

Algorithm of Dynamic Forming One Whole Raster Photo Map of the Seabed During its Shooting by an Autonomous Uninhabited Underwater Vehicle

A A Timoshenko^{1,2}, A V Zuev¹ and E S Mursalimov¹

¹Institute of Marine Technology Problems of Far Eastern Branch of the Russian Academy of Sciences, 5a Sukhnova str., Vladivostok 690091, Russia

²Far Eastern Federal University, 8 Sukhnova str., Vladivostok 690091, Russia

E-mail: {timoshal, zuev, mursalimov} @ marine.febras.ru

Abstract. An algorithm has been developed for creating one whole raster photo map of the seabed from images obtained from vertically downward cameras of autonomous underwater vehicles (AUV) using tile graphics. Tile representation of graphical information allows to quickly access the images after lifting the AUV to the surface and reduce the time spent by the operator to analyse the results of the mission. The images were combined using simple geometric transformations based on the data received from the navigation systems of the AUV and the parameters of its camera, so the algorithm can be implemented on the AUV with low-performance onboard computer, as shown in the experiment.

1. Introduction

Autonomous underwater vehicles (AUVs) are already being used to perform a number of tasks under water, for example, inspection of pipelines and underwater oil production facilities, search for objects on the surface of the bottom, including biological, archaeological research of coastal waters, etc. To solve these problems, the AUV is equipped with technical vision systems (TVS). It helps to carry out the survey of the sea area set by the operator. During the survey AUV storages obtained images on the onboard storage device. The volume of accumulated data during the execution of hours-long survey missions is such that copying them to the surface computer for further analysis requires several hours even with use of high speed wired data transfer interfaces. This problem is especially critical when quick searching for sunken in accidents objects is needed.

For communication with a carrier vessel or a coastal control post, almost all modern AUVs are equipped with wireless acoustic and LAN data transmission interfaces [1-3]. There are active developments in the field of wireless data transmission from AUVs, which in the future will allow analyzing data during the mission. There are three main groups of wireless data transmission interfaces: radio frequency, acoustic and optical interfaces. Due to low penetrating ability of radio waves in water, the first group is usually used when AUV is raised to the surface, while the remaining groups can be used for underwater data transmission. The paper [4] describes an approach in which only the part of the video data that contains the objects of interest is transmitted through the acoustic channel. An experiment, performed in [4], shows that bitmaps can be transmitted through considerable distances, but the data loss is too high for real-time image transmission and analyses. Optical communication channels



are used more successfully for continuous wireless transmission of bitmaps. They have a significantly higher data transfer rate, but strongly depend on the transmission distance [5-7]. This technology allows operators to analyze data from the AUV in real time, but only for shallow dives and with direct visibility between receiver and transmitter.

Therefore, it is relevant to quickly access the images obtained during the missions, which should be organized on board AUV in such manner that they could be available for analyses without copying immediately after the AUV rises to the surface, when wireless communication interfaces are usually activated. One of the ways to perform this task is to combine obtained images in a one whole raster photo map during its shooting by AUV.

2. Related work

The approach of combining a sequence of images into a single raster map has already been used in [8-10] for aerial, underwater, and satellite imagery. The main effective methods of combining intersecting images used in these studies are the Scale Invariant Feature Transform (SIFT) [11], in conjunction with the random sample consensus algorithm (RANSAC) [12], and the Fourier-Mellin Invariant descriptor (FMI) [8, 9]. But the use of these methods is not possible for multi-hour AUVs' missions, because of the limited performance of the onboard controller and accumulating error of combining photos. To improve the accuracy of image matching, the authors of [13] propose a method in which a photo map is constructed from video frames taken at different depths. The obvious disadvantage of this approach is the need to pass several times over the same area, as well as the inability to build a map during the movement of the AUV, which is contrary to the task set above.

Another example of combining images is the construction of a transect for studying the biocenosis of water spaces using specialized equipment. In [14], a system consisting of a cascade of cameras extending at a certain depth behind a floating vehicle is presented for the automated construction of a transect in the form of a photo image and a 3D model. This system allows to get high-quality images and a model of the studied area, but all the processing of the image sequence takes place after the AUV mission. In addition, there should be no waves and undercurrents that can cause errors in determining the position of the cameras in space.

In this paper we propose a new algorithm for combining images into a photo map of the seabed directly during its shooting. To reduce the load on the onboard AUV's computer and improve performance of the photo map in long-term missions we use the navigation data of the AUV and the parameters of its camera for the image mosaicing. To provide the quick access to the map after mission via wireless interfaces it should be organized as tile map. Tiles are small images of the same size, from which a large one is made up. Tile graphics are widely used in geographic information systems (GIS). At first, the map presented this way is stored as separate images, which can be useful for automated analysis using recognition algorithms. At second, it can be easily scaled and conveniently viewed using Web interfaces.

3. Description of the algorithm

The main purpose of the algorithm is to merge individual images of a certain area of seabed into a single scalable photo map. The images that come from the TWS are combined using the navigation data of the AUV, namely the position in the horizontal plane, the height above the bottom, the angle of the course and the camera parameters that determine the characteristics of the images. This is achieved by scaling, rotating, and shifting each new image, then dividing it into tiles of the same size and overlaying them on the generated map. To be able to quickly zoom the map, it is represented in several layers, each is a four times reduced copy of the previous layer. Figure 1 shows the general scheme of the algorithm. The dotted line on the diagram indicates the processes that are performed cyclically. Next, we will consider all the stages of the algorithm in more detail.

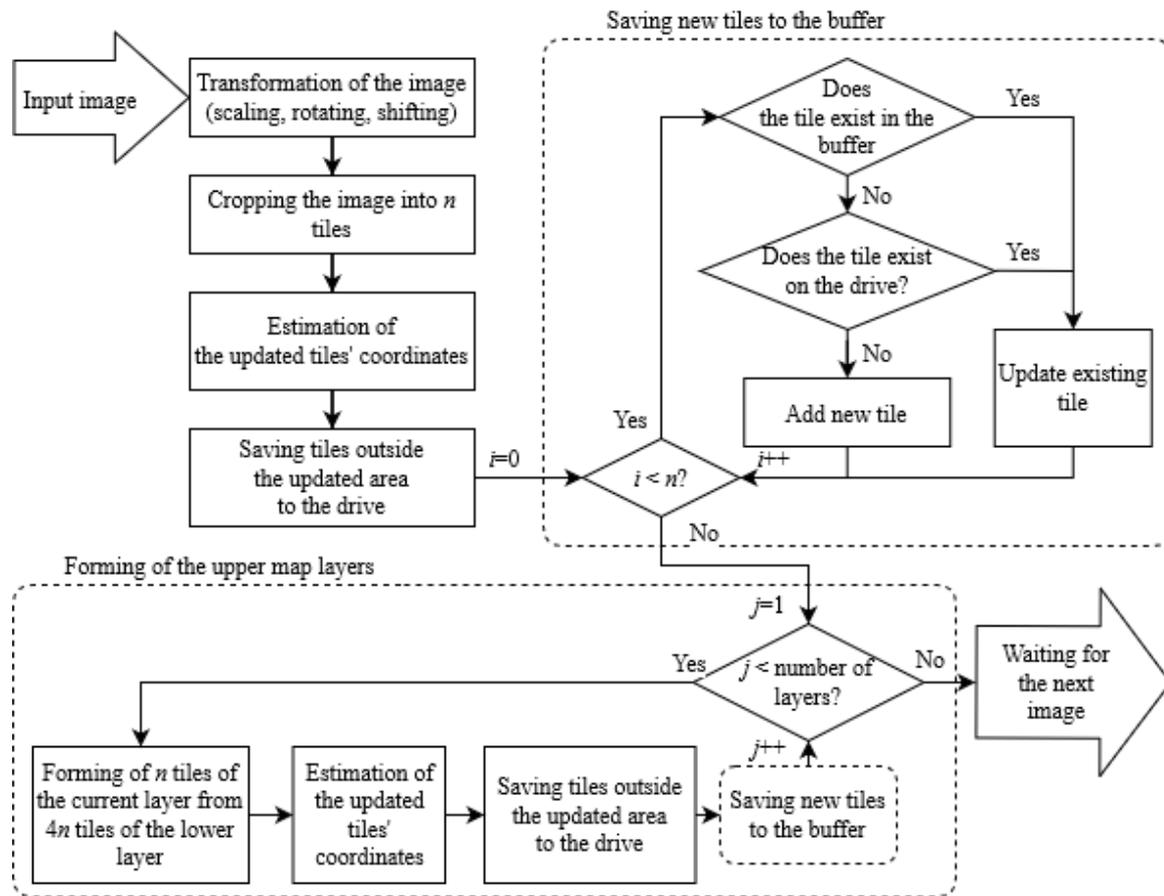


Figure 1. General scheme of an algorithm.

At the input, the algorithm receives an image that needs to be transformed (rotated and scaled) before being divided into tiles, so that all objects on two separate intersecting images are equally oriented and have the same size.

In general, an image is a matrix of pixels $C = R^{[k \times n]}$, the coordinates of any pixel in the image can be expressed as a vector $\mathbf{r} = [x \ y]^T$. The image is rotated and scaled with the next rotation matrix:

$$A^{[2 \times 3]} = \begin{bmatrix} \alpha & \beta & x_c(1-\alpha) - y_c\beta \\ -\beta & \alpha & x_c\beta - y_c(1-\alpha) \end{bmatrix}, \quad (1)$$

where $0 \leq x_c \leq n-1$, $0 \leq y_c \leq k-1$ – coordinates of the point around which the image is rotated; $\alpha = s \cos \theta$; $\beta = s \sin \theta$; θ – the angle of rotation of the image, in this case, the reverse angle of the AUV's course; s – scale factor.

The scale factor of the image in this case depends on the height of the vehicle above bottom and can be calculated using the formula:

$$s = \frac{H_{nom}}{H} = \frac{fm}{H}, \quad (2)$$

where H – the height at which the each picture was taken, $H_{nom} = fm$ – nominal shooting height, f – camera focus, m – the selected map scale [15].

In computer graphics, in particular when using OpenCV tools for image processing, the corners of the image can be cropped during rotation, since the rotation matrix (1) is applied to the coordinates of the pixels of the image C , but the size and orientation of the image itself does not change. Therefore, in order to avoid cutting corners when rotating the image, the center of the original image is mentally aligned with the center of the square image $\tilde{C} = R \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix}$, where d - the diagonal of the original image, s - scale factor. Then the coordinates of the pixels of the image C in the coordinate system of the image \tilde{C} will take the form $\tilde{\mathbf{r}} = [\tilde{x} \quad \tilde{y}]^T = [x - x_c + \tilde{x}_c \quad y - y_c + \tilde{y}_c]^T$, where \tilde{x}_c, \tilde{y}_c - coordinates of the center of the image \tilde{C} .

After converting the image, you need to find out its location on the photo map. To do this, the coordinates of the AUV, expressed in meters, must be converted to pixels. The number of image pixels per meter of the map at the selected scale is calculated using the formula $P = 1000 \frac{p}{m}$, where p - the number of pixels per millimeter of the camera matrix, P - the number of pixels per meter of the photo map at the nominal scale m . Coordinates of the center \mathbf{r}_C of the image converted by the formula: $\mathbf{r}_C = [x_C \quad y_C] = [Px_m \quad -Py_m]^T$, where x_m, y_m - coordinates of the center of the image, expressed in meters. Then, using the transformed AUV's coordinates we should calculate coordinates \mathbf{r}_{ul} of the tile which contains the upper-left corner of the image:

$$\mathbf{r}_{ul} = [x_{ul} \quad y_{ul}]^T = \begin{bmatrix} \frac{x_C - \frac{d}{2}}{s} & \frac{y_C - \frac{d}{2}}{s} \end{bmatrix}^T \quad (3)$$

where s - the length of one side of the tile in pixels. Then the image is expanded so that it fits into the tile grid and divides it into an integer number of tiles. Lately it will be easy to calculate the coordinates of the other tiles relatively to the upper-left tile.

The transformed image can now be split into tiles. Each tile is assigned coordinates within its own scale. Then they should be written to a physical drive, but during the experiments the fact was established that the process of saving tiles directly to the drive is the slowest process in the algorithm. Instead of direct writing, a different approach was used. Each new input image may intersect with the previous one, so the tiles describing the intersection area will be updated and overwritten with each new image. To reduce the number of overwrites and speed up the processing of images, only those tiles that will not be updated with a new input image should be saved to the drive. For this purpose, a temporary storage (buffer) is implemented in the RAM of the AUV's computer, where tiles are stored after processing the input image. If the tiles stored in the buffer do not fall within the area that the new image describes, they are written to the drive.

Next, tiles for smaller map scales should be formed. Separate buffers are created for each scale level. The tiles of the previous layer are used to compose each new one. First, the tiles of the previous layer are combined by four, and then their size is reduced by four times. Resulting tiles are assigned coordinates and with described principle they are written to the drive.

After forming a new set of basic-level tiles, the corresponding buffer is cleared: all tiles outside the area of the input image are written to the drive, since these tiles are considered finished. Then a new set of tiles is saved to the corresponding buffer according to the following algorithm. First, it checks whether there are tiles with the same coordinates in the buffer. If yes, the tiles are combined. If not, the presence of such files on the physical drive is checked. If tiles already exist, the new should be combined with them and be added to the buffer. Otherwise, new tiles are simply added to the buffer. After that, similar actions are performed for tiles of a smaller scale. The described algorithm can be implemented and effectively applied even on low-performance AUV's onboard computers.

4. Experimental verification of the developed algorithm

For the experiments, a series of images taken by the underwater vehicle MMT-3000 at a depth of approximately 35 m was used. The shooting was carried out using the onboard Videoscan-285 camera with a frequency of 5 frames per second, an ICX285 matrix with a resolution of 1392x1040 pixels and a focal length of 8 mm. The physical size of the matrix is 8.8 mm by 6.6 mm. Thus, the number of pixels per millimeter of the matrix is approximately 158. The images are black and white, the resolution is reduced by half, i.e. to 696x520 pixels. The width and height of the tiles are set to 256 pixels. The mapping starts from the 17th scale level and goes up to 0. For each image, information about the position, orientation and speed of the underwater vehicle was saved. The algorithm was tested on a “LattePanda” onboard computer with a 1.4 GHz processor. Images were read and written on a solid-state drive (SSD).

The table 1 shows a sample of the results of the experiment. The average image processing time is 0.162 seconds, which is less than the period of shooting. Figure 2 shows a fragment of the map obtained from experimental images.

Table 1. Results of the experiment.

№	Image name	Basic layer (sec)	Higher layers (sec)	General time (sec)	Tiles written
1	09203831.jpg	0.152	0.091	0.243	0
2	09203835.jpg	0.093	0.078	0.171	6
3	09203839.jpg	0.090	0.087	0.177	4
4	09203843.jpg	0.069	0.064	0.133	0
5	09203847.jpg	0.109	0.083	0.192	9
6	09203851.jpg	0.153	0.101	0.254	9
7	09203855.jpg	0.153	0.094	0.247	7
8	09203859.jpg	0.223	0.090	0.313	4
9	09203903.jpg	0.086	0.077	0.163	9
10	09203907.jpg	0.118	0.096	0.214	4

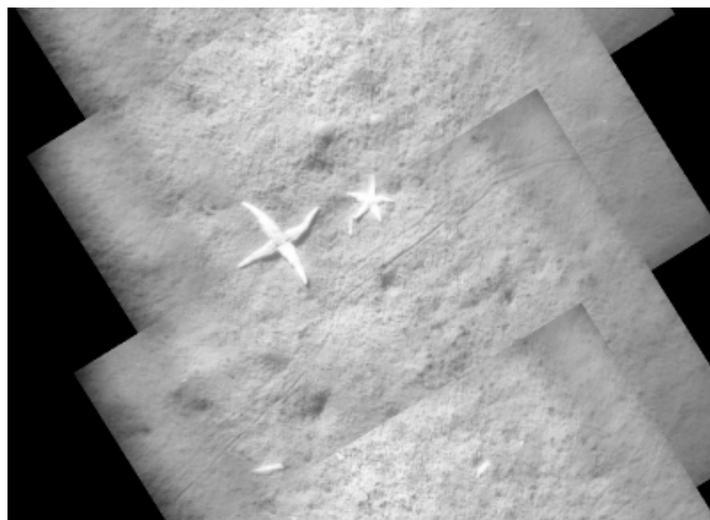


Figure 2. A fragment of the generated map.

5. Conclusion

There in this paper was proposed a new algorithm for combining images into a one whole raster photo map from a sequence of individual images or video frames using tile graphics and simple

transformations of input images based on the AUV's navigation data. The use of tiles allows to present the generated map in a convenient form both for a person and for the on-board control system of the AUV. The use of navigation data for comparing images instead of existing methods, in addition to reducing the processing time of images, also allows to link objects on the map to the global coordinate system with high accuracy. As a result of the experiments, it is clear that the developed algorithm makes it possible to use it on the onboard standard computers of the AUV. At the same time, it should be noted that the quality of image mosaicing itself is not high, but still enough for practical uses in rescue, searching and survey missions. In the future, it is planned to improve general quality of the formed map and use tiles of different scales for automatic recognition of objects and determining their location directly during the movement of the AUV.

6. References

- [1] Johansson B, Siesjö J and Furuholmen M 2010 Seaeye sabertooth a hybrid AUV/ROV offshore system *OCEANS 2010 MTS/IEEE* (Seattle: IEEE) pp 1–3
- [2] Albiez J, Joyeux S, Gaudig C, Hilljegerdes J, Kroffke S, Schoo C et al 2015 Flatfish – a compact subsea-resident inspection AUV *OCEANS 2015-MTS/IEEE* (Washington: IEEE) pp 1-8
- [3] Gornak, V E, Inzartsev A V, Lviv O Yu, Matvienko Yu V, Shcherbatyuk A F and Inzartsev A V 2007 MMT-3000 is a new small-sized autonomous uninhabited underwater vehicle of the Institute of Marine Technologies Problems of the FEB RAS *Underwater Research and Robotics* **1** 12-20
- [4] Ahn J, Yasukawa S, Sonoda T, Nishida Y, Ishii K and Ura T 2018 An optical image transmission system for deep sea creature sampling missions using autonomous underwater vehicle *IEEE J. of Oceanic Eng.* **45(2)** 350-61
- [5] Miroshnikova N E, Petruchin G S and Sherbakov A V 2019 Problems of Underwater Optical Links Modeling *2019 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO)* pp 1-9
- [6] Wu T C, Chi Y C, Wang H Y, Tsai C T and Lin G R 2017 Blue laser diode enables underwater communication at 12.4 Gbps *Sci. reports* **7(1)** 1-10
- [7] Pontbriand C, Farr N, Hansen J, Kinsey J C, Pelletier L P, Ware J and Fourie D 2015 Wireless data harvesting using the AUV Sentry and WHOI optical modem *OCEANS 2015-MTS/IEEE* (Washington: IEEE) pp 1-6
- [8] Bülow H, Birk A 2009 Fast and robust photomapping with an unmanned aerial vehicle (UAV) *2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems* pp 3368-73
- [9] Bulow H, Birk A, Unnithan V 2009 Online generation of an underwater photo map with improved fourier mellin based registration *OCEANS 2009-EUROPE* pp 1-6
- [10] Ait-Aoudia S, Mahiou R, Djebli H, and Guerrout E H 2012 Satellite and aerial image mosaicing- a comparative insight *2012 16th Int. Conf. on Information Visualization* pp 652–7
- [11] Fischler M A, Bolles R C 1981 Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography *Communications of the ACM* **24(6)** 381–95
- [12] Lowe D G 2004 Distinctive image features from scale-invariant keypoints *Int. j. of computer vision* **60(2)** pp 91–110
- [13] Gracias X, Negahdaripour S 2005 Underwater mosaic creation using video sequences from different altitudes *Proc. of OCEANS 2005 MTS/IEEE* pp 1295–300
- [14] Mizuno K, Tabeta S, Matsumoto Y, Sakamoto S, Sugimoto Y, Ogawa T et al 2018 Development of a towed optical camera array system (SSS: Speedy Sea Scanner) for sea environmental monitoring *2018 OCEANS-MTS/IEEE Kobe Techno-Oceans (OTO)* pp 1-5
- [15] Krasnopevtsev B V 2008 *Photogrammetry* (Moscow: SCP "Reprography" MSUGaC) pp 160

Acknowledgments

This work was supported by Russian Foundation for Basic Research, project 20-38-70161.