

FedCSIS_2023_paper_2821.pdf

Image Registration for Virtual Drone Simulation

1

Konrad Koniarski

Systems Research Institute

Polish Academy of Sciences

ul. Newelska 6, 01-447 Warsaw, Poland

Email: konrad.koniarski@gmail.com

Andrzej Myśliński

Systems Research Institute

Polish Academy of Sciences

ul. Newelska 6, 01-447 Warsaw, Poland

Email: andrzej.myslinski@ibspan.waw.pl

Abstract—In this paper, we introduce a method that utilizes image registration techniques to construct a 3D map of a real environment, specifically for drone flight simulation. Our approach is based on a modified version of the Iterative Closest Point (ICP) algorithm. By incorporating feature point filtering and local optical flow, we are able to expedite the processing of 3D points and focus on static scenes. One major drawback of the standard ICP algorithm is the challenge of selecting appropriate initial points to ensure effective matching between consecutive point clouds. To address this issue, our modified approach tackles this problem and optimizes the algorithm for the construction of a 3D map.

Since our application involves RGB-D image sequences, the changes between consecutive keyframes are relatively small. Therefore, we only select a small subset of key points from the source images using scale-space pyramid and FAST approaches. This selection process significantly reduces the number of processed image points. Then dense point cloud is used for refinement. Additionally, as we employ a point-matching technique using local optical flow, the expensive point-matching procedure can be avoided in each optimization step of the ICP algorithm.

We demonstrate the practical utility of our method through a virtual 3D drone flight simulator. Furthermore, we validate our proposed approach through numerical examples.

I. INTRODUCTION

A. Autonomous flying drones

In recent years, unmanned aerial vehicles (UAVs), including drones (quadrocopters), have found increasing applications in various fields such as surveillance, environment mapping, disaster monitoring, industrial inspection, and aerial photography [1]. Drones, specifically, possess the ability to fly at low speeds and hover, along with high maneuverability, making them well-suited for navigating restricted spaces like buildings. Traditionally, drones have been operated remotely by human pilots, requiring a stable connection for transmitting control commands and video streams from onboard cameras [2]. Alternatively, the operator needs to maintain a direct line of sight with the drone. These requirements limit the practicality of such systems. Consequently, the development of drones that can autonomously hold their position, avoid obstacles, and provide shared autonomy to the pilot would be a significant improvement. In this scenario, the primary task of the pilot would be to select new positions to achieve the best possible view of the situation [3], [4].

Unlike previous methods that utilize sparse visual feature points in images, our approach estimates the camera motion

by aligning consecutive images using all available image information based on the photo-consistency assumption.

To achieve precise control, it is necessary to measure the position relative to a fixed coordinate system. In outdoor scenarios, the Global Positioning System (GPS) or similar satellite-based systems can be employed. However, when operating in GPS-denied environments such as buildings, alternative sensors must be utilized for position estimation.

When all the sensors required for position estimation are carried by the drone itself, it is referred to as odometry. One of early-stage transition measurement systems using acceleration sensors, but this approach is sensitive to accumulated errors for longer paths.

One type of sensor commonly used is cameras or lidar. The camera image stream can be utilized to estimate the drone motion, and this odometry technique is known as visual odometry.

Various systems exist, including those using monocular or stereo cameras. The use of a stereo camera simplifies the problem as the absolute depth of the scene is known [1]. Standard approaches in visual odometry reduce the data processing load by extracting feature points from the images. Subsequently, motion is estimated based on correspondences between these feature points in two or more images, which are commonly referred to as sparse visual odometry methods.

With the advancement of more powerful computing resources, dense visual odometry methods have emerged. This progress has been facilitated by the availability of novel commodity stereo cameras that provide accurate color and depth information at relatively high resolutions. These cameras are known as RGB-D cameras. Dense visual odometry methods offer advantages such as increased accuracy and a simpler processing pipeline [5]. While sparse visual odometry methods have already been employed to stabilize and control drones, the utilization of dense approaches for this purpose has not been explored until recently. In literature [6], [7] there is discussion on how to apply dense and sparse visual odometry as the trade off speed and accuracy.

B. Augmented Reality

Augmented Reality (AR) is a technology that enhances our understanding of the real world by overlaying digital content, such as images, videos, or 3D models, onto our surroundings. By blending computer-generated elements with the physical

environment, AR allows us to simultaneously interact with virtual and real objects [8].

To achieve this, AR typically relies on devices like smartphones, tablets, or specialized AR glasses, which are equipped with a camera and display. These devices capture our view of the real world and project virtual content onto it, seamlessly integrating virtual objects with our physical surroundings in real-time. This integration creates an engaging and interactive experience where users can observe and interact with digital elements that appear to exist alongside the real world [9].

C. Application of AR for drone simulation

More recently, the combination of drones with virtual and augmented reality has introduced new experiences, including first-person view (FPV) flights, AR training, and mixed-reality games [1].

Learning to effectively pilot a drone can be a challenging task. Traditional remote controllers are often unintuitive, requiring the use of two joysticks to control flight parameters like pitch and roll, and throttle yaws. Additional wheels and buttons are used to manipulate the drone's camera. While basic movements can be achieved with short training sessions under relaxed circumstances, executing complex maneuvers can be more difficult. Moreover, in challenging or stressful situations, such as training, the lack of intuitive controls imposes an additional cognitive load on the pilot, impacting safety and efficiency. Therefore, the use of partial autonomy in the movement of the drone is essential, so that the operator can devote his or her attention to the task at hand rather than to controlling the drone.

II. RELATED WORK

A. Registration

Registration refers to the process of aligning multiple images or frames to create a unified 3D representation of a scene. Registration involves estimating the camera poses and positions from which the images or frames were captured, then aligning them to establish the spatial relationships between the views. This alignment is essential for accurately reconstructing the 3D geometry of the scene and obtaining a consistent and coherent representation. The registration process typically involves two main steps: feature extraction and matching. Feature extraction involves identifying distinctive points or patterns in the images or frames, such as corners or edges. These features serve as reference points for matching and aligning the views. In the matching step, corresponding features across different views are identified, and the camera poses and scene structure are estimated based on these correspondences. Once the registration is complete, the algorithm can proceed to triangulate the matched features to compute the 3D structure of the scene.

B. Rigid body motion

The typical process of geometric registration involves two main stages: global alignment and local refinement. In the global alignment stage, an initial estimate of the rigid motion

between two frames is computed. The local refinement stage then refines this initial estimation to achieve a more precise registration. The focus of this paper is primarily on the second step, which utilizes the Iterative Closest Point (ICP) method [10].

Most registration methods rely on candidate correspondences for their operation. Popular methods utilize point-to-point matches based on local geometric descriptors [11], while others define correspondences based on pairs or tuples of points [12]. Once candidate correspondences are gathered, the alignment is estimated iteratively from a sparse subset of correspondences. This iterative process often employs randomized algorithms such as RANSAC [11], [12]. However, existing pipelines may require numerous iterations to obtain a suitable correspondence set and achieve a reasonable alignment, especially in the presence of noise or partial surface overlap.

In many applications, there is prior knowledge that consecutive data observations exhibit relatively small transformations. This approach, known as local refinement, involves refining the alignment based on a rough initial estimate, leading to a tight registration that utilizes denser correspondences compared to global alignment [13]. The ICP method and its variations are commonly used for local refinement. The simplest version of the ICP algorithm begins with an initial alignment and alternates between establishing correspondences by finding the closest point and recalculating the alignment based on the current correspondence set. ICP can yield accurate results when initialized near the optimal position, but it becomes unreliable without proper initialization. Various approaches have been explored in different papers [14], [15] to increase ICP's sensitivity to local optima. Modifications of the ICP method exist that involve adjustments in the correspondence or transformation parameter estimation steps [16].

However, these existing approaches still heavily rely on a satisfactory initialization. Our research demonstrates that the accuracy achieved by well-initialized local refinement algorithms can be reliably attained without the need for initialization. Moreover, this can be achieved with a significantly reduced computational cost compared to the coarse global alignment algorithms discussed earlier.

In a study by Park [14], a modification was proposed that incorporates both geometry and the intensity of RGB color values. This approach is particularly relevant when registration involves a data stream from a source where conditions remain unchanged, such as a continuous stream of frames with consistent camera intrinsic parameters. The accumulated 3D model can take various forms, such as a volumetric representation [15], a 3D point cloud [5], or a set of depth maps.

C. Optical flow

One popular method for image processing, specifically for determining the movement (speed and direction) of image components, is optical flow. According to Akpinar et al. [17], optical flow estimation algorithms can be categorized based on their theoretical approach to interpreting optical flow. These

categories include differential techniques, region-based matching, energy-based methods, and phase-based techniques. There are two main groups of optical flow methods: local optical flow, introduced by Lucas-Kanade [18], and global optical flow, introduced by Horn and Schunck [19]. In our paper, we primarily focus on the geometric aspect of registration, making local optical flow more suitable for our purposes [8], [20]. The input data stream in our case consists of color RGB intensity frames along with depth information.

D. ICP and modifications

The typical pipeline of geometric registration consist of two stages: global alignment, which computes and initial transition estimation of the rigid motion between two time frames and local refinement, which refines the initial transition estimation to obtain a tight registration. In this paper we mostly focused on the second step where ICP method is applicable [10].

In this approach RGB-D images are used projected into 3D space as point cloud. Typically, a sample point cloud from an object's surface is obtained from two or more points of view in different reference frames. Registration task is to place the data on a common frame of reference by estimating the transformations between data sets. Since the data may contain noise and not to be accurate, the estimating the transformation parameters is difficult. A popular approach to solving this problem is the class of algorithms based on the Iterative Closest Point (ICP) technique [16].

ICP method is attractive for its simplicity, efficiency and relatively quick converge if the algorithm provided an initial estimate is enough good. This difficult form of the problem occurs when reconstructing the scene, searching for 3D objects, relocating cameras, AR and other applications. To deal with noisy data and partial overlap, practical registration pipelines use iterative model fit structures such as RANSAC. Each iteration takes a sample of a set of candidate correspondences, creates a match based on those correspondences, and evaluates that match. If a satisfactory match is found, it is refined by a local registration algorithm such as ICP.

The combination of rough alignment based on sampling and iterative local refinement is common in practice and is not designed to achieve tight registration even with difficult inputs. While such registration flows are common, they have significant drawbacks. Both the model fitting and the local refinement steps are iterative and perform computationally expensive Nearest Neighbor queries in their inner loops. Much computational effort is spent testing candidate matches, which then are rejected. And the inelegant decomposition into the global alignment stage and the local refinement stage is itself a consequence of the low precision of global frame alignment.

In literature [21], [16], [22] are proposed many methods to estimate the pose of the 3D objects including feature extraction methods, generalized surface approximations, shape-based template matching methods, deep neural networks or methods minimizing a cost metric of the measured points. Feature based methods are negatively affected with sensor noise. Also, they are computationally expensive methods for

complex shaped objects. Generalized geometric shapes are sensitive to occlusions. For template based methods a template defined with 3D meshes, and matching template requires an extensive stored library. Deep learning methods applied for pose estimation of the objects require sufficient data as well as local and global features. When the correspondences between the model and object point clouds in 3D space are not known the Iterative Closest Point (ICP) [16], [7] based methods are relatively simple and attractive solution to this pose estimation problem.

Nevertheless, these approaches still rely on a satisfactory initialization. Our work demonstrates that the accuracy achieved by well-initialized local refinement algorithms can be achieved reliably without an initialization, at a computational cost that is more than an order of magnitude lower than the coarse global alignment algorithms described earlier.

III. PROPOSED METHOD

Our objective is to compute the rigid body motion transition, denoted as T , which comprises a translation vector t and a rotation matrix. This transition aims to minimize the disparity between two sets of points, O and M . In the following subsections, we describe the process of identifying, matching, and filtering suitable feature points. Additionally, the subsequent section presents our proposed calculation for pose estimation.

A. Detection and Matching of Visual Features

An RGB-D image encompasses a color image I and a depth image D captured within the same coordinate framework. We assume the availability of a pair of RGB-D images (I_i, D_i) and (I_j, D_j) , along with an initial transformation T^0 that roughly aligns (I_i, D_i) with (I_j, D_j) . Moreover, let $p = (u, v)^T$ represent a pixel in (I_i, D_i) , and $p' = (u', v')^T$ denote the corresponding pixel in (I_j, D_j) .

Our aim is to determine the optimal transformation that achieves dense alignment between the two RGB-D images. Given that the individual frames are temporally close, such that the color intensity of pixel does not change rapidly, we can set the initial match T^0 as the identity matrix.

The initial step in registration involves selecting feature points using the FAST method on the RGB image I . By employing the FAST point detection method, we obtain a set of feature points, denoted as P . To address issues related to scale and rapid movement of individual points, we construct a scale pyramid. The original RGB image I serves as the first layer, and subsequent layers are generated by successively reducing the image size by a factor of 2. Consequently, we obtain a set $N = P_k$ of points, where k represents the number of layers in the pyramid.

Next, we employ the local optical flow method to track the selected points $N_{i \rightarrow j}$ from image I_i onto the consecutive image I_j . Let $O_{i \rightarrow j}$ denote the set of successfully tracked points, and $C_{i \rightarrow j}$ represent their corresponding movement vectors.

Then, the mapping of the RGB-D image pixel, $p = (u, v)^T$, with its corresponding depth value, $d = D(p)$, to 3D space is achieved through the projection function, denoted as π :

$$\pi(u, v, d) = \left[\frac{(u - c_x)d}{f_x}, \frac{(v - c_y)d}{f_y}, d, 1 \right]^T, \quad (1)$$

Here, f_x and f_y represent the camera focal lengths, and (c_x, c_y) denotes the principal point. Conversely, the inverse projection function, π^{-1} , is defined as follows:

$$\pi^{-1}(x, y, z, 1) = \left(\frac{xf_x}{z} + c_x, \frac{yf_y}{z} + c_y, z \right). \quad (2)$$

By applying the projection function, π , to the set of points, $O_{i \rightarrow j}$, the 3D point set, $W_{i \rightarrow j}$, is obtained:

$$W_{i \rightarrow j} = \pi(O_{i \rightarrow j}) \quad (3)$$

In practice, certain RGB-D image pixels may lack a defined depth component, rendering them unable to be projected into 3D space. In such cases, these pixels are not used for 3D projection. The set $W_{i,j}$ is also referred to as the point cloud.

The photometric objective is to find a transformation, T that satisfies:

$$p = \pi^{-1}(T\pi(p', D(p'))). \quad (4)$$

This implies that the projection of pixels, p and p' , should correspond to the same point in 3D space:

$$\pi(p, D(p)) = T\pi(p', D(p')). \quad (5)$$

36

B. Pose Estimation

The Iterative Closest Point (ICP) algorithm consists of two steps: correspondence estimation and transformation parameter estimation. In the first iteration, we consider two consecutive image frames: the current (I_i, D_i) and the registered (I_j, D_j). The objective of the correspondence estimation step in ICP is to establish a mapping function, ϕ , that defines the correspondence between I_i and I_j :

$$p = \phi(p'). \quad (6)$$

In the proposed method, the function ϕ is defined by $C_{i \rightarrow j}$, and it does not need to be recalculated at each loop step.

The estimation of transformation parameters is accomplished by seeking a transformation, T that minimizes the objective function:

$$\begin{aligned} T_{n+1} &= \arg\min_T \sum_{i=1}^O \left\| \pi(p_i, D(p_i)) - T_n \pi(p'_i, D(p'_i)) \right\|^2 \\ &= \arg\min_T \sum_{i=1}^O \left\| \pi(\phi(p'_i), D(\phi(p'_i))) - T_n \pi(p'_i, D(p'_i)) \right\|^2, p_i, p'_i \in O \end{aligned} \quad (7)$$

Here, n denotes the iteration step counter. The computations may be terminated after a predetermined number of iterations or when the estimate.

The iteration step counter, n , determines the completion of computations, either when the predetermined number of iterations has been reached or when the estimated T is deemed sufficient.

The optimization problem (7) is solved using the Gauss-Newton method. In each iteration, T is locally linearized as a 6-element vector, $\xi = (\omega_1, \omega_2, \omega_3, t_1, t_2, t_3)$, which consists of the rotation component, ω , and the translation component, t :

$$T \approx \begin{pmatrix} 1 & -\omega_3 & \omega_2 & t_1 \\ \omega_3 & 1 & -\omega_1 & t_2 \\ -\omega_2 & \omega_1 & 1 & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} T_n \quad (8)$$

By utilizing the Gauss-Newton optimization scheme, ξ is calculated by solving the linear system:

$$J_r^T J_r \xi = -J_r^T r \quad (9)$$

Here, r represents the residual vector, and J_r denotes the Jacobian. In each optimization step, n , r and J_r are computed. Subsequently, T_n is derived from ξ (see equation 8). The next update of T involves applying T_n using the transformation to the SE(3) group.

The calculation of transition T between two consecutive frames can be extended for the next frames, then the sequence of frames can be merged in one point cloud. The stability of this method depends mainly on key point selection and optimization criteria. There are techniques for improving the quality of calculation and reducing drift. One of them is using keyframes and triangulation of keyframes to improve performance.

C. 3D drone simulator

The presented environment mapping algorithm was applied to a 3D simulation of a virtual drone. The main objective of the simulator is to learn how to pilot a virtual drone in a real environment. This requires an accurate map of the environment and locating the drone model based on information from the environment. It is assumed that the system detects collisions between the drone model and the mapped environment and can autonomously determine a route without causing a collision. The main data collection device is a camera in the hands of the operator. This can be a mobile camera or a smartphone. In this case, the model of the drone will be shown on the smartphone screen in the correct position relative to the map of the environment.

For point cloud rendering Open3D framework (<http://www.open3d.org/>) was used, and presented 3D maps are test data from Open3D.

By employing image registration techniques in this manner, the drone simulator can generate a virtual environment that accurately represents the real-world scenes captured by the RGB-D camera.



Fig. 1. AR image with drone and 3D environment map.

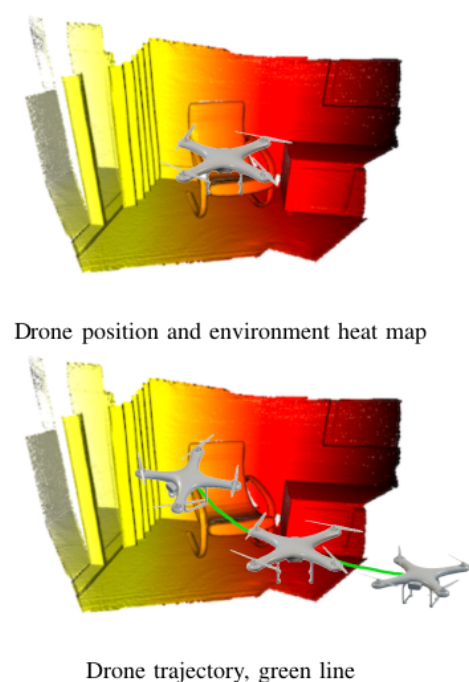


Fig. 2. Drone movement trajectory.

D. Experimental results

1) *Reconstructing the Scene*: Figure 1 illustrates the process of reconstructing a scene using a set of frames. Both images present a point cloud of environment and virtual drone location.

2) *Application for 3D virtual drone simulator*: The proposed method was implemented to develop an AR system that allows maneuvering 3D virtual drone. Image 2 presents a point cloud map of the environment using heat map colors. The top image presents a virtual 3D drone in the start location. On the bottom row there is presented drone position at keyframe locations and trajectory as the green solid line.

IV. CONCLUSION

The article introduces an approach to address image registration in a sequence of multiple frames. The method proposed in the article utilizes data from an RGB-D camera that moves within a static environment. It is important to note that the article does not address the loop closure problem. This is a subject for future work.

REFERENCES

- [1] K. Konstantoudakis, K. Christaki, D. Tsiakmakis, D. Sainidis, G. Albanis, A. Dimou, and P. Daras, "Drone Control in AR: An Intuitive System for Single-Handed Gesture Control, Drone Tracking, and Contextualized Camera Feed Visualization in Augmented Reality," *Drones* 2022, Vol. 6, Page 43, vol. 6, no. 221, 43, 2 2022. doi: 10.3390/DRONES6020043. [Online]. Available: <https://www.mdpi.com/2504-446X/6/2/43/html><https://www.mdpi.com/2504-446X/6/2/43>
- [2] R. S. Pahwa, J. Lu, N. Jiang, T. T. Ng, and M. N. Do, "Locating 3D Object Proposals: A Depth-Based Online Approach," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, pp. 626–639, 9 2017. doi: 10.1109/TCSVT.2016.2616143. [Online]. Available: <http://arxiv.org/abs/1709.02653><http://dx.doi.org/10.1109/TCSVT.2016.2616143>
- [3] J. N. Yasin, H. Mahboob, M. H. Haghbayan, M. M. Yasin, and I. Plosila, "Cellular Formation Maintenance and Collision Avoidance Using Centroid-Based Point Set Registration in a Swarm of Drones," *Lecture Notes in Networks and Systems*, vol. 294, pp. 391–408, 3 2021. doi: 10.1007/978-3-030-82193-7_26. [Online]. Available: <https://arxiv.org/abs/2103.02480v1>
- [4] X. Wan, Y. Shao, and S. Li, "Planetary UAV localization based on Multi-modal Registration with Pre-existing Digital Terrain Model," 6 2021. [Online]. Available: <https://arxiv.org/abs/2106.12738v1>
- [5] K. Koniarski and A. Mysliński, "Feature Point Cloud Based Registration in Augmented Reality," *Lecture Notes in Networks and Systems*, vol. 364 LNNS, pp. 418–427, 12 2021. doi: 10.1007/978-3-030-92604-5_37. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-92604-5_37
- [6] A. W. Fitzgibbon, "Robust registration of 2D and 3D point sets," *Image and Vision Computing*, vol. 21, no. 13–14, pp. 1145–1153, 12 2003. doi: 10.1016/J.IMAVIS.2003.09.004
- [7] K. Zhang, H. Chen, H. Wu, X. Zhao, and C. Zhou, "Point cloud registration method for maize plants based on conical surface fitting—ICP," *Scientific Reports* 2022 12:1, vol. 12, no. 1, pp. 1–15, 4 2022. doi: 10.1038/s41598-022-10921-6. [Online]. Available: <https://www.nature.com/articles/s41598-022-10921-6>
- [8] K. Koniarski, "Augmented reality using optical flow," *Proceedings of the 2015 Federated Conference on Computer Science and Information Systems, FedCSIS 2015*, pp. 841–847, 10 2015. doi: 10.15439/2015F202. [Online]. Available: <http://fedcsis.org/proceedings/2015/drpf/202.html>
- [9] J. Motejlek and A. Alpay, "Taxonomy of Virtual and Augmented Reality Applications in Education," *IEEE Transactions on Learning Technologies*, vol. 14, no. 17, pp. 415–429, 12 2021. doi: 10.1109/TLT.2021.3092964. [Online]. Available: <http://arxiv.org/abs/2112.04619><http://dx.doi.org/10.1109/TLT.2021.3092964>
- [10] J. Mo, M. J. Islam, and J. Sattar, "Fast Direct Stereo Visual

- SLAM," *IEEE Robotics and Automation Letters*, 7, no. 2, pp. 778–785, 4 2022. doi: 10.1109/LRA.2021.3133860. [Online]. Available: <https://arxiv.org/abs/2112.01890v1>
- [11] Mahesh and M. V. Subramanyam, "Automatic feature based image registration using SIFT algorithm," in *2012 3rd International Conference on Computing, Communication and Networking Technologies, ICCCNT 2012*, 2012. doi: 10.1109/ICCCNT.2012.6396024
 - [12] A. Fontes and J. E. B. Maia, "Visual Odometry for RGB-D Cameras," 3 2022. doi: 10.48550/arxiv.2203.15119. [Online]. Available: <https://arxiv.org/abs/2203.15119v1>
 - [13] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *IEEE International Conference on Intelligent Robots and Systems*, 2012. doi: 10.1109/IROS.2012.6385773. ISBN 9781467317375. ISSN 21530858 pp. 573–580.
 - [14] J. Park, Q. Y. Zhou, and V. Koltun, "Colored Point Cloud Registration Revisited," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2017-October, pp. 143–152, 12 2017. doi: 10.1109/ICCV.2017.25
 - [15] R. A. Newcombe, R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon, "KinectFusion: Real-time dense surface mapping and tracking," in *2011 10th IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2011*, 2011. doi: 10.1109/ISMAR.2011.6092378. ISBN 9781457721830 pp. 127–136. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.221.100>
 - [16] E. Marchand, H. Uchiyama, and F. Spindler, "Pose Estimation for Augmented Reality: A Hands-On Survey," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 12, pp. 2633–2651, 12 2016. doi: 10.1109/TVCG.2015.2513408
 - [17] S. Akpinar and F. N. Alpaslan, "Optical flow-based representation for video action detection," *Emerging Trends in Image Processing, Computer Vision and Pattern Recognition*, pp. 331–351, 1 2015. doi: 10.1016/B978-0-12-802045-6.00021-1
 - [18] B. D. Lucas and T. Kanade, "Iterative Image Registration Technique With an Application to Stereo Vision," vol. 2, 1981, pp. 674–679. [Online]. Available: https://www.researchgate.net/publication/215458777_An_Iterative_Image_Registration_Technique_with_an_Application_to_Stereo_Vision_IJCAI
 - [19] B. K. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, no. 1-3, pp. 185–203, 8 1981. doi: 10.1016/0004-3702(81)90024-2
 - [20] A. Bruhn, J. Weickert, and C. Schnörr, "Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods," *International Journal of Computer Vision*, vol. 61, no. 3, pp. 1–21, 2 2005. doi: 10.1023/B:VISI.0000045324.43199.43
 - [21] J. Rambach, A. Pagani, M. Schneider, O. Artemenko, and D. Stricker, "6DoF Object Tracking based on 3D Scans for Augmented Reality with Live Support," *Computers*, vol. 7, no. 1, p. 6, 1 2018. doi: 10.3390/computers7010006. [Online]. Available: <http://www.mdpi.com/2073-431X/7/1/6>
 - [22] J. Engel, J. Sturm, and D. Cremers, "Semi-dense visual odometry for a monocular camera," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013. doi: 10.1109/ICCV.2013.183. ISBN 9781479928392. ISSN 1550-5499 pp. 1449–1456.

FedCSIS_2023_paper_2821.pdf

ORIGINALITY REPORT

50%

SIMILARITY INDEX

PRIMARY SOURCES

1	annals-csis.org Internet	1196 words — 25%
2	link.springer.com Internet	202 words — 4%
3	"Advances in Systems Engineering", Springer Science and Business Media LLC, 2022 Crossref	153 words — 3%
4	vision.in.tum.de Internet	97 words — 2%
5	vladlen.info Internet	74 words — 2%
6	docplayer.net Internet	68 words — 1%
7	www.mdpi.com Internet	43 words — 1%
8	vcl.itl.gr Internet	40 words — 1%
9	arxiv.org Internet	39 words — 1%

10	jahid.ece.ufl.edu Internet	31 words — 1%
11	sol.sbc.org.br Internet	31 words — 1%
12	www.utupub.fi Internet	31 words — 1%
13	Fitzgibbon, A.W.. "Robust registration of 2D and 3D point sets", Image and Vision Computing, 20031201 Crossref	28 words — 1%
14	Kamak Ebadi, Kyle Coble, Dima Kogan, Deegan Atha, Russell Schwartz, Curtis Padgett, Joshua Vander Hook. "Toward Autonomous Localization of Planetary Robotic Explorers by Relying on Semantic Mapping", 2022 IEEE Aerospace Conference (AERO), 2022 Crossref	27 words — 1%
15	www.arxiv-vanity.com Internet	26 words — 1%
16	archive.org Internet	23 words — < 1%
17	e-space.mmu.ac.uk Internet	22 words — < 1%
18	Naimin Xu, Guoxiang Sun, Yuhao Bai, Xinzhu Zhou, Jiaqi Cai, Yinfeng Huang. "Global Reconstruction Method of Maize Population at Seedling Stage Based on Kinect Sensor", Agriculture, 2023 Crossref	20 words — < 1%
19	"Computer Vision – ECCV 2016", Springer Science and Business Media LLC, 2016	18 words — < 1%

-
- 20 nozdr.ru
Internet 18 words — < 1%
-
- 21 biblio.cinvestav.mx
Internet 15 words — < 1%
-
- 22 Christian Kerl, Jurgen Sturm, Daniel Cremers.
"Robust odometry estimation for RGB-D
cameras", 2013 IEEE International Conference on Robotics and
Automation, 2013
Crossref 14 words — < 1%
-
- 23 kids.frontiersin.org
Internet 13 words — < 1%
-
- 24 Konstantinos Konstantoudakis, Kyriaki Christaki,
Dimitrios Tsiakmakis, Dimitrios Sainidis et al.
"Drone Control in AR: An Intuitive System for Single-Handed
Gesture Control, Drone Tracking, and Contextualized Camera
Feed Visualization in Augmented Reality", Drones, 2022
Crossref 12 words — < 1%
-
- 25 Samet Akpınar, Ferda Nur Alpaslan. "Optical flow-
based representation for video action detection",
Elsevier BV, 2015
Crossref 12 words — < 1%
-
- 26 dspace.mit.edu
Internet 12 words — < 1%
-
- 27 kth.diva-portal.org
Internet 12 words — < 1%
-
- 28 www.cs.cmu.edu
Internet 12 words — < 1%

29	ebin.pub Internet	10 words — < 1%
30	pure.bond.edu.au Internet	9 words — < 1%
31	thesai.org Internet	9 words — < 1%
32	Anguo Chen, Jinlei Zhuang, Xiuqin Han. "An Improved ICP Algorithm for 3D Point Cloud Registration", 2022 3rd International Conference on Pattern Recognition and Machine Learning (PRML), 2022 Crossref	8 words — < 1%
33	dokumen.pub Internet	8 words — < 1%
34	escholarship.org Internet	8 words — < 1%
35	espace.curtin.edu.au Internet	8 words — < 1%
36	Michael J. Landau, Peter A. Beling, Michael D. DeVore. "Efficacy of statistical model-based pose estimation of rigid objects with corresponding CAD models using commodity depth sensors", IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society, 2014 Crossref	6 words — < 1%
37	www.ncbi.nlm.nih.gov Internet	6 words — < 1%

EXCLUDE QUOTES OFF
EXCLUDE BIBLIOGRAPHY OFF

EXCLUDE SOURCES OFF
EXCLUDE MATCHES OFF