JOURNAL OF UNIVERSAL SCIENCE RESEARCH

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF 2024 = 5.073/Volume-2, Issue-3

The Optical Flow Method and Graham's Algorithm Implementation Features for Searching for the Object Contour in the Mobile Robot's Workspace

Amer Abu-Jassar [1], Vladyslav Yevsieiev [2], Svitlana Maksymova [2]

[1] Faculty of Information Technology, Department of Computer Science, Ajloun National University, Ajloun, Jordan

[2] Department of Computer-Integrated Technologies, Automation and Robotics, Kharkiv National University of Radio Electronics, Ukraine

Abstract:

This article examines the optical flow method and graham algorithm implementation features for searching for the object contour in the mobile robot's workspace. The mathematical models of both methods were discussed in detail and then implemented in a Python program using the PyCharm development environment. As part of the study, a number of experiments were carried out, the purpose of which was to evaluate the performance of the optical flow method and the Graham algorithm for extracting the contour of an object. The research results presented in the article highlight the effectiveness of the optical flow method and the Graham algorithm in real-time conditions.

Key words: Industry 5.0, Computer Vision Systems, Mobile Robots

Introduction

Currently, with the development of the Industry 5.0 concept and the introduction of modern technologies into production processes, the task of increasing the efficiency and safety of mobile robots in workspaces is becoming increasingly urgent [1]-[10]. One of the key elements to ensure their operation is Computer Vision Systems [11]-[24], which allow robots to obtain information about the environment and make appropriate decisions based on this information. In this context, tracking objects in the work area of a mobile robot becomes especially important in order to prevent collisions and ensure the safety of the work process [25]-[28]. And here various approaches for analysis can be used [29]-[32].

To create computer vision systems, many algorithms are used, among which are Canny, Sobel, Graham, etc. Computer vision algorithms such as Graham's algorithm play a significant role in this process as they are capable of detecting and tracking objects in real time. The use of Graham's algorithm to track objects in the

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF 2024 = 5.073/Volume-2, Issue-3

work area of a mobile robot is an effective approach to solve this problem. However, due to the peculiarities of the working environment and the technical characteristics of mobile robots, there is a need to develop specialized implementations of the Graham algorithm that take into account data about a specific work area, type of objects, lighting conditions and other factors.

Such research has important practical implications for Industry 5.0, as it helps improve the efficiency of mobile robots, reduce the likelihood of accidents and ensure personnel safety. At the same time, the development and optimization of implementations of the Graham algorithm requires taking into account the specifics of the work area, which makes research in this area relevant and promising for the further development of industry and the introduction of modern technologies.

Related works

Many scientists are considering using Graham's algorithm to detect the outlines of objects in real time. However, there are some shortcomings in this algorithm; hence researchers have proposed various ways to solve these problems. Let us look at some works on this topic.

First of all, let us consider the work [33]. There is presented an efficient algorithm, based on the idea of Graham’s convex hull algorithm, for finding the connected orthogonal convex hull of a finite planar point set. Authors in [34] analyzed standard Graham’s algorithm and proposed a fast filtering technique that reduces the computational cost for computing a convex hull for a large set of points.

Graham’s algorithm is also analyzed in [35] there is suggested the developed algorithm for making up a convex hull based on a binary tree. The new algorithm quite quickly exceeds the running time of the Graham algorithm with a large number of points in the convex hull. This is due to the best complexity of the algorithm. The paper [36] presents an optimization technique that reduces the computational cost for building the Convex Hull from a set of points. Experimental results show that for a normal distribution of points in two-dimensional space, the filtering approach in conjunction with the Graham scan is up to 10x faster than the qhull library, and between 1.7x to 10x faster than the Convex Hull methods available in the CGAL library. There is a study [37] that considers machine learning methods for automatic classification problems using computational geometry. The results of experimental

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF 2024 = 5.073/Volume-2, Issue-3

studies on the real medical diagnostics problem are presented. An efficiency comparison of the proposed classifier and other types of classifiers, both based on convex hull analysis and not, has shown the high efficiency of the proposed method for estimating proximity based on linear programming.

Scientists in [38] propose an improved Graham scan convex hull algorithm using the convex hull region shrinkage algorithm and the sample selection decision algorithm. In the study [39] the method of identifying and calculating the wave-shaped defects on the working side of strip steel based on the convex hull detection algorithm is discussed. Here the classic Graham's Scan algorithm is used to detect the processed strip images. And the improved Graham's Scan algorithm was used to detect the wave-shaped defects of strip steel again. The list of works on this topic goes on. Next, we will present our vision of how to use Graham's algorithm to highlight the contours of objects in the workspace of a mobile robot.

The optical flow method and Graham's algorithm implementation features for searching for the object contour in the mobile robot's workspace

Optical flow is a vector field that describes the movement of objects between two consecutive video frames. It allows you to track the movement of pixels from one frame to another and detect moving objects. The operating principle of the optical flow method can be described as follows: for each pixel (x, y) in the image at a moment in time t, we can find its new position $(x+\Delta x, y+\Delta y)$ at a moment in time $t+\Delta t$ using the following expression:

$$I(x+\Delta x, y+\Delta y, t+\Delta t) = I(x, y, t) \quad (1)$$

$I(x, y, t)$ – pixel intensity at a moment in time t.

Graham's algorithm is used to find the outline of an object in an image. It is based on constructing a convex hull for a given set of points. The principle of operation of Graham's algorithm can be described in next way: at the first stage, the starting point of the contour (P_0), which is the leftmost and lowest point or the point with the smallest coordinates (x_0, y_0), is selected. The next step is to sort $P_1, P_2, ..., P_n$

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF 2024 = 5.073/Volume-2, Issue-3

by polar angle relative to the starting point. Let $P_0 = (x_0, y_0)$ be the coordinates of the starting point, and let $P_i = (x_i, y_i)$ be the coordinates of the remaining points. Then the polar angle θ_i for each point P_i can be calculated by the next expression:

$$\theta_i = arctg(\frac{y_i - y_0}{x_i - x_0}) \quad (2)$$

θ_i – polar angle of point P_i relative to the starting point P_0.

After calculating the polar angles for all other points, we can sort them by ascending or descending angles. This way the points will be ordered by polar angle relative to the starting point P_0.

After this, we construct a convex hull using a stack to store the vertices of the shell, for this purpose on the basis of sorted all points ($P_1, P_2, ..., P_n$) in the order of their polar angle (θ_i) relative to the starting point (P_0). Next, starting from the leftmost and bottom point (P_0), we go through the sorted points. Add the first two points to the stack. For each next point, we check whether it is “left” or “right” relative to the previous two points in the stack. This can be done using the three-point P_i, P_j, P_k rotation matrix determinant , where P_i and P_j are the last two points added to the stack and P_k is the current point. If the determinant (O_p) is positive, then the point is P_k “to the left” of the segment P_i, P_j and it is added to the stack. If the determinant is negative or zero, then the point P_k is to the "right" of the segment P_i, P_j, and the last two points are popped from the stack while the condition is true, after which the point P_k is added to the stack. Mathematically, this can be represented through the determinant of the three-point rotation matrix $P_i(x_i, y_i), P_j(x_j, y_j), P_k(x_k, y_k)$.

$$O_p = (x_j - x_i)(y_k - y_i) - (y_j - y_i)(x_k - x_i) \quad (3)$$

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF 2024 = 5.073/Volume-2, Issue-3

An example of constructing a convex hull using a stack to store the vertices of the hull is shown in Figure 1.

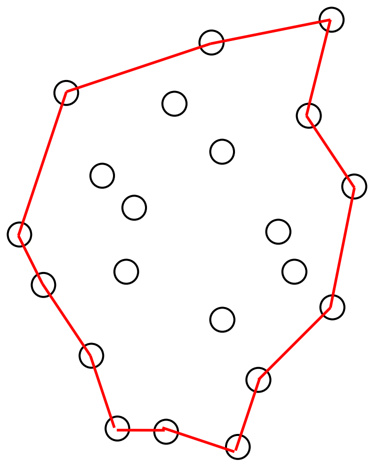


Figure 1: Example of constructing a convex hull using a stack to store the vertices of the hull

At the last step, we remove non-convex corners, if they are present, by checking for left (or right) rotation for every three consecutive points P_i, P_j, P_k. Graham's algorithm can effectively extract the outline of an object in an image, which can be especially useful for mobile robots in work areas where it is necessary to detect obstacles and other objects for safe navigation. Using optical flow to pre-filter moving objects can improve the performance of Graham's algorithm and the accuracy of object contour detection.

Software implementation and experiments

To check the correctness of the reasoning, we will develop a program in Python in the PyCharm 2022.2.3 (Professional Edition) development environment. Let us give an example of software implementation of the above described mathematical expressions.

```
# Read the first frame to initialize the optical flow method
    ret, prev_frame = cap.read()
    if not ret:
```

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF 2024 = 5.073/Volume-2, Issue-3

```
        print("Error: Unable to get first frame.")
        return
```

This piece of code is responsible for initializing the optical flow method by reading the first frame from the video stream and handling possible errors during this operation.

```
    # Using optical flow to get moving points
        flow = cv2.calcOpticalFlowFarneback(prev_gray, gray, None, 0.5, 3, 15, 3, 5, 1.2, 0)
```

This piece of code calculates the optical flow between two consecutive frames of a video stream and returns the result in the flow variable. This result is a set of optical flow vectors that show the direction of movement of points from the previous frame to the current one.

```
    # Application of Graham's algorithm to detect the object contour with visualization of points
        contour_image = apply_graham_scan(frame.copy(), points)
```

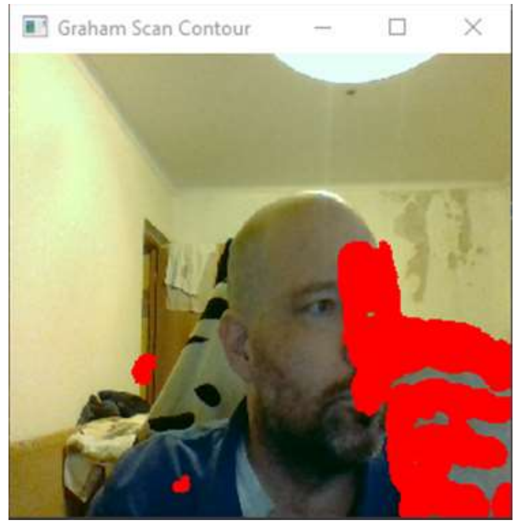
This piece of code applies Graham's algorithm to detect the object contour using pre-computed moving points and stores the result in the contour_image variable.

The following hardware was used for research: CPU Intel(R) Core(TM) i5-9300H CPU @ 2.40GHz, RAM 16 Gb, GPU NVideo GeForce GTX 1660Ti (Ram 8Gb), Web-camera HD WebCam, OS Windows 10 Pro (Version 22H2).

A program for implementing the optical flow method and Graham's algorithm for searching for the contour of an object in the working area of a mobile robot from a camera was developed in the PyCharm 2022.2.3 (Professional Edition) environment in Python. The results of the program are presented in Figure 2.

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF 2024 = 5.073/Volume-2, Issue-3

a)



b)

c)

d)

a),c) – Original Video; b), d) – Graham Scan Control

Figure 2: Results of the program for the optical flow method and Graham's algorithm implementing for searching for the object contour in the mobile robot's workspace

The results obtained (Fig. 2.a, b) of the optical flow method and Graham's algorithm for searching the contour of an object in the working area of a mobile robot showed a frame processing speed of 10.32 frames per second, which confirms the effectiveness of the algorithm in real time. At the same time, the results are presented in Figure 2.c, d; the contour construction speed was 8.17 frames per second,

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF 2024 = 5.073/Volume-2, Issue-3

indicating a potential decrease in performance when contouring large objects to the frame.

Conclusion

As a result of the research carried out on the optical flow method and Graham's algorithm implementation for searching for the object contour in the mobile robot's workspace, significant conclusions were obtained. The mathematical models of both methods were carefully studied and implemented in a Python program using the PyCharm development environment. During the experiments, the speed of frame processing was assessed, as well as the speed of constructing the contour of an object.

Analysis of the results showed that the optical flow method demonstrates fairly high performance, processing frames at a speed of 10.32 frames per second. This indicates its effectiveness and applicability for solving real-time problems in the mobile robot's workspace. The experimental results are presented in Figures 2.c and 2.d, which allows you to visually evaluate the performance of the optical flow method. However, the speed of constructing the contour using the Graham's algorithm was 8.17 frames per second. This indicates a potential performance hit when contouring large objects to the frame. Such results may have an impact on the actual application of Graham's algorithm in real-time conditions where it is necessary to process frames at high speed. In general, the conducted studies confirm the effectiveness of the optical flow method and the Graham algorithm for processing the video stream in the mobile robot's workspace.

References:

1. Abu-Jassar, A. T., Al-Sharo, Y. M., Lyashenko, V., & Sotnik, S. (2021). Some Features of Classifiers Implementation for Object Recognition in Specialized Computer systems. TEM Journal: Technology, Education, Management, Informatics, 10(4), 1645-1654.

2. Matarneh, R., Maksymova, S., Deineko, Z., & Lyashenko, V. (2017). Building robot voice control training methodology using artificial neural net. International Journal of Civil Engineering and Technology, 8(10), 523-532.

3. Baker, J. H., Laariedh, F., Ahmad, M. A., Lyashenko, V., Sotnik, S., & Mustafa, S. K. (2021). Some interesting features of semantic model in Robotic

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF 2024 = 5.073/Volume-2, Issue-3

Science. SSRG International Journal of Engineering Trends and Technology, 69(7), 38-44.

4. Al-Sharo, Y. M., Abu-Jassar, A. T., Sotnik, S., & Lyashenko, V. (2021). Neural Networks As A Tool For Pattern Recognition of Fasteners. International Journal of Engineering Trends and Technology, 69(10), 151-160.

5. Sotnik, S., Mustafa, S. K., Ahmad, M. A., Lyashenko, V., & Zeleniy, O. (2020). Some features of route planning as the basis in a mobile robot. International Journal of Emerging Trends in Engineering Research, 8(5), 2074-2079.

6. Sotnik, S., & Lyashenko, V. (2022). Prospects for Introduction of Robotics in Service. Prospects, 6(5), 4-9.

7. Ahmad, M. A., Sinelnikova, T., Lyashenko, V., & Mustafa, S. K. (2020). Features of the construction and control of the navigation system of a mobile robot. International Journal of Emerging Trends in Engineering Research, 8(4), 1445-1449.

8. Al-Sharo, Y. M., Abu-Jassar, A. T., Sotnik, S., & Lyashenko, V. (2023). Generalized Procedure for Determining the Collision-Free Trajectory for a Robotic Arm. Tikrit Journal of Engineering Sciences, 30(2), 142-151.

9. Lyashenko, V., & Sotnik, S. (2020). Analysis of Basic Principles for Sensor System Design Process Mobile Robots. Journal La Multiapp, 1(4), 1-6.

10. Matarneh, R., & et al.. (2019). Development of an Information Model for Industrial Robots Actuators. IOSR Journal of Mechanical and Civil Engineering (IOSR-JMCE), 16(1), 61-67.

11. Yevsieiev, V., & et al. (2024). Active Contours Method Implementation for Objects Selection in the Mobile Robot's Workspace. Journal of Universal Science Research, 2(2), 135-145.

12. Akopov, M., & et al. (2023). Choosing a Camera for 3D Mapping. Journal of Universal Science Research, 1(11), 28-38.

13. Yevsieiev, V., Abu-Jassar, A., & Maksymova, S. (2024). Object Recognition And Tracking Method in the Mobile Robot's Workspace in Real Time. Technical Science Research In Uzbekistan,2(2), 115-124.

14. Maksymova, S., & Velet, A. (2022) Development of an Automated System of Terminal Access to Production Equipment Using Computer Vision. In Manufacturing & Mechatronic Systems 2022: Proceedings of VIst International Conference, Kharkiv, P. 22-23.

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF 2024 = 5.073/Volume-2, Issue-3

15. Yevsieiev, V., & et al. (2024). Using Contouring Algorithms to Select Objects in the Robots' Workspace. Technical Science Research In Uzbekistan, 2(2), 32-42.

16. Kobylin, O., & Lyashenko, V. (2014). Comparison of standard image edge detection techniques and of method based on wavelet transform. International Journal, 2(8), 572-580.

17. Rabotiahov, A., Kobylin, O., Dudar, Z., & Lyashenko, V. (2018, February). Bionic image segmentation of cytology samples method. In 2018 14th International Conference on Advanced Trends in Radioelecrtronics, Telecommunications and Computer Engineering (TCSET) (pp. 665-670). IEEE.

18. Гиренко, А. В., Ляшенко, В. В., Машталир, В. П., & Путятин, Е. П. (1996). Методы корреляционного обнаружения объектов. Харьков: АО "БизнесИнформ, 112.

19. Matarneh, R., Tvoroshenko, I., & Lyashenko, V. (2019). Improving Fuzzy Network Models For the Analysis of Dynamic Interacting Processes in the State Space. International Journal of Recent Technology and Engineering, 8(4), 1687-1693.

20. Lyubchenko, V., & et al.. (2016). Digital image processing techniques for detection and diagnosis of fish diseases. International Journal of Advanced Research in Computer Science and Software Engineering, 6(7), 79-83.

21. Lyashenko, V. V., Matarneh, R., Kobylin, O., & Putyatin, Y. P. (2016). Contour Detection and Allocation for Cytological Images Using Wavelet Analysis Methodology. International Journal, 4(1), 85-94.

22. Orobinskyi, P., Deineko, Z., & Lyashenko, V. (2020). Comparative Characteristics of Filtration Methods in the Processing of Medical Images. American Journal of Engineering Research, 9(4), 20-25.

23. Lyashenko, V., Matarneh, R., & Kobylin, O. (2016). Contrast modification as a tool to study the structure of blood components. Journal of Environmental Science, Computer Science and Engineering & Technology, 5(3), 150-160.

24. Boboyorov Sardor Uchqun o‘g‘li, Lyubchenko Valentin, & Lyashenko Vyacheslav. (2023). Image Processing Techniques as a Tool for the Analysis of Liver Diseases. Journal of Universal Science Research, 1(8), 223–233.

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF 2024 = 5.073/Volume-2, Issue-3

25. Abu-Jassar, A., & et al. (2023). Obstacle Avoidance Sensors: A Brief Overview. Multidisciplinary Journal of Science and Technology, 3(5), 4-10.

26. Borysov, H., & et al. (2023). Parameters for Mobile Robot Kinematic Model Development Determination. Multidisciplinary Journal of Science and Technology, 3(4), 85-91.

27. Basiuk, V., & et al. (2023). Mobile Robot Position Determining Using Odometry Method. Multidisciplinary Journal of Science and Technology, 3(3), 227-234.

28. Igor, N., & et al. (2023). Using Mecanum Wheels for Radio Shuttle. Multidisciplinary Journal of Science and Technology, 3(3), 182-187.

29. Dadkhah, M., Lyashenko, V. V., Deineko, Z. V., Shamshirband, S., & Jazi, M. D. (2019). Methodology of wavelet analysis in research of dynamics of phishing attacks. International Journal of Advanced Intelligence Paradigms, 12(3-4), 220-238.

30. Sotnik, S., Matarneh, R., & Lyashenko, V. (2017). System model tooling for injection molding. International Journal of Mechanical Engineering and Technology, 8(9), 378-390.

31. Babker, A., & Lyashenko, V. (2018). Identification of megaloblastic anemia cells through the use of image processing techniques. Int J Clin Biomed Res, 4, 1-5.

32. Khan, A., Ahmad, M., Joshi, S., & Lyashenko, V. (2016). Synthesis of alumina fibre by annealing method using coir fibre. American Chemical Science Journal, 15(2), 1-7.

33. An, P. T., & et al. (2021). A modified Graham's convex hull algorithm for finding the connected orthogonal convex hull of a finite planar point set. Applied Mathematics and Computation, 397, 125889.

34. Alshamrani, R., & et al. (2020). A preprocessing technique for fast convex hull computation. Procedia Computer Science, 170, 317-324.

35. Aung, N. L., & et al. (2020). Developed algorithm for making up convex hull based on binary tree. In 2020 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM), IEEE, 1-6.

36. Ferrada, H., & et al. (2020). A filtering technique for fast Convex Hull construction in R2. Journal of Computational and Applied Mathematics, 364, 112298.

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF 2024 = 5.073/Volume-2, Issue-3

37. Nemirko, A. P., & Dulá, J. H. (2021). Machine learning algorithm based on convex hull analysis. Procedia Computer Science, 186, 381-386.

38. Xue, H., & et al. (2021). Cross Product and Partitioned Filtering-Based Graham Convex Hull for Buoy Drifting Area Demarcating. Scientific Programming, 2021, 1-9.

39. Ma, K., & et al. (2020). A strip steel wave-edge detection algorithm based on convex hull detection. In Eleventh International Conference on Graphics and Image Processing (ICGIP 2019), SPIE, 11373, 167-172.