Scholarly Publisher
RS Global Sp. z O.O.
ISNI: 0000 0004 8495 2390

Dolna 17, Warsaw, Poland 00-773
Tel: +48 226 0 227 03
Email: editorial_office@rsglobal.pl

JOURNAL	World Science
p-ISSN	2413-1032
e-ISSN	2414-6404
PUBLISHER	RS Global Sp. z O.O., Poland

ARTICLE TITLE	VISION SYSTEMS OF UAVs AND SIMULATION OF OBJECT RECOGNİTİON IN MATLAB ENVIRONMENT
AUTHOR(S)	T.A. Safarov.
ARTICLE INFO	T.A. Safarov. (2023) Vision Systems of Uavs and Simulation of Object Recognition in MATLAB Environment. *World Science*. 1(79). doi: 10.31435/rsglobal_ws/30032023/7967
DOI	https://doi.org/10.31435/rsglobal_ws/30032023/7967
RECEIVED	15 February 2023
ACCEPTED	29 March 2023
PUBLISHED	30 March 2023
LICENSE	This work is licensed under a **Creative Commons Attribution 4.0 International License.**

VISION SYSTEMS OF UAVs AND SIMULATION OF OBJECT RECOGNİTİON IN MATLAB ENVIRONMENT

T.A. Safarov
Azerbaijan State Oil and Industry University, Azerbaijan

DOI: https://doi.org/10.31435/rsglobal_ws/30032023/7967

ARTICLE INFO

Received: 15 February 2023
Accepted: 29 March 2023
Published: 30 March 2023

KEYWORDS

Object Recognition, Vision Systems, MATLAB Realization, Unmanned Aerial Vehicle, Remote Control.

ABSTRACT

In recent years, we are aware of the increase in the production of unmanned aerial vehicles (UAVs), their application in various fields and their future potential. An unmanned aerial vehicle (UAV) is an aircraft without a crew. UAVs can have variable characteristics: they can be remotely piloted or fully automatic. In these articles, we will consider UAVs, which we consider as dynamic objects, to receive the coordinates of ground objects according to their flight modes, to simulate them, to observe them, to draw their images, and to classify the objects in the image according to various characteristics in real time. Among the researches of recent times, the researches related to the detection of moving or stationary objects from real-time images taken from unmanned vehicles are of special importance. Because continuous surveillance cannot be done with old video surveillance systems. Integration with systems that automatically detect and recognize objects is used for continuous monitoring.

Citation: T.A. Safarov. (2023) Vision Systems of Uavs and Simulation of Object Recognition in MATLAB Environment. *World Science.* 1(79). doi: 10.31435/rsglobal_ws/30032023/7967

Introduction.

Before the vision systems of UAVs, object recognition systems, I would like to give information about vision systems in general, computer vision. When we say computer vision, we understand both theoretically and technologically the possibility of such a machine that can detect objects and divide them into classes. From an engineering perspective, it seeks to understand and automate tasks that the human vision system can perform. Computer vision uses the technological and theoretical foundations of creating artificial intelligence products. Examples of computer vision applications include: Creating a 2D or 3D model of the surrounding environment, mapping, creating video imaging systems, etc. Object recognition is done by assigning the attributes of the objects in the image to the class with the closest attributes to it. In all these processes, the models representing the classes should be prepared in advance. Models registered in the database must be entered together with all its attributes [4].

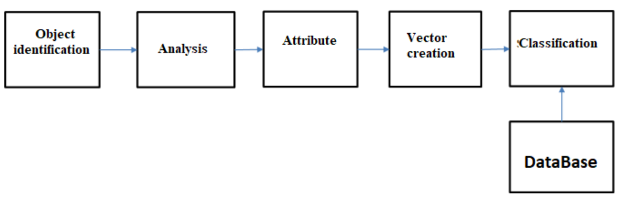

Figure 1. Object recognition algorithm

Materials and Methods.

The most optimal software for application, simulation and research of computer vision can be considered the Image Processing Toolbox application package in MATLAB. With this software package, it is possible to model images, process them, perform a number of operations on images. The main elementary functions here are: imcrop - framing of images; imresize - resizing; imrotate - image rotation. When implementing UAVs in the MATLAB environment, a description of the main 3 coordinate states that must be considered to keep them in equilibrium is given. These coordinate states are: D -Drone state; S – Stabilizing state; W – initial state. Designing UAVs with MATLAB and Simulink package saves time and reduces our costs. UAVs make it possible to receive the coordinates of real objects in images taken from the earth's surface, observe and calculate area units. BWLABEL, IMFEATURE functions in the MATLAB system will be used for the study of image object recognition methods. With these functions, search and calculation of object features is performed [1].

BWLABEL function - Syntax:

L = bwlabel(BW)

L = bwlabel(BW,conn)

[L,n] = bwlabel(___)

L – Label matrix. Also called label matrix of intersection points. It is in the form of a matrix of non-negative integers. Pixels labeled "0" are background. Pixels labeled "1" form an object. Pixels labeled "2" form the second object, and so on. BW – binary (2D) image; n – the number of objects in the image; conn is the pixel connection.

IMFEATURE function. - Syntax:

stats = imfeature(L, dimensions)

stats = imfeature(L, dimensions, n)

Dimensions are given in comma-separated matrix form. Within the IMFUTURE function, there are ready-made values as in the table shown in Figure 2, which we can use in the functions we write to perform the necessary calculations.

'Area'	'Image'	'EulerNumber'
'Centroid'	'FilledImage'	'Extrema'
'BoundingBox'	'FilledArea'	'EquivDiameter'
'MajorAxisLength'	'ConvexHull'	'Solidity'
'MinorAxisLength'	'ConvexImage'	'Extent'
'Eccentricity'	'ConvexArea'	'PixelList'
'Orientation'		

Figure 2. Ready-made values of the IMFUTURE function

Let's consider some of them, which will be needed to recognize objects in the image processed by the UAV system and to search for them according to their various characteristics: Area', 'Centroid', and 'BoundingBox' are calculated if dimensions are not known [2].

Area – represents the total number of pixels.

Centroid – the coordinates of the coordinate origin.

BoundingBox – to find the coordinates of the rectangle.

MATLAB implementation of searching for objects in images captured by the UAV system according to their features.

Let's assume that there are 4 surface objects of different geometric sizes in the image processed by the system, and we want to filter only the geometric objects of the required size among the common objects in the image by means of a pre-written program. First, we introduce the image captured by the UAV to the program with the imread function.

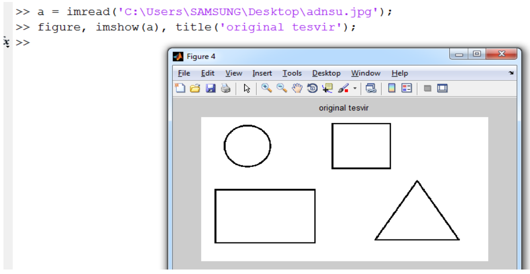

Figure 3. Introducing the image to the program

Before filtering the image, some operations are performed on the image. So, the BWLABEL function mentioned above is used for filtering. Since this function is for binary images, the processed image is generated in binary form.

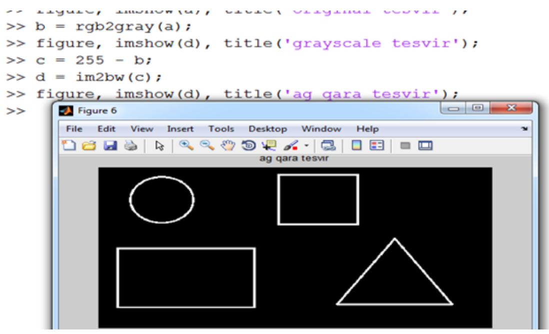

Figure 4. Image conversion to binary format

In the image generated to binary form, the boundaries of the objects will be shown in white. But since we want to calculate the areas of the objects in addition to the coordinates, we will have to find how many pixels the objects are made of in the scaled image, so we fill the interior of the object with the help of the imfill function.

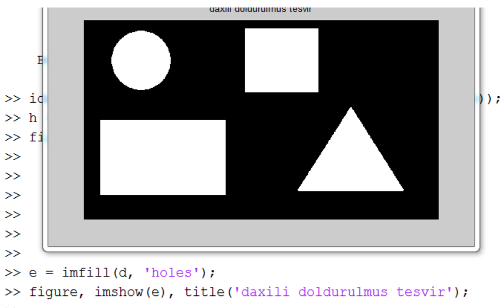

Figure 5. Filling the interior of the objects in the image

Results and discussions.

We can then calculate the values of the objects' fields with the help of the regionprops function. At the same time, considering each object separately, we can find both the value of its area and its coordinates using the ready-made value elements of the IMFUTURE function in Figure 2.

```
>> g(1)

ans =

           Area: 49997
    BoundingBox: [38.5000 230.5000 289 173]

>> g(2)

ans =

           Area: 14511
    BoundingBox: [64.5000 25.5000 136 136]

>> g(3)

ans =

           Area: 24990
    BoundingBox: [372.5000 19.5000 170 147]
```

Figure 6. Finding coordinates and areas of objects in the image

```
>> f = bwlabel(e);
>> g = regionprops(f, 'Area', 'BoundingBox');
>> area_values = [g.Area]

area_values =

       49997       14511       24990       24186

>> |
```

Figure 7. Finding areas of objects in the image

Filtering can then be done by specifying certain conditions and using the find command. In the example we looked at, let's find objects with a field value greater than 20000 and less than 30000.

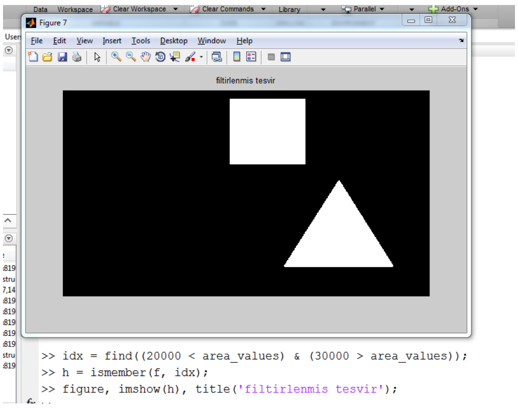

Figure 8. Searching for the features of the objects in the processed image

Thus, with this algorithmic sequence, it is possible to perform recognition, classification and filtering of ground objects on images taken by UAV.

Conclusions.

In this article, the vision systems of UAVs were investigated in Matlab software, and it was emphasized that the above-mentioned algorithm can perform object recognition from images obtained from UAVs. Thus, taking into account that UAVs are mostly used in military equipment, they can be considered an excellent tool for easier detection of enemy ground objects. At the same time, its application in surveillance drones, in the non-military field, is considered favorable.

Special packages of Matlab Software were used during the research.

REFERENCES

1. Shuang Qiao, ”Research of Improving the Accuracy of License Plate Character Seg-mentation”, Fifth International Conference on Frontier of Computer Science Technology, 2010.
2. Baudoin, Y. and Colon, E., Humanitarian Demining and Robotics State-of-the-art, Specifications, and On-going Research Activities, Proc. of the Third Int. Conf. on Climbing and Walking Robots (CLAWAR2000), October 2-4, 2000, Madrid, pp.869-877.
3. Hofmann, Martin. ”Support Vector Machine-Kernel and The Kernel Trick”, Bamberg University, 2006.
4. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Cham, Switzerland, 2016; pp. 21–37.
5. Uchida, H. and Nonami, K., Biological Walking of a Mine Detecting Six-Legged Robot Using Rhythm Generation and Fuzzy Control, Proc. of 5th Int. Conf. on Climbing and Walking Robots, (CLAWAR2002), 2002, Paris, pp.741-748.
6. Raibert, M.H. and Craig, J.J., Hybrid Position/Force Control of Manipulators, ASME Journal of Dynamic Systems, Measurement, and Control, Vol.103, No.2 (1981), pp.126-133.
7. Belmonte, L.M.; Morales, R.; Fernández-Caballero, A. Computer Vision in Autonomous Unmanned Aerial Vehicles—A Systematic Mapping Study. Appl. Sci. 2019, 9, 3196.
8. Wu, X.; Li, W.; Hong, D.; Tao, R.; Du, Q. Deep Learning for Unmanned Aerial Vehicle-Based Object Detection and Tracking: A survey. IEEE Geosci. Remote Sens. Mag. 2022, 10, 91–124.