing of an AUV to an SPS as a whole and to each object individually, a method needs to be developed with the goal of finding the FPs of SPS objects in the 3D cloud. These objects are captured by a stereo camera. Next, a method for calculating the respective matrices of geometric transformation from the AUV/camera CS to SPS CS, and finally to the CS of each object, is developed.

The rest of the paper is organized as follows. Section 2 describes the suggested method for the coordinate referencing of an AUV to an SPS. In particular, Section 2.1. presents a geometric model of the SPS utilized for the identification of underwater SPS objects. Additionally, Section 2.2. presents a detailed algorithm for the identification of underwater objects using FPs. Next, Section 2.3. describes a procedure for the direct computation of the desired matrix of referencing using the feature points of an object, which are matched to a model. In Section 2.4, the visual navigation method (visual odometry) is specified and the obtained coordinate references for the navigation of the AUV are used in the SPS space. Section 3 discusses the results of computational experiments with model scenes, and also presents the evaluation of how the method works via an example of processing real data, which is collected using a stereo camera under laboratory conditions.

The main contributions of this work include a method for identifying an underwater object, the core of which is the algorithm for searching the points belonging to the SPS object in the 3D cloud (Sections 2.2 and 3 for a discussion of the results). Furthermore, a method is proposed for calculating the matrix of binding the AUV to the CS of the object (Section 2.3.) and the calculation of the movement of the AUV in the coordinate space of the SPS object (Section 2.4.).

2. Method for Coordinate Referencing

The following designations will be applied hereafter (Table 1).

Table 1. Coordinate systems and transformations matrices.

WCS	–	**World Coordinate System**
CS^{AUV_i}	–	Coordinate system associated with AUV in position i.
CS^{AUV_0}	–	Coordinate system associated with AUV in the initial position.
CS^{SPS}	–	SPS coordinate system.
CS^{ob_id}	–	Coordinate system of object id, belonging to SPS.
$H_{CS^{AUV\_0},CS^{AUV\_i}}$	–	Transformation matrix from the coordinate system in the initial AUV position to the coordinate system in position i. This matrix is formed by multiplying out local matrices of relative displacement, each of which connects the css of the two adjacent positions.
$H_{CS^{AUV\_i},CS^{ob\_id}}$	–	Transformation matrix from the coordinate system of AUV in position i to the coordinate system of object No.id.

Table 1. *Cont.*

WCS	–	World Coordinate System
$H_{CS^{ob\_id},WCS}$	–	Transformation matrix from the coordinate system of object id to the world coordinate system.
$H_{CS^{AUV\_0},WCS}$	–	Transformation matrix from the coordinate system of AUV in position 0 to the world coordinate system.

2.1. SPS Model

The SPS consists of several objects that are remote from one another. As a pre-formed model that uniquely identifies the SPS object, the set of its feature points (FP) and the set of measured distances between them are considered. Accordingly, a set of 3D points visible via the camera (3D cloud) is used (during the AUV movement) to search FPs corresponding to the object model. The 3D cloud is formed by matching 2D features of the images of a stereo pair (a Harris corner detector is used to extract corners and infer features of an image, and a SURF detector is used to match the selected features in a pair of images by descriptors) and by the triangulation of rays on the matched features. Let the set of the FPs of object *ob_id* be denoted by M^{ob_id}. In set M^{ob_id}, the three FPs are singled out, for which the absolute coordinates are measured when the object is mounted on the bottom. The coordinates in the external CS, which are designated as the world CS (*WCS*), are called the absolute coordinates. The CS of this object is constructed using this triplet according to the rule demonstrated in Figure 2.
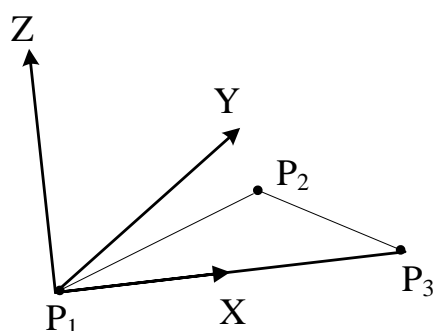

Figure 2. Construction of an object coordinate system based on points P_1, P_2, and P_3 specified in the *WCS*: the X axis is determined by points P_1P_3, the Z axis is normal to the plane of the P_1P_2 and P_1P_3 vectors, and the Y axis is normal to the ZX plane.

The transformation matrix $H_{CS^{ob\_id},WCS}$ connecting CS^{ob_id} and *WCS* is formed based on the unit vectors CS^{ob_id} specified in *WCS*:

$$H_{CS^{ob\_id},WCS} = \begin{vmatrix} e1_x & e1_y & e1_z & 0 \\ e2_x & e2_y & e2_z & 0 \\ e3_x & e3_y & e3_z & 0 \\ r_x & r_y & r_z & 1 \end{vmatrix} \tag{1}$$

where $e1$, $e2$, $e3$ are unit vectors CS^{ob_id}, and r is the vector of the CS^{ob_id} origin, specified in *WCS*.

The first point out of the above three FPs is the origin of the object CS. All the object FPs are specified in the object CS (relative coordinates). Thus, for each object, there is a matrix used to transform the coordinates of its points from the object CS to the *WCS*. For each object, a min–max-shell for the object CS and the *WCS* is computed, which is required to simplify the problem of creating inspection AUV trajectories and controlling the AUV's autonomous movement close to the object. The points of interest in terms of inspection are specified in the object CS.

The SPS CS must also be defined, for which the CS of one of the objects can be used. All the objects (points of the CS origin for each object) are coordinated in the SPS CS. The min–max-shell of the SPS in the *WCS* is constructed in a similar way.

There are two stages of the coordinate referencing of an AUV to SPS objects. At the stage of rough referencing, the displacement of the AUV to the SPS localization area is controlled. To do this, the AUV standard acoustic navigation equipment can be used. Subsequent to the displacement of the AUV to the SPS localization area, the problem of the accurate coordinate referencing of the AUV to the object CS using video information is resolved.

Two approaches to solving this problem can be taken. The first one is based on the estimation of the AUV's movement in the *WCS* space using a SLAM algorithm (see, for example, review [27,28]). Then, the SPS object is coordinated in the *WCS* via the previously obtained transformation $H_{CS^{ob\_id},WCS}$, and the AUV is coordinated in the *WCS* using transformation $H_{CS^{AUV\_0},CS^{AUV\_i}}$, obtained in our case by our own SLAM algorithm [34] and the previously obtained transformation $H_{CS^{AUV\_0},WCS}$. Then, the desired transformation can be computed in the following manner (Figure 3):

$$H_{CS^{AUV\_i},CS^{ob\_id}} = (H_{CS^{AUV\_0},CS^{AUV\_i}})^{-1} \cdot H_{CS^{AUV\_0},WCS} \cdot (H_{CS^{ob\_id},WCS})^{-1} \qquad (2)$$
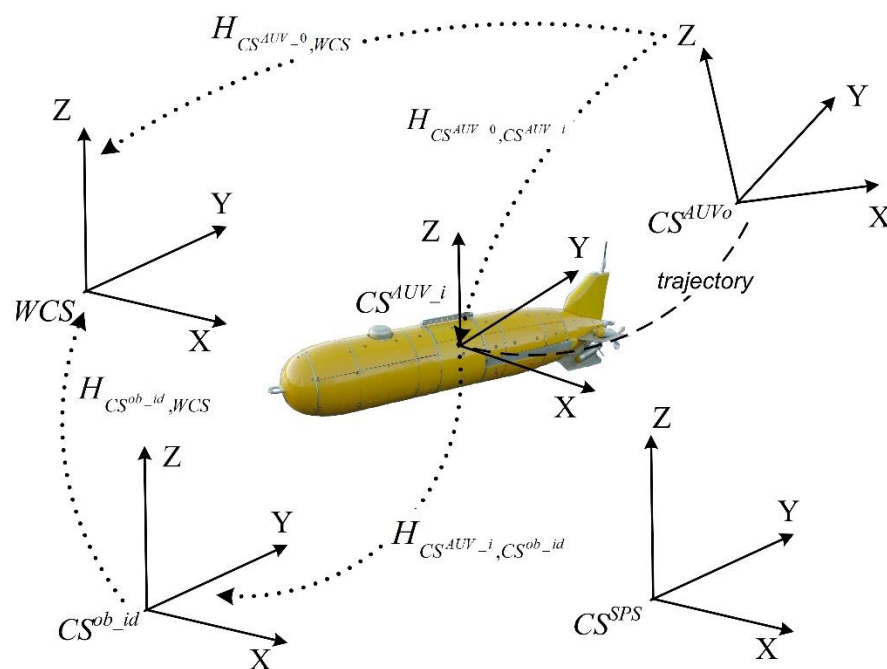

Figure 3. Coordinate systems (*WCS*, CS SPS, object CS, CS associated with an AUV in the initial and current positions) used and geometric transformations between them.

However, this method of AUV navigation in the space of the object CS can be not quite accurate, owing to the error introduced by the visual navigation method. It is known that this method involves an accumulation of errors when there are long-term AUV displacements. In this instance, the error will be accumulated while calculating matrix $H_{CS^{AUV\_0},CS^{AUV\_i}}$.

Hence, it is suggested that the second approach to solving the problem of referencing the coordinates of AUV to the SPS be used, which employs the estimation of the AUV's movement relative to the SPS object. This approach eliminates the above-mentioned drawback, and is intended to ensure high-precision navigation in the SPS space. Nevertheless, it requires that the problem of identifying the feature points of the SPS object should be solved using a priori knowledge of the object model. Therefore, the data of the SPS model (models of all objects and coordinate transformation matrices connecting the CS of objects

with the CS SPS) are loaded into the on-board AUV program for the subsequent operation of the object recognition algorithm during the AUV inspection mission.

2.2. Identification of the SPS Feature Points

A set of spatial points $C \{C_1 \ldots C_M\}$ (3D cloud), seen by the camera, is formed in the current position of the AUV trajectory. The absolute coordinates of the points from the 3D cloud are calculated using the visual navigation method according to the known procedure:

- FPs matching on images of two stereo pairs;
- Calculation of 3D coordinates of the corresponding points in CS^{AUV_i};
- Computation of the local matrix to transform the coordinates at the current stage;
- Computation of matrix $H_{CS^{AUV\_0}, CS^{AUV\_i}}$ via combining local matrices from the previous stages;
- Calculation of absolute coordinates, sequentially applying two transformations—$(H_{CS^{AUV\_0}, CS^{AUV\_i}})^{-1} \cdot H_{CS^{AUV\_0}, WCS}$.

Next, the object is recognized using the algorithm described below for searching in a 3D cloud of points belonging to the SPS object. The algorithm for searching is carried out in two stages. As the main recognition criterion, the principle of structural coherence is implemented: the same mutual arrangement of points is desired in two comparable groups of points. In the context of the problem being solved, this means that a subset of points is searched for in a 3D cloud, the mutual arrangement of which corresponds to a certain subset of points of the SPS object model. Since searching in a 3D cloud for a subset of points of the corresponding object model involves enumerating a large number of options, the problem of reducing this enumeration arises. Therefore, the first stage of the algorithm consists of selecting candidate points for belonging to an object using rough filtering, which drastically reduces the search for options. Filtering is based on the construction for each point of the object model of a spatial rectangular shell in the 3D cloud, inside which candidate points are searched for that match the point of the model. The spatial shell is built using knowledge of the absolute coordinates of both the points of the SPS object and the points of the 3D cloud. In the second stage, for the obtained limited set of candidate points, a search is performed for a subset of points of an object with unambiguous identification based on a criterion that implements the principle of structural coherence. It should be noted that the algorithm in [35], unlike the approach suggested in this paper, considered the complete enumeration of possible options of matching the 3D cloud points to the object model points.

2.2.1. Stage 1

The min–max-shell in the *WCS* space is calculated for each FP of the SPS object *ob_id* (set M^{ob_id}). The absolute coordinates of the point are found using the available matrix $H_{CS^{ob\_id}, WCS}$ of the transformation of the relative coordinates of the object to the *WCS* coordinates. The linear dimension of the rectangular shell is selected by taking into account the known error in the used method of the AUV's visual navigation.

The points from the 3D cloud are checked in succession to see whether they belong to the shell. If the point is inside the shell, it may be an appropriate point of the SPS object. If there are several such points in the cloud, all of them will be deemed candidate points for matching with the analyzed point of the SPS object.

The outcome of the above check of all the object FPs is the subset of points $S^{ob\_id}\left\{P_{i_1}, \ldots P_{i_p}, \ldots P_{i_n}\right\} \in M^{ob\_id}\{P_1, \ldots P_n\}$, of which the points potentially belonging to the object were found in the 3D cloud. Here, *N*—number of dimensions of model M^{ob_id}, *n*—number of dimensions of S^{ob_id}, i_p—number of a point in set M^{ob_id}, and the index p relates to the numbering in S^{ob_id}. Each point P_{i_p} is related to the list $l_p\left\{C_{j_1}, \ldots C_{j_t}, \ldots C_{j_m}\right\}$ of the points in the 3D cloud, which are considered suitable for correspondence with this point of SPS. Here, m—length of list l_p, C_{j_t}—the point in the 3D cloud with number j_t,

t—a number in list l_p. Each list contains one or more points of the 3D cloud. Thus, the integration of lists l_{i_p} is the subset of points in the 3D cloud, in which the points belonging to the SPS are searched for and identified.

2.2.2. Stage 2

At this stage, a search is carried out for a subset in the 3D cloud that meets the criterion of structural coherence. The implementation of the criterion is based on checking the mutual distances between 3D points. Taking into account the fact that when more points are identified, the degree of object identification certainty will be higher, the search starts from longer subsets. In accordance with stage 1, set $S^{ob\_id}\{P_{i_1},\dots P_{i_n}\}$ is the model subset with the maximum length in this context.

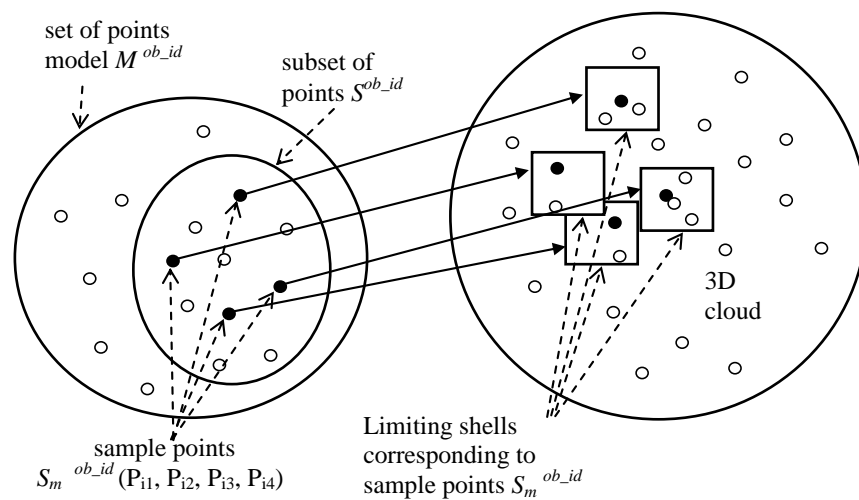The search algorithm is schematically applied as follows (Figure 4):

Figure 4. Points in min–max-shells in the 3D cloud, marked in black, are identified by the search algorithm as belonging to the SPS object.

1. On set S^{ob_id} of the points of the object model (see *stage 1*), a set of samples $s_m^{ob\_id}\{P_{i_1},\dots P_{i_q}\}$ is generated, where m—the sample number, q—the sample length. The number of possible samples is defined by the number of permutations of $n-q$ at a given time—$A_n^q = \frac{n!}{(n-q)!}$;

2. The set of distances $D_m^{ob\_id}\{d_{i_1,i_2}^{ob\_id},\dots d_{i_{q-1},i_q}^{ob\_id}\}$ is constructed, where $d_{i_k,i_s}^{ob\_id}$ is the distance between points P_{i_k} and P_{i_s}. Here, i_k and i_s are the numbers of points in the object model M^{ob_id}, and indices k and s are related to the numbering of points in sample $s_m^{ob\_id}$, which is linked to each sample $s_m^{ob\_id}\{P_{i_1},\dots P_{i_q}\}$. There are $\frac{q(q-1)}{2}$ elements in set $D_m^{ob\_id}$;

3. The set of samples $c_n^{cloud}\{c_1,\dots c_p,\dots c_q\}$, comprised of the 3D cloud points, is generated for each sample $s_m^{ob\_id}\{P_{i_1},\dots P_{i_p},\dots P_{i_q}\}$ of the object model. Here, n is the number of samples. The point from list $l_p\{C_{j_1},\dots C_{j_m}\}$, connected to point P_{i_p} (see *stage 1*), is taken as the c_p element of sample c_n^{cloud}. The number of the generated samples c_n^{cloud} is defined by the number of lists q and the lengths of these lists. For example, if $q = 3$, and the lengths of the corresponding lists are length1, length2, length3, the number of samples will be length1· length2 · length3;

4. The set of distances $D_n^{cloud}\{d_{1,2}^{cloud},\dots d_{k,s}^{cloud},\dots d_{q-1,q}^{cloud}\}$ is constructed, where $d_{k,s}^{cloud}$ is the distance between points C_k and C_s—here, indices k and s are related to the numbering of points in the sample, and c_n^{cloud} is linked to each sample c_n^{cloud}. There are $\frac{q(q-1)}{2}$ elements in set D_n^{cloud};

5.　For a sample $s_m^{ob\_id}\left\{P_{i_1}, \ldots P_{i_q}\right\}$ (step 1) from the object model, the sample $c_n^{cloud}\left\{c_1, \ldots c_q\right\}$ (par.3) from the 3D cloud is sought, such that $D_m^{ob\_id} = D_n^{cloud}$. Here, the equivalence means the equivalence between all the corresponding pairs of elements: $\left|d_{i_k,i_s}^{ob\_id} - d_{k,s}^{cloud}\right| \leq \Delta$. The error Δ is determined by the accuracy of measuring the coordinates of the 3D cloud points (depending on the resolution of pictures and the distance between the camera and the points). In that case, with consideration for the above-described rules of forming samples, the determined correspondence between sample c_n^{cloud} and sample $s_m^{ob\_id}$ enables the unambiguous identification of the points of the 3D cloud that belong to the SPS object, and for them to be matched with the object model points;

6.　If there are no corresponding points found in the 3D cloud for the specified length q of sample $s_m^{ob\_id}$, the correspondence for a smaller sample shall be searched for, i.e., for $q = q - 1$. It should be noted that the implementation of searching, aimed at detecting the maximum number of points matched to the SPS object model's points, in the 3D cloud increases the degree of certainty of object identification. Subsequent to the identification of several points (three as a minimum) belonging to the SPS object, in the 3D cloud, the coordinate referencing of the AUV to the SPS can be performed. Using more FPs would improve the accuracy of the method.

2.3. Calculation of the Matrix of the Geometric Transformation of the Points from the AUV CS to the SPS Object CS

The desired matrix referencing the coordinates of the AUV to the SPS object can be computed based on the fact that the coordinates of the SPS object's points, identified in the 3D cloud, are known both in the AUV CS (CS^{AUV_i}) and in the object CS (CS^{ob_id}). Let C_1, C_2 and C_3 be the object points identified (applying the algorithm as described above) in the 3D cloud. Let the auxiliary CS (CS^{ad}) be constructed on the identified object points, according to the rule shown in Figure 1; i.e., let unit vectors e1_AUV, e2_AUV and e3_AUV of the CS^{ad} coordinate system be constructed in the CS^{AUV_i} coordinate system.

Then, the transformation matrix $H_{CS^{ad},CS^{AUV\_i}}$, connecting CS^{ad} and CS^{AUV_i}, is formed from unit vectors:

$$H_{CS^{ad},CS^{AUV\_i}} = \begin{vmatrix} e1\_AUVx & e1\_AUVy & e1\_AUVz & 0 \\ e2\_AUVx & e2\_AUVy & e2\_AUVz & 0 \\ e3\_AUVx & e3\_AUVy & e3\_AUVz & 0 \\ rC1\_AUVx & rC1\_AUVy & rC1\_AUVz & 1 \end{vmatrix} \tag{3}$$

where rC1_AUV is the vector of CS^{ad}'s (point C_1) origin, specified in CS^{AUV_i}.

On the other hand, the coordinates of points C_1, C_2, C_3 in CS^{ob_id} are known, which means that the constructed unit vectors of the CS^{ad} coordinate system can be defined in CS^{ob_id} as well. Let $e1\_ob\_id$, $e2\_ob\_id$, $e3\_ob\_id$ denote these vectors. Accordingly, the matrix of transformation from CS^{ad} to CS^{ob_id} can be formed from the unit vectors specified in CS^{ob_id}:

$$H_{CS^{ad},CS^{ob\_id}} = \begin{vmatrix} e1\_ob\_idx & e1\_ob\_idy & e1\_ob\_idz & 0 \\ e2\_ob\_idx & e2\_ob\_idy & e2\_ob\_idz & 0 \\ e3\_ob\_idx & e3\_ob\_idy & e3\_ob\_idz & 0 \\ rC1\_ob\_idx & rC1\_ob\_idy & rC1\_ob\_idz & 1 \end{vmatrix} \tag{4}$$

Then, the desired transformation from the AUV CS to the SPS object CS can be obtained as follows (Figure 5):

$$H_{CS^{AUV\_i},CS^{ob\_id}} = \left(H_{CS^{ad},CS^{AUV\_i}}\right)^{-1} \cdot H_{CS^{ad},CS^{ob\_id}} \tag{5}$$
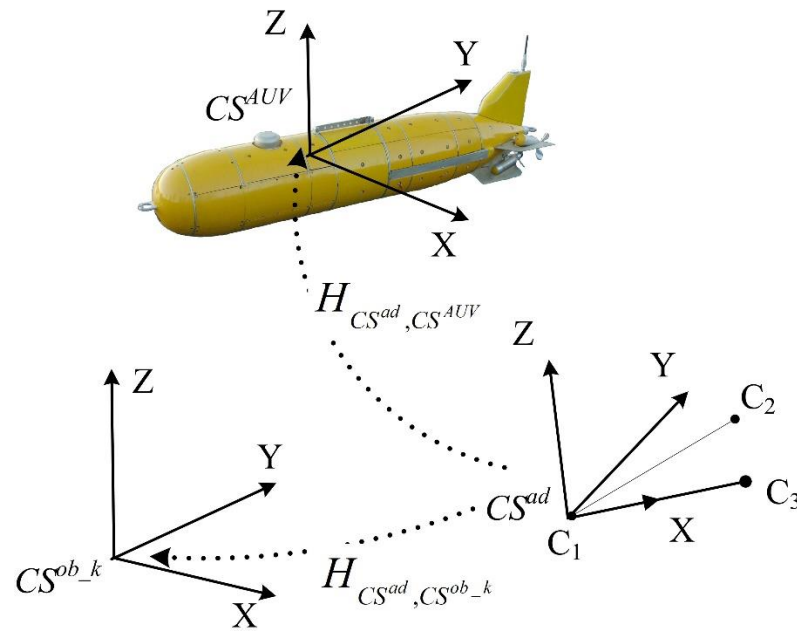
Figure 5. Calculation of the geometric transformation matrix from the AUV CS (CS^{AUV}) to the CS of the SPS object (CS^{ob_k}) using the auxiliary CS (CS^{ad}).

Other Methods for Calculating the Transformation from the AUV CS to the CS of the SPS Object

When there are more than three points of the SPS object identified in the 3D cloud, to calculate the transformation matrix $H_{CS^{AUV\_i},CS^{ob\_id}}$, a standard method can be applied to minimize the total discrepancy between the two matched sets of points that overlap within one coordinate space: $\min \sum_{k=1}^{n} \|P_k - C_k H_{CS^{AUV\_i},CS^{ob\_id}}\|$; here, P_k and C_k are the matched points of the object model and the 3D cloud, respectively. An alternative method for finding the matrix can also be utilized: the method for the fast computation of the local matrix of geometric transformation [36].

2.4. Calculation of the AUV Coordinates in the SPS CS

The parameters of the AUV's movement during the SPS inspection are computed using the visual navigation method, which ensures the calculation of the coordinates in the CS connected to some initial position of the trajectory (relative motion). These coordinates are transformed to the CS of the SPS object via the previously obtained transformation $H_{CS^{AUV\_i},CS^{ob\_id}}$. However, a one-time referencing of the AUV to the SPS CS (at the beginning of the inspection) is inadequate, since the visual navigation method is known to accumulate errors when long-term displacements occur. Hence, in accordance with the suggested procedure, the coordinate referencing of the AUV to the SPS shall be performed regularly to avoid the massive accumulation of navigation error due to the visual navigation method. Then, the AUV coordinates in the current trajectory position, derived using the visual navigation method, shall be recalculated to the SPS CS with the use of the last obtained transformation (2), as follows:

$$H_{CS^{AUV\_j},CS^{ob\_id}} = (H_{CS^{AUV\_i},CS^{AUV\_j}})^{-1} \cdot H_{CS^{AUV\_i},CS^{ob\_id}} \tag{6}$$

where *j*—current position of the AUV, *i*—position of the last coordinate referencing of the AUV to the CS of the SPS object, and $H_{CS^{AUV\_i},CS^{AUV\_j}}$—a product of local matrices $H_{l,l+1}$ computed by the method of visual navigation in each position (from pos. *i* to pos. *j*) of the trajectory.

To conclude the discussion of the suggested method of coordinate referencing, we present the following summary. We used a small number of FPs of the model with known

absolute coordinates only to optimize the search for FPs of an object in the 3D cloud. Recognizing multiple FPs of an object then allows the AUV to reference the object and work in its coordinate system. Knowledge of the absolute coordinates of the FPs of the object is not required with this approach, and nor is knowledge of the absolute coordinates of the AUV (due to the inevitable accumulation of error of absolute coordinates for visual odometry). Even if the SPS object is displaced from its original state, the coordinate reference of the AUV to that SPS object will not be affected.

3. Experiments

Since operations with a real AUV are quite expensive, we carried out computational experiments on model scenes (Figure 6) (in the simulator for an AUV [37]) and with real data obtained using a Karmin2 camera (Nerian's 3D Stereo Camera, baseline 25 cm) in laboratory conditions. The parameters of the used PC were as follows: AMD (Advanced Micro Devices) Ryzen 7 3700X 8-Core Processor 3.60 GHz, 32Gb, AMD Radeon 5600XT (Advanced Micro Devices, Santa Clara, CA, USA). Although the experiment with the Karmin2 camera was not conducted in an underwater environment but in a laboratory environment, it was useful because it allowed us to evaluate the effectiveness of the basic algorithms via calibration of a real camera (which was not ideal, as for a virtual scene). Two series of experiments were carried out. In the experiments of the first series, the error of the proposed method for the direct coordinate referencing of the AUV to the CS of the SPS object was estimated. The maximum navigation error of the AUV when moving along the expected trajectory in the SPS space was estimated in the experiments of the second series. When carrying out model experiments, it was assumed that the AUV was equipped with thrusters, could be controlled with five degrees of freedom (5-DOF), and had the ability to move in the speed range of approximately 0–2 m/s, which is optimal for this type of work.
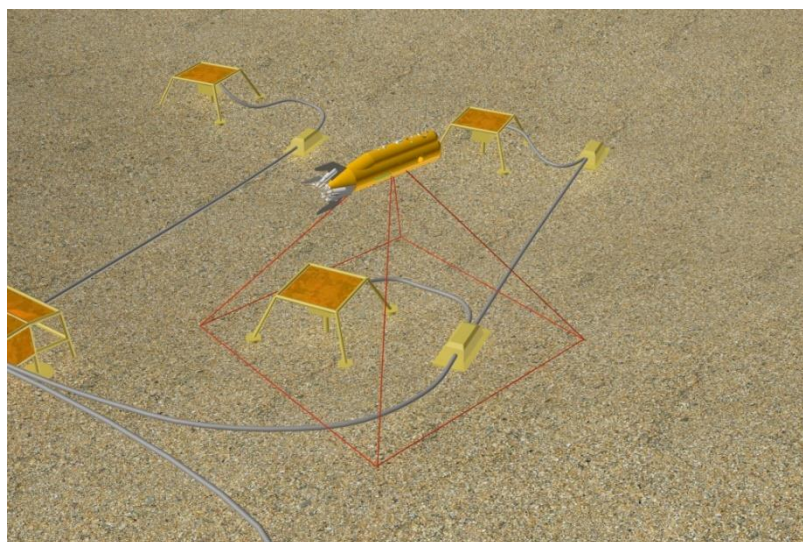

Figure 6. Virtual stage: AUV performs SPS inspection.

3.1. Experiments with a Virtual Scene

The virtual SPS included seven objects (production center manifold and wells) (Figure 1). In the model, 50 points were specified and evenly distributed over the objects.

A real texture was used in the digital seabed elevation model. The virtual camera parameters were as follows: the image resolution was 1024 × 768, the pixel size was 0.2 mm, the focal length was f = 100 mm, and the photography frequency when the AUV moved along a trajectory was 10 Hz. The AUV movement speed was set at approximately 0.2–0.5 m/s. Taking into account the fact that SPS objects are significantly far apart in space (the distance between objects is up to 50 m), trajectories with different heights of the

passage of the AUV over the SPS were tested. For a high altitude position of the AUV in relation to the SPS, several SPS objects with visible FPs fell into the camera's field of view. However, only movements at heights of 5 m or less were of practical importance, because movement at high altitudes is of little use due to the possible turbidity of the water at the work site. In addition, problems arose with the organization of lighting. At a low height (<5 m), no more than one object with a small number of FPs fell into the camera's field of view (Figures 6 and 7).
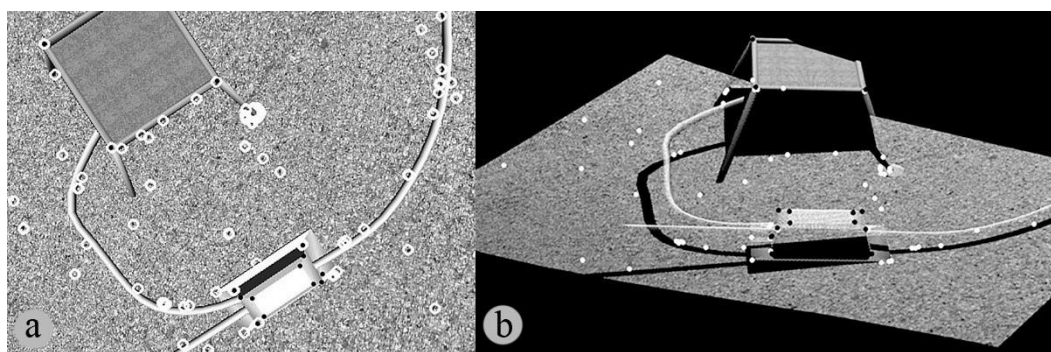

Figure 7. Identification of points belonging to the SPS object: (**a**) extraction of characteristic points by the Harris angle detector in the image taken by the camera; (**b**) a set of 3D points constructed from 2D images. All selected points in the scene are marked in white. Points belonging to SPS objects are marked in black. Six points belonging to SPS have been identified.

The geometric transformation matrix between the CS of the camera and the CS of the object was calculated in the experiment of the first type, using the described method from the identified SPS points in a 3D cloud based on a stereo pair of images (Figure 7). To calculate the matrix, the three most reliable points were selected from the set of identified FPs. Since the calibration of the stereo camera was known, before building the 3D cloud, the set of FPs was filtered based on the verification of the epipolar correspondence. Then, the calculated matrix was used to estimate the location error of FPs. The error was calculated as the difference between the calculated coordinates and the FP coordinates in the model (in the CS of the object). The resulting errors were in the range of 1.6–4 cm for a depth range of 2–5 m (this corresponds to the height of the AUV above the seabed).

In the second experiment, the movement of the AUV was carried out in a virtual scene along a trajectory that was 200 m long, with periodic coordinate referencing of the AUV to the CS of the object. The calculation of the trajectory while driving was carried out using the visual navigation program (visual odometry). The first goal of the experiment was to evaluate the effectiveness of the proposed method of object recognition and referencing to the object. Since the accumulation of navigation accuracy errors, generated by the visual method, occurs during the long-term movement of the AUV, the second goal of the experiment was to evaluate the effectiveness of the technique of regularly linking the AUV to the object. Presumably, the referencing should periodically clear the accumulated error and thus provide "constant" AUV navigation accuracy. In this case, the error in the accuracy of navigation is the sum of the error in the method of binding the AUV to the CS of the object and the error in the visual navigation method used. The time between adjacent bindings varied from 0 to 40 s. The error generated by the visual navigation method for a period of 40 s did not exceed 2 cm. Taking into account the fact that the error of the referencing method obtained in the first experiment was in the range of 1.6–4 cm (for a trajectory height of 2–5 m above the bottom), the total error was in the range of 3.6–6 cm. Thus, the experiments showed that the regular updating of the bindings in a predictable way limits the increase in accumulated navigation error.

3.2. Experiments with the Karmin2 Camera

In the first experiment, the instantaneous coordinate referencing of the camera to the CS of a complex of six objects was evaluated (Figure 8). According to the experimental technique, the operator indicated the characteristic points that belonged to the objects and represented the geometric model of the objects. The coordinates were directly measured in the fixed CS of the complex of objects.
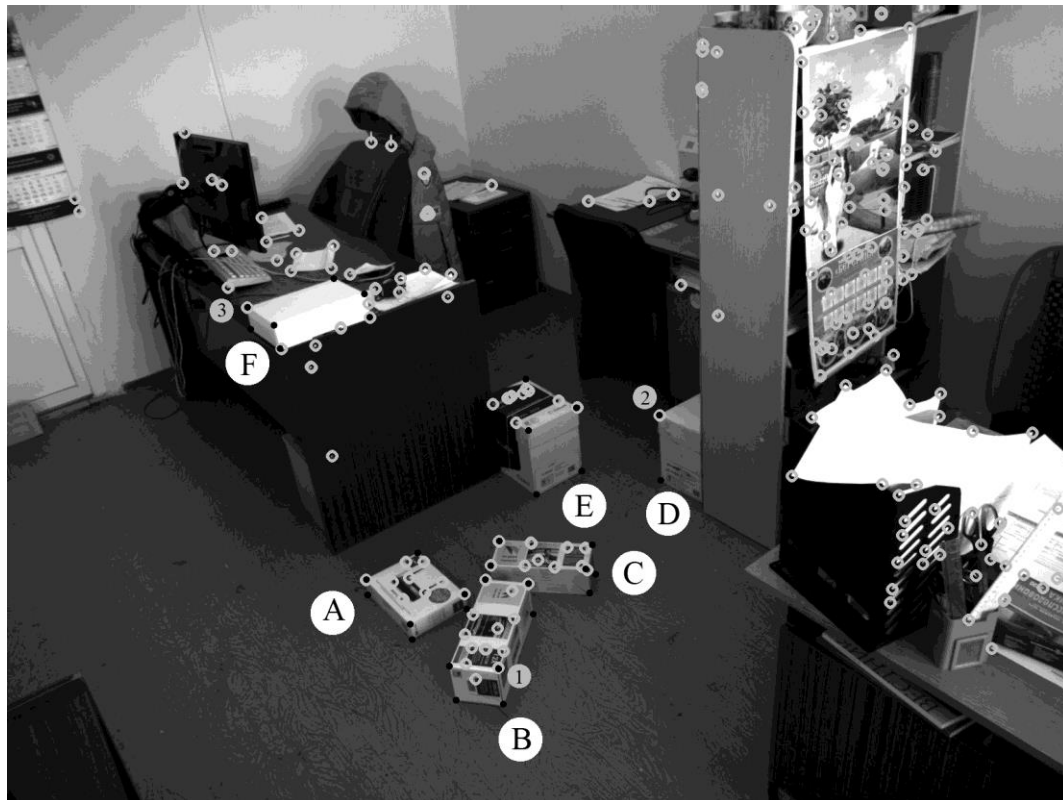

Figure 8. In the photo taken by the Karmin2 camera, the desired objects are marked: A–F. Points belonging to the desired objects (model) are marked in black. Their number is 48, as indicated by the operator showing eight on each box. Of these, 33 points fell into the camera's field of view: on object A—7, on B—7, on C—6, on D—2, on E—4, on F—7. The coordinate system (CS), in which all points of objects (model) were set, was built on three corner points of object A. The points built by the Harris detector are marked with white circles. There are 89 of them in the scene.

After using the Harris detector to produce the stereo pair images of the special points, which was achieved by calculating their 3D coordinates and processing via the described method, a set of points belonging to objects and compared with the model were identified (Figure 9). From this set, a set of three points was selected to calculate the matrix of the geometric transformation of the coordinates of points from the CS of the camera to those from the CS of the complex of objects. The calculated matrix was further used to estimate the error in the calculated location of all points of the object model.
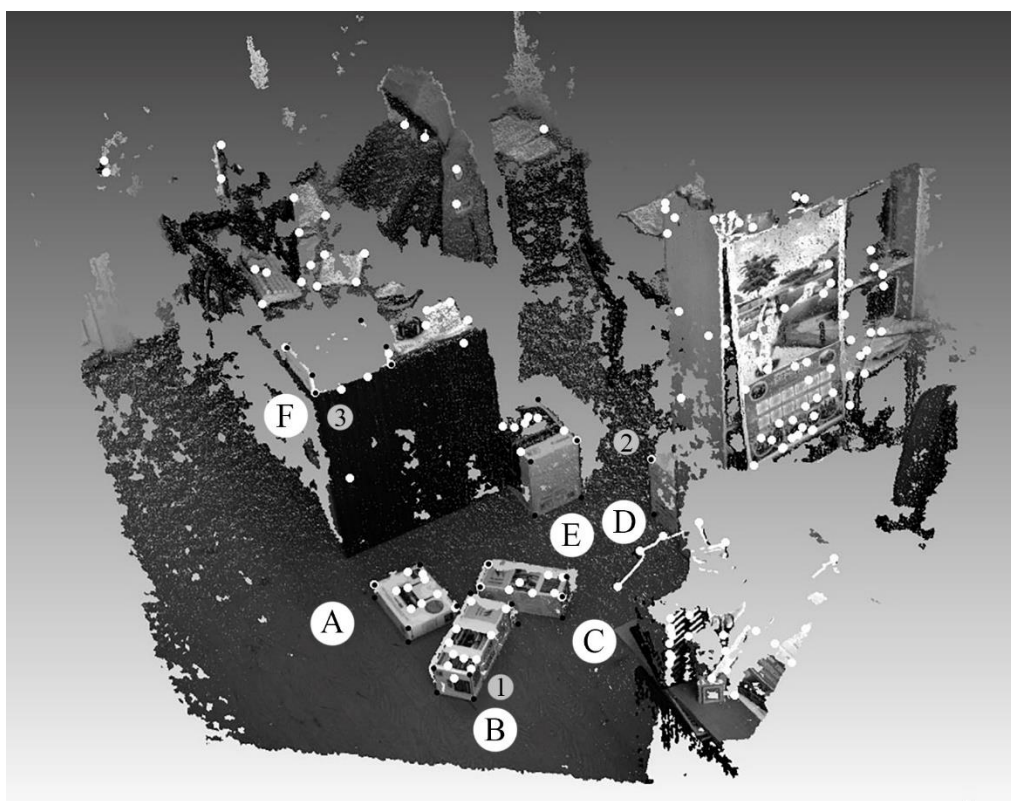
Figure 9. The figure shows the points identified in the 3D cloud (marked in black) as belonging to the sought objects. Their number was 13: on object A—2, on B—3, on C—3, on D—1, on E—1, on F—3. The matrix connecting the CS of objects with the CS of the camera was calculated by 3 points (they are marked with numbers 1, 2, 3), which were selected by the algorithm from the found points.

The camera parameters used were as follows: the image resolution was 1600×1200, the pixel size was 4.45 μm, the focal length was 6 mm and the shooting frequency was fps = 10. At a distance of 3–3.5 m from objects in the scene (which corresponds to the height of the AUV's passage above the seabed), the measured errors were in the range of 1–3 cm (0.3–0.86%). In the experiment with a distance from the objects of 1.5–2 m, the error did not exceed 0.5 cm (0.25–0.33%).

In the second experiment, the camera was moved manually to the height of 1.5 m, starting from the floor and traveling along a trajectory that was 30 m long, at a speed of $\approx$0.25 m/s. The regularity of the referencing of the AUV to an object was set by the tuning parameter of the method. In this experiment, the coordinate referencing was updated every 10 m to prevent the accumulation of the errors generated by the visual method.

The error was calculated as the difference between the calculated and the directly measured coordinates (in the CS of the object). The accumulated error during movement was within 2 cm. Thus, the resulting navigation error did not exceed 2.5 cm.

3.3. The Discussion of the Results and Comparison with Other Approaches

An inspection mission requires the recognition of underwater objects and the precise localization of the AUV in the object's coordinate space. In the performed experiments, the SLAM algorithm was used to calculate the trajectory of the AUV (author's implementation, [34,36]). However, the emphasis in this work is on the method proposed for recognizing an underwater object using an estimate of AUV position in the CS of an SPS object.

There are many works on 3D object recognition in underwater scenes, and many proposed methods in this area. Many existing methods focus on a specific type of object or scene, or require prior segmentation. A more universal approach was proposed in [38].

Here, recognition is based on the recognition of pipes, planes and their combinations, with the simultaneous creation of library models, which makes it possible to recognize more complex scenes later. In [18], an overview and a comparison of state-of-the-art methods for object recognition are provided, which are intended to assist AUVs in performing autonomous interventions in underwater inspection, maintenance and repair applications. At the conceptual level, a typical local feature-based 3D object recognition system consists of three main phases: 3D keypoint detection, local surface feature description and surface matching. A detailed description of some of the methods can be found in [39]. The surface feature description stage extracts geometric information that is encoded into a representative feature descriptor. In addition to characteristic points, surface curvature, edges and contour information, specific surface elements are used as 3D shape feature objects. At the stage of "surface matching", the object is recognized directly using the existing model (or library of models). As noted in [18], the main bottlenecks of existing methods include the presence of occlusions and the high computation cost in scenes. Comparing the method proposed in this article with those considered above, we note the following. The proposed recognition method, based on the model represented by characteristic points, corresponds to the general approach, but without reference to specific surface shapes. The main difference is associated with the specific formulation of the problem (the presence of several points of the object with absolute coordinates), which made it possible to implement an effective algorithm for finding points in a 3D cloud corresponding to the model. Compliance is based on the implementation of the structural coherence criterion. Efficiency is achieved due to the construction of limited shells in 3D space, within which the search for points associated with the model is carried out. This method of searching for points firstly reduces the likelihood of erroneous comparisons, and secondly reduces the amount of checks and associated computations. The experiments carried out (on two types of scenes) confirmed the efficiency of the proposed method for underwater inspection with an acceptable navigation accuracy and a relatively low computational complexity. Using the technique of the regular binding of the AUV to the CS of an SPS object enabled the elimination of the accumulated visual odometry error during movement, and the planning of the trajectory in the space of the scene with predictable accuracy, which is necessary for the reliable implementation of inspection missions with an autonomous robot.

Of course, as many researchers note, in a real underwater environment, the negative influence of the external environment (low illumination, turbidity of water, currents) limits the effectiveness of visual methods of navigation and object recognition. However, it is possible to reduce this negative impact through special techniques; in particular, methods based on data filtering. For example, in [40], the authors proposed an approach that allows satisfactory visual navigation in an environment when visibility conditions are far from ideal. The method discussed in our work is based on processing a 3D point cloud obtained in a standard way using the SURF detector. Therefore, we believe that the more thorough filtering of data can improve the quality of the initial 3D data and accordingly keep the efficiency of the method at an acceptable level. It is also possible to take into account the influence of currents in the method due to the corresponding correction. These issues will be addressed in future work.

4. Conclusions

The paper presents a new approach to ensuring accurate AUV navigation in the SPS coordinate space when performing underwater inspection based on processing stereo images. Its distinctive features are as follows:

1. The object recognition algorithm uses a predetermined 3D point model of the object, in which there are a limited number of characteristic points with known absolute coordinates;
2. The method uses a structural coherence criterion when comparing the 3D points of an object with a model;

3. The method references the AUV coordinate matrix to the object using the matched points;
4. High accuracy during the continuous movement of an AUV in SPS space is ensured by regular referencing to the SPS object coordinate system.

The experiments carried out on the model scenes and the local experiment performed with the real data in principle confirmed the appropriateness of the suggested method and its potential effectiveness for SPS applications. The method can also be applied when inspecting other artificial objects for which there is an a priori model.

The further development of this work will involve conducting underwater experiments to study the effect of ocean currents and water turbidity on navigation accuracy. It is also planned to improve the algorithmic base by: (a) including the visible edges of the object in the object model; (b) using the semi-automatic formation of the object model derived from the images of the preliminary AUV trajectory, and (c) taking into account the "loop closing problem".

References

1. Mai, C.; Pedersen, S.; Hansen, L.; Jepsen, K.L.; Yang, Z. Subsea infrastructure inspection: A review study. In Proceedings of the 6th International Conference on Underwater System Technology: Theory and Applications, Penang, Malaysia, 13–14 December 2016. [CrossRef]
2. Zhang, Y.; Zheng, M.; An, C.; Seo, J.K.; Pasqualino, I.; Lim, F.; Duan, M. A review of the integrity management of subsea production systems: Inspection and monitoring methods. *Ships Offshore Struct.* **2019**, *14*, 789–803. [CrossRef]
3. Paull, L.; Saeedi, S.; Seto, M.; Li, H. AUV Navigation and Localization: A Review. *IEEE J. Ocean. Eng.* **2013**, *39*, 131–149. [CrossRef]
4. Terracciano, D.; Bazzarello, L.; Caiti, A.; Costanzi, R.; Manzari, V. Marine Robots for Underwater Surveillance. *Curr. Robot. Rep.* **2020**, *1*, 159–167. [CrossRef]
5. Underwater Inspection System Using an Autonomous Underwater Vehicle ("AUV") in Combination with a Laser Micro Bathymetry Unit (Triangulation Laser) and High-Definition Camera. Available online: https://patents.google.com/patent/WO2015134473A2 (accessed on 15 September 2021).
6. Vidal, E.; Palomeras, N.; Istenič, K.; Hernández, J.D.; Carreras, M. Two-Dimensional Frontier-Based Viewpoint Generation for Exploring and Mapping Underwater Environments. *Sensors* **2019**, *19*, 1460. [CrossRef]
7. Maurelli, F.; Carreras, M.; Salvi, J.; Lane, D.; Kyriakopoulos, K.; Karras, G.; Fox, M.; Long, D.; Kormushev, P.; Caldwell, D. The PANDORA project: A success story in AUV autonomy. In Proceedings of the OCEANS 2016, Shanghai, China, 10–13 April 2016.
8. Wirth, S.; Carrasco, P.L.N.; Oliver-Codina, G. Visual odometry for autonomous underwater vehicles. In Proceedings of the 2013 MTS/IEEE OCEANS, Bergen, Norway, 10–13 June 2013; pp. 1–6. [CrossRef]
9. Albiez, J.; Cesar, D.; Gaudig, C.; Arnold, S.; Cerqueira, R.; Trocoli, T.; Mimoso, G.; Saback, R.; Neves, G. Repeated close-distance visual inspections with an AUV. In Proceedings of the OCEANS 2016 MTS/IEEE Monterey, San Diego, CA, USA, 19 September 2016; pp. 1–8. [CrossRef]
10. Jung, J.; Li, J.-H.; Choi, H.-T.; Myung, H. Localization of AUVs using visual information of underwater structures and artificial landmarks. *Intell. Serv. Robot.* **2016**, *10*, 67–76. [CrossRef]
11. Gao, J.; Wu, P.; Yang, B.; Xia, F. Adaptive neural network control for visual servoing of underwater vehicles with pose estimation. *J. Mar. Sci. Technol.* **2016**, *22*, 470–478. [CrossRef]
12. Xu, H.; Oliveira, P.; Soares, C.G. L1 adaptive backstepping control for path-following of underactuated marine surface ships. *Eur. J. Control.* **2020**, *58*, 357–372. [CrossRef]
13. Fan, S.; Liu, C.; Li, B.; Xu, Y.; Xu, W. AUV docking based on USBL navigation and vision guidance. *J. Mar. Sci. Technol.* **2018**, *24*, 673–685. [CrossRef]
14. Jacobi, M. Autonomous inspection of underwater structures. *Robot. Auton. Syst.* **2015**, *67*, 80–86. [CrossRef]

15. Ferrera, M.; Moras, J.; Trouvé-Peloux, P.; Creuze, V. Real-Time Monocular Visual Odometry for Turbid and Dynamic Underwater Environments. *Sensors* **2019**, *19*, 687. [CrossRef]
16. Zacchini, L.; Bucci, A.; Franchi, M.; Costanzi, R.; Ridolfi, A. Mono visual odometry for Autonomous Underwater Vehicles navigation. In Proceedings of the 2019 OCEANS-Marseille, Marseille, France, 17–20 June 2019; pp. 1–5. [CrossRef]
17. Sangekar, M.N.; Thornton, B.; Bodenmann, A.; Ura, T. Autonomous Landing of Underwater Vehicles Using High-Resolution Bathymetry. *IEEE J. Ocean. Eng.* **2019**, *45*, 1252–1267. [CrossRef]
18. Himri, K.; Ridao, P.; Gracias, N. 3D Object Recognition Based on Point Clouds in Underwater Environment with Global Descriptors: A Survey. *Sensors* **2019**, *19*, 4451. [CrossRef]
19. Neves, G.; Ruiz, M.; Fontinele, J.; Oliveira, L. Rotated object detection with forward-looking sonar in underwater applications. *Expert Syst. Appl.* **2019**, *140*, 112870. [CrossRef]
20. Nađ, Đ.; Mandić, F.; Mišković, N. Using Autonomous Underwater Vehicles for Diver Tracking and Navigation Aiding. *J. Mar. Sci. Eng.* **2020**, *8*, 413. [CrossRef]
21. González-García, J.; Gómez-Espinosa, A.; Cuan-Urquizo, E.; García-Valdovinos, L.G.; Salgado-Jiménez, T.; Cabello, J.A.E. Autonomous Underwater Vehicles: Localization, Navigation, and Communication for Collaborative Missions. *Appl. Sci.* **2020**, *10*, 1256. [CrossRef]
22. Tamjidi, A.; Ye, C. A pose estimation method for unmanned ground vehicles in GPS denied environments. In Proceedings of the SPIE—The International Society for Optical Engineering, Baltimore, MD, USA, 25 May 2012; pp. 83871K–83871K-12. [CrossRef]
23. Burguera, A.; Bonin-Font, F.; Oliver, G. Trajectory-Based Visual Localization in Underwater Surveying Missions. *Sensors* **2015**, *15*, 1708–1735. [CrossRef]
24. Papadopoulos, G.; Kurniawati, H.; Shariff, A.S.B.M.; Wong, L.J.; Patrikalakis, N.M. Experiments on Surface Reconstruction for Partially Submerged Marine Structures. *J. Field Robot.* **2013**, *31*, 225–244. [CrossRef]
25. Li, A.Q.; Coskun, A.; Doherty, S.M.; Ghasemlou, S.; Jagtap, A.S.; Modasshir, M.; Rahman, S.; Singh, A.; Xanthidis, M.; O'Kane, J.M.; et al. Experimental Comparison of Open Source Vision-Based State Estimation Algorithms. *Int. Symp. Exp. Robot.* **2017**, 775–786. [CrossRef]
26. Mur-Artal, R.; Montiel, J.M.M.; Tardos, J. ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Trans. Robot.* **2015**, *31*, 1147–1163. [CrossRef]
27. Schonberger, J.L.; Frahm, J.-M. Structure-from-Motion Revisited. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 4104–4113. [CrossRef]
28. Cadena, C.; Carlone, L.; Carrillo, H.; Latif, Y.; Scaramuzza, D.; Neira, J.; Reid, I.; Leonard, J.J. Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Trans. Robot.* **2016**, *32*, 1309–1332. [CrossRef]
29. Vu, M.T.; Le, T.-H.; Thanh, H.L.N.N.; Huynh, T.-T.; Van, M.; Hoang, Q.-D.; Do, T.D. Robust Position Control of an Over-actuated Underwater Vehicle under Model Uncertainties and Ocean Current Effects Using Dynamic Sliding Mode Surface and Optimal Allocation Control. *Sensors* **2021**, *21*, 747. [CrossRef]
30. Vu, M.T.; Choi, H.-S.; Nguyen, N.D.; Kim, S.-K. Analytical design of an underwater construction robot on the slope with an up-cutting mode operation of a cutter bar. *Appl. Ocean Res.* **2019**, *86*, 289–309. [CrossRef]
31. Vu, M.T.; Jeong, S.-K.; Choi, H.-S.; Oh, J.-Y.; Ji, D.-H. Study on down-cutting ladder trencher of an underwater construction robot for seabed application. *Appl. Ocean Res.* **2018**, *71*, 90–104. [CrossRef]
32. Vu, M.T.; Van, M.; Bui, D.H.P.; Do, Q.T.; Huynh, T.-T.; Lee, S.-D.; Choi, H.-S. Study on Dynamic Behavior of Unmanned Surface Vehicle-Linked Unmanned Underwater Vehicle System for Underwater Exploration. *Sensors* **2020**, *20*, 1329. [CrossRef]
33. Vu, M.T.; Choi, H.-S.; Nhat, T.Q.M.; Nguyen, N.D.; Lee, S.-D.; Le, T.-H.; Sur, J. Docking assessment algorithm for autonomous underwater vehicles. *Appl. Ocean Res.* **2020**, *100*, 102180. [CrossRef]
34. Bobkov, V.A.; Kudryashov, A.; Mel'Man, S.V.; Shcherbatyuk, A.F. Autonomous Underwater Navigation with 3D Environment Modeling Using Stereo Images. *Gyroscopy Navig.* **2018**, *9*, 67–75. [CrossRef]
35. Bobkov, V.A.; Kudryashov, A.P.; Inzartsev, A.V. Technology of AUV High-Precision Referencing to Inspected Object. *Gyroscopy Navig* **2019**, *10*, 322–329. [CrossRef]
36. Bobkov, V.A.; Mel'Man, S.V.; Kudryashov, A. Fast computation of local displacement by stereo pairs. *Pattern Recognit. Image Anal.* **2017**, *27*, 458–465. [CrossRef]
37. Melman, S.; Pavin, A.; Bobkov, V.; Inzartsev, A. Distributed simulation framework for investigation of autonomous underwater vehicles' real-time behavior. In Proceedings of the OCEANS'15 MTS/IEEE, Washington, DC, USA, 19–22 October 2015. [CrossRef]
38. Pang, G.; Qiu, R.; Huang, J.; You, S.; Neumann, U. Automatic 3D industrial point cloud modeling and recognition. In Proceedings of the 14th IAPR International Conference on Machine Vision Applications (MVA), Tokyo, Japan, 18–22 May 2015; pp. 22–25. [CrossRef]
39. Guo, Y.; Bennamoun, M.; Sohel, F.; Lu, M.; Wan, J.; Kwok, N.M. A Comprehensive Performance Evaluation of 3D Local Feature Descriptors. *Int. J. Comput. Vis.* **2015**, *116*, 66–89. [CrossRef]
40. Pérez-Alcocer, R.; Torres-Méndez, L.A.; Olguín-Díaz, E.; Maldonado-Ramírez, A.A. Vision-Based Autonomous Underwater Vehicle Navigation in Poor Visibility Conditions Using a Model-Free Robust Control. *J. Sensors* **2016**, *2016*, 1–16. [CrossRef]