

# ALGORITHM FOR PROCESSING DIGITAL TAGS FOR DELIVERY OF CARGO IN LIMITED TERRITORY BY UNMANNED AERIAL VEHICLES

**Alefirenko V.**

*«Belarus State University of Informatics and Radioelectronics», Ph.D, associate Professor, Minsk*

**Nguyen Trong Phuong**

*«Belarus State University of Informatics and Radioelectronics», master student, Minsk*

## АЛГОРИТМ ОБРАБОТКИ ЦИФРОВЫХ МЕТОК ДЛЯ ДОСТАВКИ ГРУЗОВ НА ОГРАНИЧЕННОЙ ТЕРРИТОРИИ БЕСПИЛОТНЫМИ ЛЕТАТЕЛЬНЫМИ АППАРАТАМИ

**Алефиренко В.М.**

*«Белорусский государственный университет информатики и радиоэлектроники», канд. техн. наук, доцент,  
г. Минск*

**Нгуен Чонг Фьюнг**

*«Белорусский государственный университет информатики и радиоэлектроники», магистрант,  
г. Минск*

### **Abstract**

An algorithm is proposed for processing images of digital **tags** used for delivery of small-size cargoes by unmanned aerial vehicles to a destination in limited territories. The algorithm includes image reading, grayscale conversion, stabilization, filtering, segmentation, contour extraction, projective conversion, and marker recognition operation.

### **Аннотация**

Предложен алгоритм обработки изображений цифровых меток, используемых для доставки малогабаритных грузов беспилотными летательными аппаратами к месту назначения на ограниченных территориях. Алгоритм включает операции считывания изображения, преобразования в оттенки серого, стабилизации, фильтрации, сегментации, выделения контуров, проективного преобразования и распознавания маркера.

**Keywords:** processing algorithm, digital tags, unmanned aerial vehicles, cargo delivery, limited territory.

**Ключевые слова:** алгоритм обработки, цифровые метки, беспилотные летательные аппараты, доставка грузов, ограниченная территория.

Для доставки грузов на небольших ограниченных территориях могут использоваться беспилотные летательные аппараты (БЛА) или дроны [1, 2]. К таким территориям относятся помещения для производства продукции (сборочные цеха, теплицы), территории заводов, фабрик, больниц, и другие подобные объекты или территории. Решение такой задачи особенно актуально в тех случаях, когда по каким-либо причинам невозможен доступ или ограничен контакт с территорией (зона затопления, зона карантина и др.). В такие зоны небольшие грузы с медикаментами и предметами первой необходимости могут доставляться только с помощью БЛА. Для этих целей наиболее подходящими являются многороторные БЛА, оснащенные системой инерциальной навигации и датчиком изображения [3]. Такие БЛА могут выполнять поставленные задачи с более высокой точностью не привязываясь полностью к системам глобальной навигации, а используя специальные метки, размещенные в местах доставки грузов.

Для обеспечения способности работать самостоятельно БЛА оснащаются одноплатными компьютерами. Из-за ограничения компьютерных ресурсов алгоритм обработки изображений меток должен обеспечивать баланс между скоростью выполнения операций распознавания и скоростью полета БЛА. Для достижения этой цели, объект распознавания должен быть простым и легко обнаруживаемым. Для этого в большей степени подходят объекты, представляющие собой цифровые метки. QR-код является одним из известных примеров таких меток. Однако QR-код подходит только для считывания на небольших расстояниях, например, с камеры смартфона. Кроме того, он имеет достаточно сложную структуру, так как предназначен для кодирования и хранения большого количества данных. Поэтому сложная структура QR-кода является недостатком для обнаружения места доставки грузов с помощью БЛА.

Для этой цели более подходит маркер ArUco (рис.1), представляющий собой синтетический квадратный маркер, состоящий из широкой черной рамки и внутренней двоичной матрицы, которая определяет его идентификатор [4]. Благодаря простой структуре маркера, процесс его обработки будет достаточно простым и быстрым.

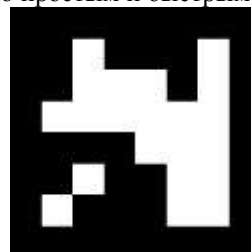


Рис. 1 Пример маркера ArUco размером 6x6

Существуют различные методы обнаружения объектов, из которых наиболее известными являются каскадный классификатор, SVM-классификатор с гистограммой направленных градиентов и методы глубокого обучения [5].

Хотя эти методы достаточно точны, они требуют больших объемов данных, а также больших вычислительных операций, которые не подходят для ограниченных вычислительных ресурсов одноплатного компьютера БЛА. Поэтому для обнаружения выбранного маркера, целесообразно использовать более простой алгоритм.

Разработанный алгоритм обнаружения маркера состоит из следующих основных этапов (рис. 2):

- считывание (захват) изображения;
- стабилизация изображения;
- фильтрация изображения;
- сегментация изображения;
- выделение контуров объектов изображения;
- проективное преобразование контуров;
- распознавание маркера.



Рис. 2 Структурная схема алгоритма обнаружения маркера

**Считывание (захват) изображения.** Для обеспечения высокой скорости обработки, изображение должно иметь подходящий размер. Хорошим вариантом является размер  $640 \times 480$  пиксел

(рис. 3). Преобразование изображения в оттенки серого (рис. 4) позволит уменьшить объем дальнейших вычислений.



Рис. 3 Исходное изображение маркера



Рис. 4 Исходное изображение маркера, преобразованное в оттенки серого

**Стабилизация изображения.** Для снижения влияния вибрации, возникающей при работе БЛА, и получения качественных изображений необходимо использовать цифровую стабилизацию. Для вычисления оптического потока использовался алгоритм Лукаса – Канаде [6]. Алгоритм исходит из трех предположений:

- интенсивность пикселей объекта между двумя последовательными кадрами постоянна;
- движение пикселя между двумя последовательными кадрами очень мало;
- пиксели, принадлежащие одному и тому же объекту, перемещаются одинаково.

Так как совместное уравнение, описывающие первое и второе предположения, включало две неизвестные переменные для любого пикселя, то это уравнение невозможно было решить. В этом случае для решения задачи стабилизации изображения было использовано третье предположение. Тогда движение центрального пикселя может быть легко рассчитано путем объединения соседних пикселей для формирования системы уравнений. Для вычисления движения центрального пикселя использовалась матрица  $5 \times 5$ , включающая соседние пиксели, что позволило создать систему из 25 уравнений, которая решалась методом наименьших квадратов.

**Фильтрация и сегментация изображения.** Для уменьшения влияния шумов, первым шагом обработки является сглаживание изображения, которое осуществляется с помощью фильтра Гаусса [7]. Фильтрация фильтром Гаусса позволяет получить взвешенное среднее значение яркости пикселя

с учетом значений яркости соседних пикселей, вносящих больший «вес» в среднее значение, что и приводит к сглаживанию изображения.

После этого, для выделения областей изображения, которые могут содержать маркер, применялась сегментация изображения. Существуют различные методы сегментации, но более подходящим является метод адаптивной пороговой сегментации [8], который очень эффективен даже в нестабильных условиях освещения и позволяет проводить вычисления с высокой скоростью. При реализации этого метода важным является правильный выбор размера окрестности пикселя, от которого зависит значение порога, используемого для дальнейшей обработки изображения. С одной стороны размер окрестности пикселя должен быть достаточно большим, чтобы покрыть достаточное количество пикселей фона и переднего плана, в противном случае полученное значение порога будет неэффективным для дальнейшей обработки изображения. С другой стороны, если значение размера окрестности пикселя будет слишком большим, то может быть нарушено предположение, на котором базируется метод адаптивной пороговой сегментации – используемые для получения порога области изображения должны иметь приблизительно равномерную яркость. Таким образом, значение размера окрестности пикселя должно выбираться экспериментально и в нашем случае составило  $35 \times 35$  пикселей

Тем не менее, после сегментации на изображении может существовать еще много помех (рис. 5). Поэтому для их удаления применялась морфологическая фильтрация [9]. Для ее реализации использовались операции сжатия и расширения (рис 6).



Рис. 5 Результат адаптивной пороговой сегментации

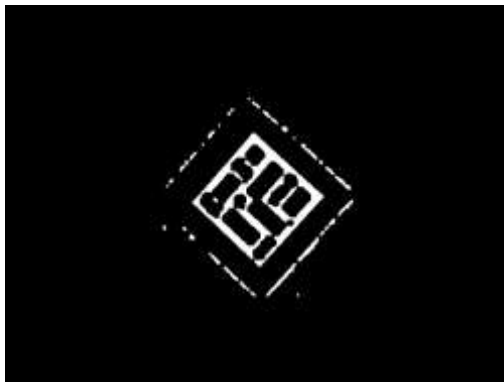


Рис. 6 Результат морфологической фильтрации

Таким образом, последовательное применение операций Гауссовой фильтрации, адаптивной пороговой сегментации и морфологической фильтрации позволяет устранить различные виды шумов и сохранить изображение маркера.

**Выделение контуров объектов изображения.** После удаления помех для выделения объектов, которые формально могут быть приняты за

маркер, используется алгоритм выделения контуров [10]. Поскольку маркер является квадратным, то контуры, которые не соответствуют этому условию, не могут быть маркерами. Подходящие по форме контуры будут извлечены для дальнейшей обработки (рис. 7).

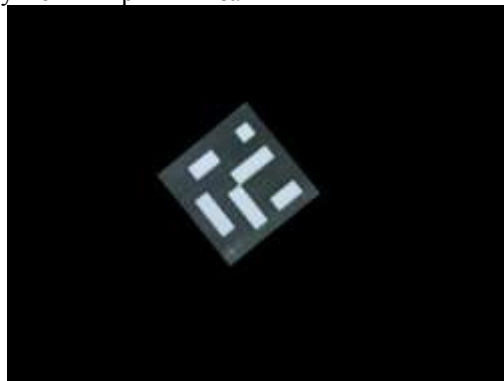


Рис. 7 Результат выделения контура

**Проективное преобразование.** Поскольку изображение объекта в каждом кадре может находиться в произвольном положении из-за угла поворота относительно траектории движения БЛА, то его необходимо преобразовать в заранее определенное положение с помощью проективного преобразования. Матрица проективного преобразования связывает положения точек на плоскости исходного изображения (обычно на плоскости формирователя изображения) с помощью следующих уравнений [11]:

$$\vec{T}_{\text{вых}} \equiv \begin{bmatrix} x_{\text{вых}} \\ y_{\text{вых}} \\ 1 \end{bmatrix} = H \cdot \vec{T}_{\text{исх}}$$

$$\vec{T}_{\text{исх}} \equiv \begin{bmatrix} x_{\text{исх}} \\ y_{\text{исх}} \\ 1 \end{bmatrix} = H^{-1} \cdot \vec{T}_{\text{вых}}$$

где  $\vec{T}_{\text{исх}}$  – исходная точка;

$\vec{T}_{\text{вых}}$  – выходная точка;

$H$  – матрица проективного преобразования.

Результат проективного преобразования показан на рис. 8.

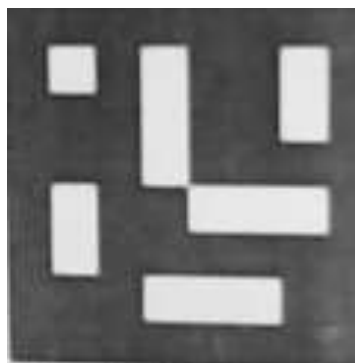


Рис. 8 Результат проективного преобразования

**Распознавание маркера.** Для подготовки выходного изображения проективного преобразования для дальнейшего извлечения структуры мар-

кера была еще раз применена фильтрация фильтром Гаусса и адаптивная пороговая сегментация (рис. 9).

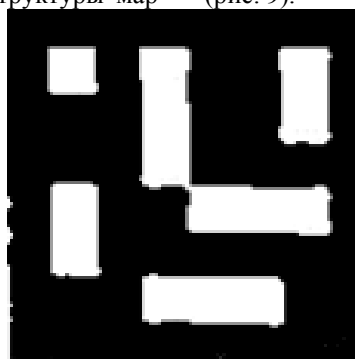


Рис. 9 Результат адаптивной пороговой сегментации

Для повышения точностью сравнения объектов, которые потенциально могут быть маркером, с идеальным маркером, необходимо осуществить извлечение структуры маркера. Для этого был выбран метод извлечения структуры «*Block-Binary-Pixel-Sum Features*» [12], обладающий простотой и высокой точностью. Принцип работы этого метода состоит в следующем. В начале изображение маркера

стандартизируется до определенного размера. После этого изображение делится на непересекающиеся блоки пикселей размером  $M \times N$  (рис. 10). Для каждого из этих блоков вычисляется отношение пикселей переднего плана к общему количеству пикселей в каждом блоке. Эти отношения представляются в виде вектора признаков. Для повышения точности, необходимо повторять эти вычисления с блоками размером  $M_i \times N_j$ .

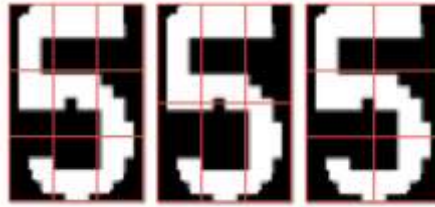


Рис.10 Пример разделения изображения на непересекающиеся блоки пикселей размерами  $3 \times 3$ ,  $3 \times 2$  и  $2 \times 3$

Структурная схема алгоритма метода «*Block-Binary-Pixel-Sum Features*» показана на рис. 11.

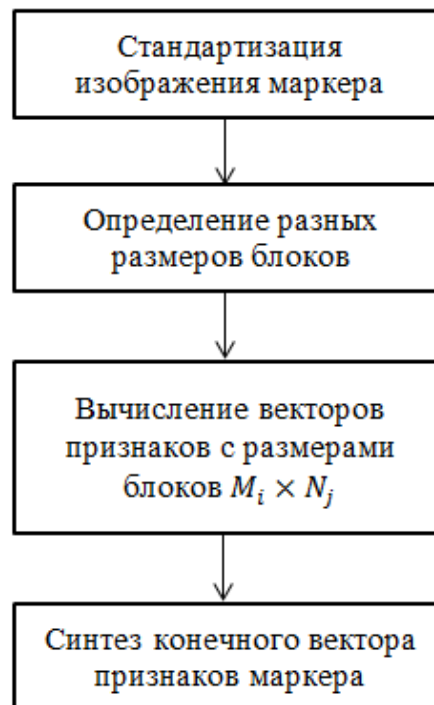


Рис. 11 Структурная схема алгоритма метода «*Block-Binary-Pixel-Sum Features*»

Конечные векторы объектов сравниваются с вектором идеального маркера с помощью косинусного коэффициента [13]. Объект считается маркером, если его косинусный коэффициент меньше

предварительно определенного порогового значения. Результат обнаружения маркера показан на рис. 12.



Рис. 12 Результат распознавания маркера

Таким образом, предложенный, разработанный и проверенный экспериментально алгоритм обработки цифровых меток, включающий такие операции обработки изображений, как преобразование исходного изображения в оттенки серого, различные виды фильтрации, сегментация, выделение контуров, проективное преобразование и распознавание объекта, выполняемые в соответствующей последовательности, позволяет использовать его для обнаружения и распознавания маркеров при доставке малогабаритных грузов беспилотными летательными аппаратами к месту назначения на ограниченных территориях.

#### СПИСОК ЛИТЕРАТУРЫ:

1. Джунипер, А. Дроны: полное практическое руководство / А. Джунипер. – Москва: КоЛибри, 2019. – 160 с.
2. Алефиренко, В.М. Использование беспилотных летательных аппаратов для доставки почты / В.М. Алефиренко, Ф.Ч. Нгуен // Современные средства связи: материалы XXIV Междунар. науч.-техн. конф., Минск, 17–18 октября 2019 г. / БГАС. – Минск, 2019. – С. 95–96.
3. Алефиренко, В.М. Возможности использования беспилотных летательных аппаратов для решения задач по доставке грузов на ограниченной территории / В.М. Алефиренко, Ф.Ч. Нгуен // Технические науки – от теории к практике : сб. публикаций научного журнала «Globus» по материалам XXXXIII Международной науч.-практ. конф., С-П., 19 мая 2019 г. / С-П., 2019. – Вып. 3 (27). – С. 59–69.
4. Sergio Garrido. Detection of ArUco Markers. [Электронный ресурс] – Режим доступа: [https://docs.opencv.org/trunk/d5/dae/tutorial\\_aruco\\_detection.html](https://docs.opencv.org/trunk/d5/dae/tutorial_aruco_detection.html)
5. Object detection. [Электронный ресурс] – Режим доступа: [https://en.wikipedia.org/wiki/Object\\_detection](https://en.wikipedia.org/wiki/Object_detection)
6. Lucas, B. D., T. Kanade. «An iterative image registration technique with an application to stereo vision», Proceedings of the 1981 DARPA Imaging Understanding Workshop (pp. 121–130), 1981.
7. Adrian Kaehler. Learning OpenCV 3 / Adrian Kaehler, Gary Bradski // Gaussian Filter / O'Reilly Media – 2017. – С. 266.
8. Adrian Kaehler. Learning OpenCV 3 / Adrian Kaehler, Gary Bradski // Adaptive Threshold / O'Reilly Media – 2017. – С. 259-261.
9. Adrian Kaehler. Learning OpenCV 3 / Adrian Kaehler, Gary Bradski // Image Morphology / O'Reilly Media – 2017. – С. 275-290.
10. Adrian Kaehler. Learning OpenCV 3 / Adrian Kaehler, Gary Bradski // Contours / O'Reilly Media – 2017. – С. 407-441.
11. Adrian Kaehler. Learning OpenCV 3 / Adrian Kaehler, Gary Bradski // Video Models and Calibration / O'Reilly Media – 2017. – С. 663.
12. Yen-Ching Chang. License Plate Character Recognition Using Block-BinaryPixel-Sum Features / International Conference on Computer, Networks and Communication Engineering (ICCNCE 2013) – С. 111-113.
13. Cosine Distance [Электронный ресурс] – Режим доступа: <https://reference.wolfram.com/language/ref/CosineDistance.html>