

# Интеллектуальные системы управления, анализ данных

© 2023 г. **Р.М. ШАКИРЗЯНОВ**, канд. техн. наук (RMShakirzyanov@kai.ru),  
**М.П. ШЛЕЙМОВИЧ**, канд. техн. наук (MPShleymovich@kai.ru)  
(Казанский национальный исследовательский технический  
университет им. А.Н. Туполева),  
**С.В. НОВИКОВА**, д-р техн. наук (SVNovikova@kai.ru)  
(Казанский национальный исследовательский технический  
университет им. А.Н. Туполева,  
Мордовский государственный университет им. Н.П. Огарева)

## МЕТОД АВТОМАТИЧЕСКОГО ПОЗИЦИОНИРОВАНИЯ БЕСПИЛОТНЫХ АППАРАТОВ НА ОСНОВЕ РАСПОЗНАВАНИЯ СИГНАЛЬНЫХ РАДИАЛЬНО-СИММЕТРИЧНЫХ МАРКЕРОВ ПОДВОДНЫХ ЦЕЛЕЙ

Описывается метод автоматического распознавания целевых точек траекторий беспилотных аппаратов, перемещающихся под водой, таких как автономные подводные аппараты и летающие подводные аппараты самолетоподобных конструкций. В качестве терминальной точки управления рассматривается координата центра объекта, обладающего свойствами радиальной симметрии. Предложен метод построения многомасштабной весовой модели изображения на основе разработанного преобразования быстрой радиальной симметрии и метода Хафа, что обеспечивает устойчивость к шумам и высокую скорость вычисления координат искомой точки. Для случая, когда объект интереса задан контуром определенного цвета, предложена модель на основе хроматической и весовой составляющих. В качестве примера детектирования приведен алгоритм обнаружения базовой подводной станции со световыми маркерами в виде сигнального люминесцирующего кольца.

*Ключевые слова:* автоматические транспортные системы, беспилотный подводный аппарат, беспилотный летательный подводный аппарат, компьютерное зрение, обработка изображений, обнаружение объектов на изображениях, метод Хафа, быстрое преобразование радиальной симметрии, весовая модель изображения, многомасштабная весовая модель изображения.

**DOI:** 10.31857/S0005231023070061, **EDN:** FDNZDT

### 1. Введение

При построении систем автоматического управления беспилотными аппаратами необходимым условием является задание конечного положения аппа-

рата в пространстве. В большинстве работ по автоматическому управлению беспилотными аппаратами конечное положение задается самим исследователем в виде набора конкретных координат [1, 2]. Однако в реальных условиях возникают задачи, когда аппарат должен самостоятельно обнаруживать объект интереса, положение которого будет задавать конечное положение аппарата при управлении. Транспортные системы, где аппарат самостоятельно определяет конечную точку своего движения и рассчитывает траекторию перемещения к ней, принято называть интеллектуальными транспортными системами [3]. Метод определения терминальной точки, как и способ построения оптимальной траектории, при этом не обязательно должен включать в себя элементы искусственного интеллекта (ИИ), такие как модели машинного обучения, нейронные сети и т.п. Однако на практике в случае, если метод расчета не предполагает использования ИИ, систему чаще определяют как «автоматическую» или «автоматизированную».

Наиболее распространенными беспилотными аппаратами в настоящее время являются беспилотные летательные аппараты (БПЛА) [4]. Основным способом позиционирования для БПЛА являются спутниковые системы GPS и ГЛОНАСС [5, 6]. Для беспилотных аппаратов, передвигающихся под водой, так называемых автономных обитаемых подводных аппаратов (АНПА) [7], позиционирование в большинстве случаев осуществляется при помощи ультразвуковых сигналов [8]. Основным недостатком такого подхода является обязательное наличие специальной навигационной аппаратуры на борту аппарата, а также необходимость ответного сигнала от целевой точки позиционирования, что делает аппарат уязвимым для внешнего обнаружения.

Системы позиционирования на основе компьютерного зрения лишены указанных недостатков. Они способны обеспечивать высокую точность распознавания и при этом не выдают своего присутствия внешними сигналами. Кроме того, для таких современных беспилотных систем, как беспилотные летательные подводные аппараты (БПЛПА) [9], визуальное позиционирование при подводном задании цели является единственно возможным. БПЛПА способны преодолевать расстояния сначала по воздуху, а затем погружаться в воду и выполнять маневры под водой. Из-за особенностей распространения и рассеивания сигналов на стыке двух сред воздух-вода позиционирование при помощи ультразвуковых сигналов является труднореализуемым.

Для БПЛА, как и для беспилотных наземных транспортных средств, задача обнаружения объектов на основе компьютерного зрения хорошо изучена [10–12]. Известны работы по визуальному позиционированию точки посадки БПЛА [13], определения положения летательного аппарата на основе предварительно сделанных спутниковых снимков местности и нейросетевых распознавателей [14] и др. Общим недостатком предлагаемых подходов является необходимость дополнительной информации для позиционирования, а также высокие требования к вычислительной мощности оборудования на борту БПЛА.

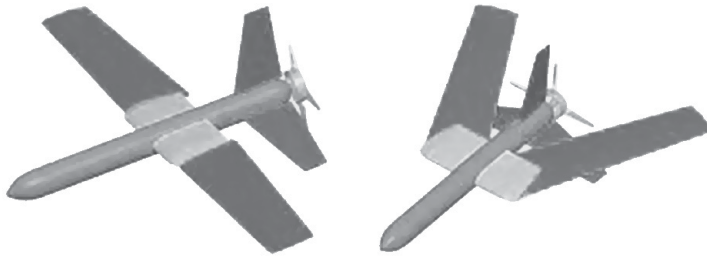


Рис. 1. Конструкция ныряющего самолета со складным крылом (источник — см. [16]).

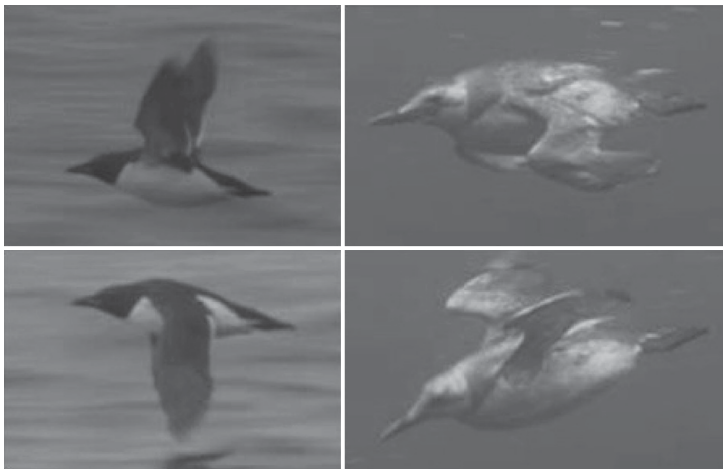


Рис. 2. Механика перемещения кайры в воздухе и под водой, примененная для имитации динамики БПЛПА (источник — см. [18]).

Для беспилотных аппаратов, передвигающихся под водой, алгоритмы компьютерного зрения должны удовлетворять дополнительным требованиям из-за особенностей распространения света в воде и дополнительных помех из-за мутности естественных водоемов [15]. В случае БПЛПА большое значение также приобретают скорость работы алгоритма из-за относительно высоких скоростей перемещения аппарата и необходимое для детектирования объекта количество вычислений, так как летающие подводные аппараты не способны нести на себе вычислители большой мощности и их энергопотребление высоко относительно АНПА. Поэтому задача разработки алгоритма визуального позиционирования для цели, заданной под водой, с минимальными энерго- и вычислительными затратами особенно актуальна именно для гибридных БПЛПА.

Случай, когда конечное целевое положение аппарата находится под водой, более характерен для летательных подводных аппаратов типа «ныряющий самолет» [16], чем, например, для плавающих квадрокоптеров [17]. Самолете-

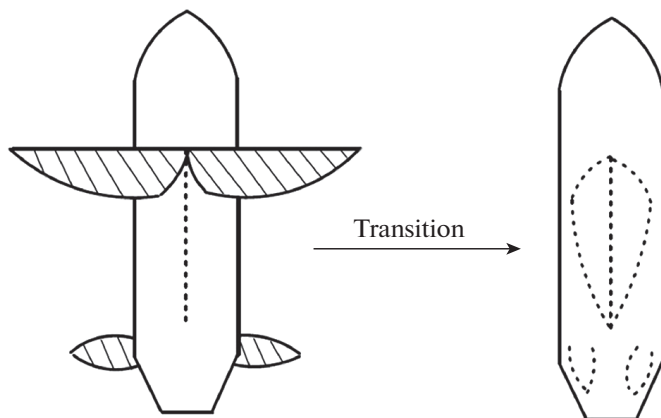


Рис. 3. Конфигурация оперения БПЛА в различных средах (источник — см. [19]).

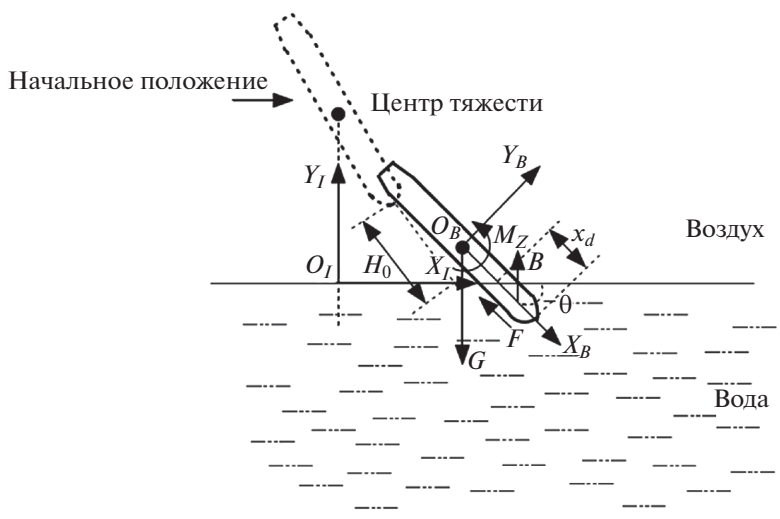


Рис. 4. Силы, влияющие на самолетоподобный БПЛА при нырянии (источник — см. [19]).

топоподобные конструкции в основном приспособлены только к прямому переходу «воздух-вода» и испытывают значительные трудности либо вообще не способны осуществлять обратный переход «вода-воздух». В частности, в [16] предложена конструкция ныряющего самолета с отклоняющимися на 65 градусов при нырянии крыльями (см. рис. 1).

Конструкция включает в себя крыло изменяемой стреловидности и двигатель на сжатом углекислом газе. В [18] предложена конструкция ныряющего БПЛА, имитирующего полет и ныряние кайры (рис. 2).

В [19] описана система автоматического управления ныряющего самолета со складывающимся оперением (рис. 3).

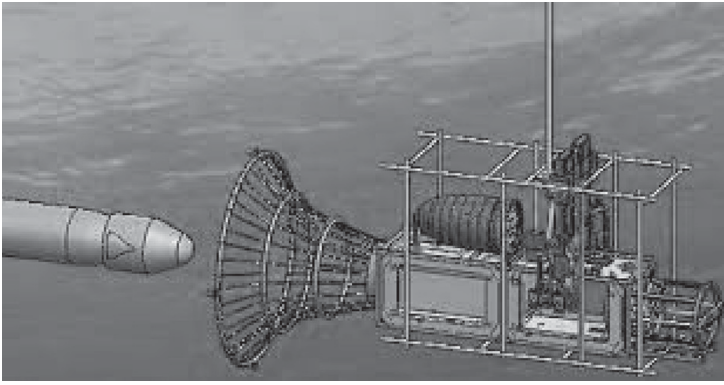


Рис. 5. Пример стыковки автономного необитаемого подводного аппарата с док-станцией (источник — см. [20]).

В [19] строится управление для достижения аппаратом конечной точки под водой (рис. 4).

Критерием оптимальности управления выбрана точность достижения заданной позиции  $(x_{target}, y_{target})$  в конечный момент времени  $t_f$ .

$$(1) \quad J = \sqrt{(x_h(t_f) - x_{target})^2 + (y_h(t_f) - y_{target})^2}.$$

Управляющими параметрами выступают начальный угол тангажа  $\theta_0$  и начальная скорость вдоль оси  $OX$   $v_{x_0}$ , расчет траектории производится без возможности маневрирования. В качестве целевых координат  $(x_{target}, y_{target})$  авторы определяют конкретную точку под водой. Однако в случае адаптивного управления указанные координаты аппарат может вычислять самостоятельно в автоматическом режиме на основе машинного зрения. Таким образом, возникает задача распознавания координат некоего целевого объекта под водой, математический центр тяжести которого будет являться целевой точкой управления  $(x_{target}, y_{target})$ . Данная задача является одним из возможных характерных примеров задачи распознавания в целях решения задачи оптимального управления.

Для подводных беспилотных аппаратов задание объектов интереса под водой является единственно возможным. Таким объектом интереса, в частности, может быть специальная стыковочная док-станция [20], используемая для подзарядки батарей АНПА или/и передачи данных (см. рис. 5).

Особенности позиционирования АНПА и БПЛПА в данном случае будут одни и те же, так как в обоих случаях цель является подводной. То обстоятельство, что цель находится именно под водой, является важным для разработки метода детектирования целевого объекта, так как замутненность воды, наличие взвеси, искажений изображения вследствие преломления света накладывают на метод детектирования жесткие требования по устойчивости к внешним шумам. Оптические методы позиционирования для подводных

целей описаны, например, в [21, 22]. При этом могут использоваться как одна [23], так и две [24] камеры. Основным ограничением предлагаемых методов является их высокое потребление вычислительных и энергетических ресурсов, так как практически все они предназначены для АНПА. Это делает предложенные методы слабоприменимыми для БПЛА. Так, в [20] в качестве математической модели распознавания используется сверточная нейронная сеть CNN. Эффективность подобных сетей высока, если они специальным образом обучаются для довольно узкого заданного класса объектов. Например, показано, что CNN способны распознавать рукописный текст с точностью до 99% [25]. Однако при распознавании произвольных реальных объектов их точность падает до 70% и ниже. Кроме того, CNN требуют мощных вычислительных устройств для обучения, которое может продолжаться от нескольких часов до нескольких суток [26]. Современные модификации могут давать либо более высокую точность, либо более высокую скорость, но пока не способны совмещать в себе оба этих качества. Так, в [27] описаны так называемые двухпроходные алгоритмы типа FASTER RCNN, достигающие точности 73,2% (датасет VOC07<sup>1</sup>). Время обучения сети не описано, однако сеть обладает относительно невысокой скоростью распознавания в 17 изображений в секунду. Однопроходные сети типа YOLO достигают скорости распознавания в 155 изображений в секунду, однако их точность при этом всего лишь 52,7% на том же датасете VOC07.

Таким образом, задача разработки алгоритма обнаружения объектов под водой с минимальными вычислительными и временными затратами является актуальной, особенно для БПЛА.

## 2. Постановка задачи

Рассмотрим задачу распознавания подводного объекта в автоматическом режиме с борта беспилотного аппарата на примере обнаружения базовой док-станции со световыми маркерами в виде сигнального люминесцирующего кольца. Центр кольца задает целевую точку траектории оптимального управления без маневрирования  $(x_{target}, y_{target})$  по примеру задачи (1). В общем случае задача может решаться одновременно для нескольких возможных целей  $N_{target}$ .

Дано:

- а) цветное изображение  $I = (I_{ij})_{N_x \times N_y}$  в цветовой модели RGB;
- б) множество радиусов окружностей  $\mathbf{N}$ , которые требуется найти;
- в) множество допустимых цветов  $\overline{\mathbf{C}}$ , в оттенках которого могут окрашиваться искомые окружности;
- г) допустимое рассогласование между положениями центров окружностей  $\Delta$ .

---

<sup>1</sup> VOC07 — набор данных свободного доступа, содержащий классифицированные изображения (Visual Object Classes-VOC) для 20 классов, редакция 2007 г.

Найти:

д) множество центров окружностей с их радиусами, определяющих терминальную точку оптимального управления  $\tilde{A} = \{(x_{target}^k, y_{target}^k), r^k\}$ ,  $k = \overline{1, N_{target}}$ .

Задачей настоящего исследования является разработка вычислительного алгоритма, принимающего на вход позиции а)– г) и генерирующего на выходе позицию д). Дополнительным требованием к алгоритму является устойчивость к внешним шумам, а также минимизация вычислительных затрат и времени вычислений.

В случае, когда реальный размер окружности (например, стыковочного кольца) известен, в качестве искомым радиусов изображения следует выбрать ряд уменьшающихся значений от реального до минимально возможного радиуса. Впоследствии по найденному значению радиуса на изображении будет возможно определить расстояние до объекта с применением дополнительных измерительных приборов.

Метод должен обеспечивать распознавание при слабых деформациях окружности в эллипс с соотношением малой и большой полуоси не менее 0,95, что соответствует смещению прямой проекции обзора не более чем на  $20^\circ$  [28].

### **3. Метод решения задачи распознавания цветных объектов круглой формы**

Важной задачей, решаемой средствами компьютерного зрения, является обработка изображений с борта беспилотного аппарата с целью обнаружения и локализации заданных объектов — световых маркеров. Данные объекты характеризуются формой и цветом. Поэтому для их обнаружения и распознавания применяются методы анализа цветовых признаков и признаков формы.

Популярные в последнее время алгоритмы на основе глубоких нейронных сетей для данной постановки задачи будут являться избыточными, так как распознаванию подлежит простая геометрическая фигура, а не сложный многоцветный объект. Кроме того, актуальными проблемами при использовании нейронных сетей остаются сложность в обобщении и длительность этапа обучения. Поэтому классические методы, в которых производится анализ изображения, остаются актуальными.

Многие применяющиеся на практике подходы к анализу формы объектов на изображениях базируются на методах анализа контуров [29]. Световые маркеры для стыковки подводных аппаратов являются цветными объектами, обладающими радиально-симметричной формой. Для их обнаружения и локализации можно применить методы, базирующиеся на анализе свойств радиальной симметрии. В основу предложенного метода легли совместное использование известного метода Хафа [30] и преобразование быстрой радиальной симметрии с использованием построения пирамиды Гаусса.



### 3.1. Преобразование быстрой радиальной симметрии

В данной работе предлагается эффективный подход к решению рассматриваемой задачи, основанный на применении быстрого преобразования радиальной симметрии FRST (Fast Radial Symmetry Transform). Преобразование FRST позволяет построить весовую модель изображения, с помощью анализа которой можно эффективно локализовать центр радиально-симметричного объекта [31, 32].

Для получения весовой модели изображения необходимо выполнить следующие шаги:

- 1) преобразовать изображение в полутоновый вид;
- 2) вычислить градиенты функции яркости на изображении;
- 3) вычислить значения элементов матриц весов;
- 4) нормировать значения элементов матриц весов;
- 5) вычислить значения элементов матриц обобщенных весов;
- 6) выполнить низкочастотную фильтрацию матриц обобщенных весов;
- 7) вычислить значения элементов матрицы усредненных весов.

На первом шаге осуществляется преобразование изображения в полутоновый вид, в котором каждому пикселю сопоставляется значение яркости. В простейшем случае преобразование заключается в вычислении средних значений интенсивностей красной, зеленой и синей цветовых составляющих. Таким образом, в результате выполнения первого шага изображение представляется в виде функции яркости, аргументами которой являются значения координат пикселей.

На втором шаге выполняется вычисление градиентов функции яркости на изображении с помощью следующих операторов:

$$(2) \quad g_x(\mathbf{p}) := \hat{\mathbf{I}}(i+1, j) - \hat{\mathbf{I}}(i, j);$$

$$(3) \quad g_y(\mathbf{p}) := \hat{\mathbf{I}}(i, j+1) - \hat{\mathbf{I}}(i, j);$$

$$(4) \quad |\mathbf{g}(\mathbf{p})| = \sqrt{g_x^2(\mathbf{p}) + g_y^2(\mathbf{p})},$$

где  $\mathbf{g}(\mathbf{p}) = (g_x(\mathbf{p}), g_y(\mathbf{p}))$  — градиент в пикселе  $\mathbf{p}$  с координатами  $(i, j)$ ;  $g_x(\mathbf{p})$ ,  $g_y(\mathbf{p})$  — составляющие градиента для горизонтального и вертикального направлений в пикселе  $\mathbf{p}$  соответственно;  $\hat{\mathbf{I}}(i, j)$ ,  $\hat{\mathbf{I}}(i+1, j)$ ,  $\hat{\mathbf{I}}(i, j+1)$  — значения яркости в пикселях полутонового изображения с координатами  $(i, j)$ ,  $(i+1, j)$  и  $(i, j+1)$  соответственно. В выражениях (2)–(4) знаком  $:=$  обозначен оператор присваивания, который будет использоваться и в дальнейшем.

На третьем шаге вычисляются значения элементов матриц весов. Для этого применяется следующая процедура:

- 1) определяется множество целочисленных значений  $\mathbf{N}$ , где  $\mathbf{N}$  — множество радиусов объектов, которые требуется найти;



2) для каждого значения  $n$  из множества  $\mathbf{N}$  формируются начальные значения элементов двух матриц весов:

$$(5) \quad \mathbf{M}^n(\mathbf{p}) := 0;$$

$$(6) \quad \mathbf{O}^n(\mathbf{p}) := 0,$$

где  $p$  — вектор координат  $(i, j)$ .

3) для всех элементов матриц весов, определяемых значениями  $n$  из множества  $\mathbf{N}$ , применяются следующие операторы:

$$(7) \quad \mathbf{p}_+ := \mathbf{p} + \left\lceil \frac{g_x(\mathbf{p})}{|\mathbf{g}(\mathbf{p})|} n \right\rceil;$$

$$(8) \quad \mathbf{p}_- := \mathbf{p} - \left\lceil \frac{g_x(\mathbf{p})}{|\mathbf{g}(\mathbf{p})|} n \right\rceil;$$

$$(9) \quad \mathbf{M}^n(\mathbf{p}_+) := \mathbf{M}^n(\mathbf{p}_+) + |\mathbf{g}(\mathbf{p})|;$$

$$(10) \quad \mathbf{M}^n(\mathbf{p}_-) := \mathbf{M}^n(\mathbf{p}_-) - |\mathbf{g}(\mathbf{p})|;$$

$$(11) \quad \mathbf{O}^n(\mathbf{p}_+) := \mathbf{O}^n(\mathbf{p}_+) + 1;$$

$$(12) \quad \mathbf{O}^n(\mathbf{p}_-) := \mathbf{O}^n(\mathbf{p}_-) - 1,$$

где  $\lceil \cdot \rceil$  — оператор округления вещественного числа до ближайшего не меньшего целого значения.

На четвертом шаге элементы матриц весов нормируются:

$$(13) \quad \mathbf{M}^n(\mathbf{p}) := \frac{|\mathbf{M}^n(\mathbf{p})|}{\max_q \{|\mathbf{M}^n(\mathbf{q})|\}};$$

$$(14) \quad \mathbf{O}^n(\mathbf{p}) := \frac{|\mathbf{O}^n(\mathbf{p})|}{\max_q \{|\mathbf{O}^n(\mathbf{q})|\}}.$$

На пятом шаге вычисляются значения элементов матриц обобщенных весов:

$$(15) \quad \mathbf{F}^n(\mathbf{p}) := (\mathbf{O}^n(\mathbf{p}))^\alpha \mathbf{M}^n(\mathbf{p}),$$

где  $\alpha$  — параметр радиальной жесткости.

На шестом шаге выполняется низкочастотная фильтрация матриц обобщенных весов. При этом, как правило, используется гауссов фильтр нижних частот. Фильтрация выражается с помощью оператора свертки

$$(16) \quad \mathbf{S}^n := \mathbf{F}^n * \mathbf{G}^n,$$

где  $\mathbf{G}^n$  — матрица коэффициентов гауссова фильтра нижних частот, определенная для значения  $n$  из множества  $\mathbf{N}$ .

На последнем, седьмом, шаге вычисляются значения элементов матрицы усредненных весов:

$$(17) \quad \mathbf{S}(\mathbf{p}) := \frac{1}{|\mathbf{N}|} \sum_{n \in \mathbf{N}} \mathbf{S}^n(\mathbf{p}).$$

Полученная в результате преобразования FRST матрица усредненных весов представляет собой весовую модель изображения, анализ которой позволяет определить параметры формы объектов, например координаты центра и радиус объектов круглой формы.

### 3.2. Весовая модель изображения

При построении весовой модели изображения с использованием быстрого преобразования радиальной симметрии необходимо выполнить относительно большое количество вычислений, которое определяется мощностью множества целочисленных значений  $\mathbf{N}$ . Элементы указанного множества применяются для анализа существенных изменений яркости на соответствующих расстояниях от текущего пикселя, т.е. имеют смысл радиусов для радиально-симметричных объектов с центрами в пикселях изображения.

Объекты на изображениях реального мира имеют многомасштабную природу, поскольку обладают различными размерами. Для рассматриваемой задачи это означает использование множества  $\mathbf{N}$  с большим количеством элементов, которые определяются диапазоном изменения радиусов объектов интереса. Поэтому целесообразно проводить анализ исходного изображения, представленного в многомасштабном виде. Такой многомасштабный анализ позволяет сократить количество вычислений и повысить точность обнаружения объектов на изображении.

Для построения многомасштабной весовой модели изображения необходимо выполнить следующие шаги:

- 1) преобразовать изображение в полутоновый вид;
- 2) построить пирамиду Гаусса;
- 3) вычислить значения элементов матриц усредненных весов для всех уровней пирамиды Гаусса;
- 4) привести матрицы усредненных весов к размерам исходного изображения;
- 5) вычислить значения элементов интегральной матрицы усредненных весов.

На первом шаге осуществляется преобразование изображения в полутоновый вид, в котором каждому пикселю сопоставляется значение яркости.

На втором шаге строится пирамида Гаусса [33]. Она представляет собой множество изображений  $\mathbf{P} = \{\mathbf{P}_l \mid l = \overline{0, L-1}\}$ , где  $L$  — число уровней в пирамиде. В качестве нулевого уровня пирамиды  $\mathbf{P}_0$  рассматривается исходное изображение в полутоновом виде  $\hat{\mathbf{I}}$ . Остальные уровни пирамиды форми-

руются следующим образом:

$$(18) \quad \mathbf{P}_l := \mathbf{G} * (2 \downarrow [\mathbf{P}_{l-1}]),$$

где  $\mathbf{G}$  — матрица коэффициентов гауссова фильтра нижних частот;  $2 \downarrow [\cdot]$  — оператор понижения дискретизации изображения, например, посредством удаления каждого второго пикселя в строке и столбце. Элемент пирамиды Гаусса уровня  $l$  будет иметь размеры в четыре раза меньше по сравнению с размерами элемента уровня  $l - 1$ .

На третьем шаге с помощью операторов (5)–(17) формируются весовые модели изображений всех уровней пирамиды Гаусса, т.е. строится множество матриц усредненных весов  $\{\mathbf{S}^{(0)}, \mathbf{S}^{(1)}, \dots, \mathbf{S}^{(L-1)}\}$ .

На четвертом шаге множество  $\{\mathbf{S}^{(0)}, \mathbf{S}^{(1)}, \dots, \mathbf{S}^{(L-1)}\}$  преобразуется в множество  $\{\mathbf{R}^{(0)}, \mathbf{R}^{(1)}, \dots, \mathbf{R}^{(L-1)}\}$ , каждый элемент которого  $\mathbf{R}^{(l)}$  представляет собой результат приведения матрицы  $\mathbf{S}^{(l)}$  к размерам исходного изображения, т.е. к размерам  $\mathbf{S}^{(0)}$ . Указанное преобразование выполняется следующим образом:

$$(19) \quad \mathbf{R}^{(l)} := \mathbf{G} * (2 \uparrow [\mathbf{S}^{(l)}]),$$

где  $\mathbf{G}$  — матрица коэффициентов гауссова фильтра нижних частот;  $2 \uparrow [\cdot]$  — оператор повышения дискретизации изображения (матрица  $\mathbf{S}^{(l)}$  рассматривается как изображение, с каждым пикселем которого ассоциируется весовое значение), например, посредством дублирования каждого пикселя в строке и столбце.

На последнем, пятом, шаге осуществляется сложение полученных матриц  $\mathbf{R}^{(0)}, \mathbf{R}^{(1)}, \mathbf{R}^{(L-1)}$  для построения интегральной матрицы усредненных весов:

$$(20) \quad \mathbf{S}(\mathbf{p}) := \frac{1}{L} \sum_{l=0}^{L-1} \mathbf{R}^{(l)}(\mathbf{p}),$$

где  $\mathbf{p}$  — пиксель исходного изображения, соответствующий координатам элементов матриц весов.

В результате выполнения рассмотренных шагов формируется многомасштабная весовая модель, анализ которой позволяет определить параметры формы объектов на изображении. Следует отметить, что применение данной модели позволяет повысить точность работы и снизить вычислительную сложность за счет меньшей площади изображения на верхнем уровне пирамиды.

### 3.3. Хроматическая модель изображения

При анализе изображений световых маркеров с беспилотного подводного аппарата целесообразно использовать модель, содержащую две составляющие — хроматическую и весовую. Первая позволяет учесть цветовые характеристики объектов интереса, а вторая — геометрические особенности их формы.

Тогда модель изображения может быть представлена в виде  $\langle \mathbf{C}, \mathbf{S} \rangle$ , где  $\mathbf{C}$  — хроматическая составляющая, а  $\mathbf{S}$  — весовая составляющая. Весовая составляющая модели формируется посредством процедур, описанных в предыдущих разделах. Хроматическая составляющая формируется в результате цветового преобразования и цветовой сегментации.

Для описания хроматической составляющей модели были исследованы преимущества и недостатки нескольких цветовых пространств: RGB, Lab и HSV. Данные пространства сравнивались с точки зрения способов описания для выделения областей изображения, цвет которых лежит в некотором диапазоне [34]. В результате было отмечено, что основным фактором, ограничивающим и усложняющим применение цветовой модели RGB для проведения цветовой сегментации, является единое описание как освещенности, так и цветовой составляющей объекта, что затрудняет четкое задание диапазонов искомых цветов [35]. Неоспоримым преимуществом цветовой модели  $L^*a^*b^*$  является отдельный канал  $L^*$ , отвечающий за освещенность, и плоскостей ( $a^*$ ,  $b^*$ ), отвечающих за цветность. Однако в этой модели диапазоны цветов (хроматические характеристики) описываются в виде геометрической фигуры на плоскости, возможно, имеющей сложное строение. Данный факт усложняет построение модели распределения цвета для проведения сегментации [36]. Цветовая модель HSV в отличие от  $L^*a^*b^*$  позволяет задавать диапазон искомых цветов путем указания интервала значений канала  $H$  (в большинстве случаев достаточно указать пороговое значение насыщенности, задаваемой в канале  $S$ , выше значения, которого чистота цвета становится приемлемой), что значительно упрощает реализацию метода [37].

Таким образом, в качестве цветового пространства для анализа цветных изображений было выбрано цветовое пространство (модель) HSV. Данная модель базируется на особенностях восприятия информации зрительной системой человека и представляется как совокупность цветовых каналов  $H$ ,  $S$  и  $V$ , которые определяют тон, насыщенность и значение (яркость) цвета соответственно.

Хроматическая составляющая  $\mathbf{C}$  модели изображения представляется следующим образом:

$$(21) \quad \begin{cases} (h(\mathbf{p}), s(\mathbf{p}), v(\mathbf{p})), & \exists k, l, m : h(\mathbf{p}) \in \mathbf{H}^k \wedge s(\mathbf{p}) \in \mathbf{S}^l \wedge v(\mathbf{p}) \in \mathbf{V}^m; \\ (0, 0, 0), & \forall k, l, m : h(\mathbf{p}) \notin \mathbf{H}^k \vee s(\mathbf{p}) \notin \mathbf{S}^l \vee v(\mathbf{p}) \notin \mathbf{V}^m, \end{cases}$$

где  $\mathbf{p}$  — пиксель исходного изображения;  $\mathbf{H}^k$ ,  $\mathbf{S}^l$ ,  $\mathbf{V}^m$  —  $k$ -й,  $l$ -й и  $m$ -й интервалы каналов  $H$ ,  $S$  и  $V$  соответственно из множества интервалов цвета заданных объектов.

#### *3.4. Алгоритм обнаружения объектов круглой формы на изображениях световых маркеров*

В общем виде процедура обнаружения объектов круглой формы заданного цвета заключается в следующем:

- 1) преобразовать исходное изображение из цветовой модели RGB в цветовую модель HSV;
- 2) построить хроматическую составляющую модели изображения;
- 3) преобразовать исходное изображение в полутоновый вид с учетом хроматической составляющей;
- 4) найти окружности методом Хафа на изображении по хроматической составляющей;
- 5) построить весовую составляющую модели изображения (весовое изображение);
- 6) выполнить пороговую обработку весового изображения;
- 7) найти на весовом изображении контуры;
- 8) вычислить центры масс для каждого контура;
- 9) найти окружности на изображении по весовой составляющей;
- 10) удалить ложно найденные на полутоновом изображении методом Хафа окружности, центры которых не совпадают с допустимой погрешностью с каким-либо центром масс контуров на весовом изображении;
- 11) сформировать множество координат целевых точек управления с указанием радиусов  $\tilde{A} = \{(x_{target}^k, y_{target}^k), r^k\}$ .

#### 4. Вычислительные эксперименты

Представленный алгоритм был опробован на задаче распознавания круговых светящихся маркеров док-станции подводного базирования под разными углами в условиях замутненной воды. Для вычислительных экспериментов использованы изображения док-станции, полученные в ходе экспериментов Liu S., Ozay M., Okatani T., опубликованных в [20]. Примеры изображений стыковочного блока док-станции в виде подсвеченных кругов, которые распознавались в результате экспериментов, представляет рис. 6.

Изображения получены монокулярной цветной камерой NanoSeaCam разрешением 620 ТВЛ (0,8 Мпикс) с углами обзора  $59^\circ \times 44^\circ \times 72^\circ$  (вверх-вниз-в сторону).

##### 4.1. Критерий оценки точности алгоритма

Оценка эффективности разработанного алгоритма проводилась на основе так называемой пиксельно-ориентированной методики, описанной в [38].

Это статистическая мера оценки, основанная на подсчете неправильно классифицированных пикселей. Для ее проведения требуется вычислить вероятности того, что случайно выбранный пиксель на отсегментированном с помощью алгоритма изображении принадлежит либо искомому объекту, либо, соответственно, фону.

В отличие от таких общепринятых метрик, как Accuracy, Precision или Recall с последующим ROC-анализом, пиксельно-ориентированная метрика оценивает не точность классификации объекта (объект обнаружен или не обнаружен), а качество обнаружения. Оценка качества обнаружения включает

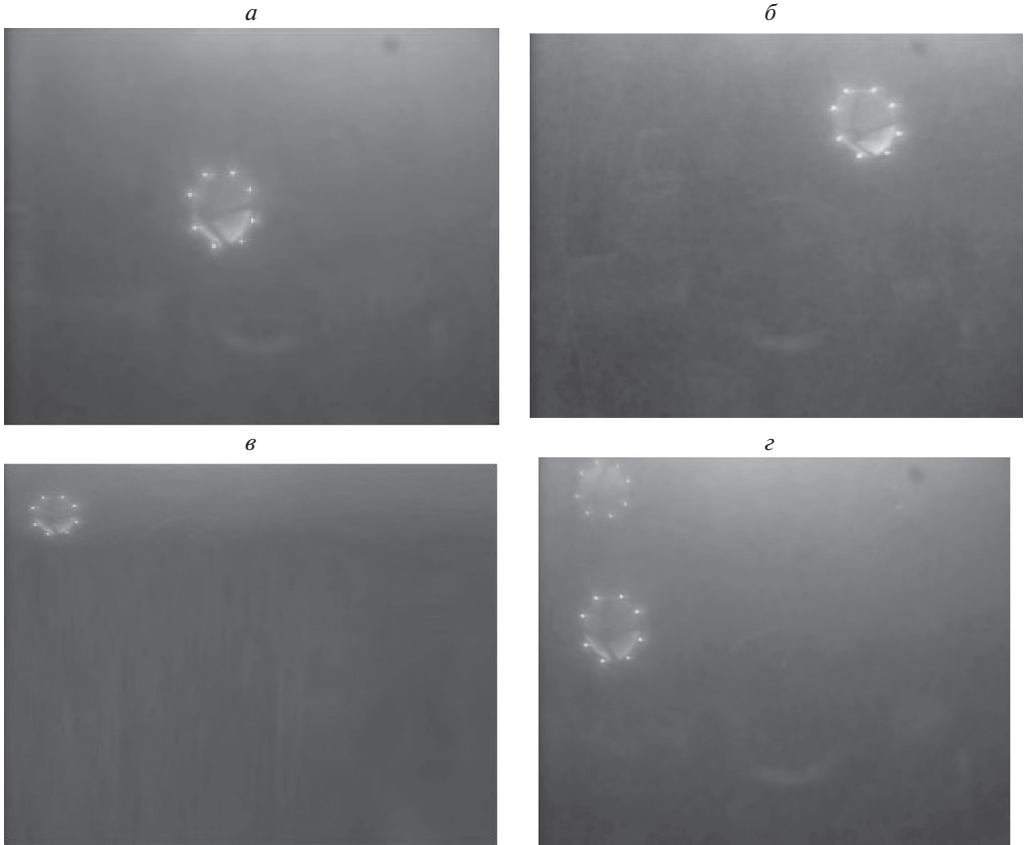


Рис. 6. Варианты изображений окружностей — стыковочного люминесцирующего кольца подводной док-станции: *а)* одиночный объект в центральной части изображения, площадь объекта интереса относительно изображения более 5%; *б)* одиночный объект, смещенный относительно центра изображения, площадь объекта интереса относительно изображения более 5%; *в)* одиночный объект, смещенный относительно центра изображения, площадь объекта интереса относительно изображения менее 5%; *г)* несколько объектов интереса на одном изображении. (источник: датасет в свободном доступе <http://vision.is.tohoku.ac.jp/liushuang/a-vision-based-underwater-docking-system/dataset>).

как оценку точности локализации объекта (определение координат центра), так и точности определения его геометрических характеристик, т.е. радиуса. Поэтому для поставленной задачи пиксельно-ориентированная оценка является предпочтительной [39].

Согласно выбранному методу вероятность ошибки сегментации всего изображения определяется следующим образом:

$$(22) \quad p_{err} = p(b|o)p(o) + p(o|b)p(b).$$

Здесь  $p(o)$  — априорная вероятность того, что случайным образом выбранный пиксель исходного изображения принадлежит объекту (отношение площади

**Таблица 1.** Исходные данные и предварительные результаты экспериментов по распознаванию тестовых изображений

№ эксперимента	Априорные данные				Данные, полученные после применения алгоритма	
	$p(o)$	$p(b)$	Относительная площадь объекта на изображении, %	Оценка относительной площади объекта	Процент ошибочных распознаваний алгоритма по критерию $p(b o)$	Процент ошибочных распознаваний алгоритма по критерию $p(o b)$
1	0,01	0,99	1%	$\leq 5\%$	1	1
2	0,01	0,99	1%	$\leq 5\%$	1	5
3	0,01	0,99	1%	$\leq 5\%$	1	10
4	0,01	0,99	1%	$\leq 5\%$	1	25
5	0,05	0,95	5%	$\leq 5\%$	1	1
6	0,05	0,95	5%	$\leq 5\%$	1	5
7	0,05	0,95	5%	$\leq 5\%$	1	10
8	0,05	0,95	5%	$\leq 5\%$	1	25
9	0,1	0,9	10%	$> 5\%$	1	1
10	0,1	0,9	10%	$> 5\%$	1	5
11	0,1	0,9	10%	$> 5\%$	1	10
12	0,1	0,9	10%	$> 5\%$	1	25
13	0,25	0,75	25%	$> 5\%$	1	1
14	0,25	0,75	25%	$> 5\%$	1	5
15	0,25	0,75	25%	$> 5\%$	1	10
16	0,25	0,75	25%	$> 5\%$	1	25
17	0,5	0,5	50%	$> 5\%$	1	1
18	0,5	0,5	50%	$> 5\%$	1	5
19	0,5	0,5	50%	$> 5\%$	1	10
20	0,5	0,5	50%	$> 5\%$	1	25

объекта к площади всего изображения),  $p(b)$  — априорная вероятность того, что случайным образом выбранный пиксель исходного изображения принадлежит фону (отношение площади фона к площади всего изображения),  $p(o|b)$  — вероятность того, что пиксель, принадлежащий фону, при сегментации ошибочно отнесен к объекту (отношение части площади фона, ошибочно отнесенной алгоритмом к объекту, ко всей площади фона). В теории статистических гипотез такая ошибка называется ошибкой первого рода,  $p(b|o)$  — вероятность того, что пиксель, принадлежащий объекту, будет ошибочно отнесен к фону (отношение части площади объекта, ошибочно отнесенной алгоритмом к фону, к площади объекта) — ошибка второго рода.

Таким образом,  $p(o|b)$  и  $p(b|o)$  являются совокупными ошибками сегментации.



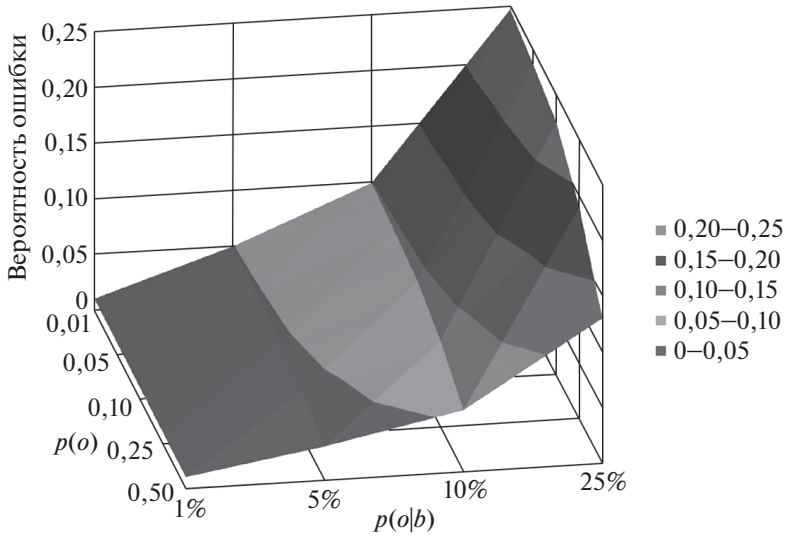


Рис. 7. Вероятность ошибки алгоритма, рассчитанной по способу (22) в зависимости от априорной вероятности  $p(o)$  и апостериорной вероятности  $p(b|o)$ .

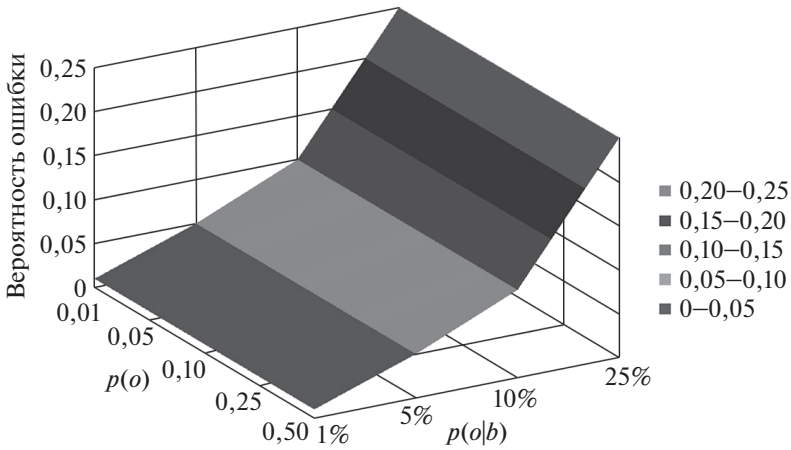


Рис. 8. Вероятность ошибки алгоритма, рассчитанной по способу (23) в зависимости от априорной вероятности  $p(o)$  и апостериорной вероятности  $p(b|o)$ .

Площадь в данном случае понимается как количество пикселей заданной области.

Для случаев, когда искомые объекты малы по размерам относительно всего изображения (занимают менее 5% площади изображения), т.е.

$$\#(o) \ll \#(\mathcal{D}),$$

где  $\#(o)$  — мощность множества пикселей, принадлежащих искомому объекту, а  $\#(\mathcal{D})$  — мощность множества пикселей всего изображения, вместо

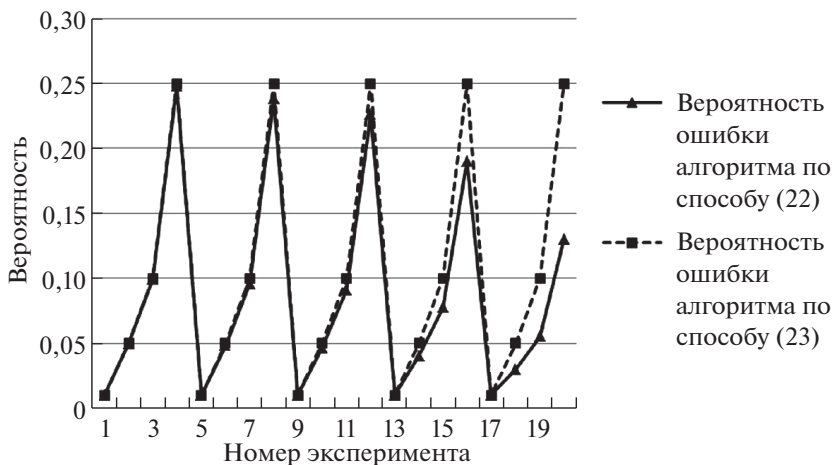


Рис. 9. Сравнительная эффективность способов оценки точности алгоритма.

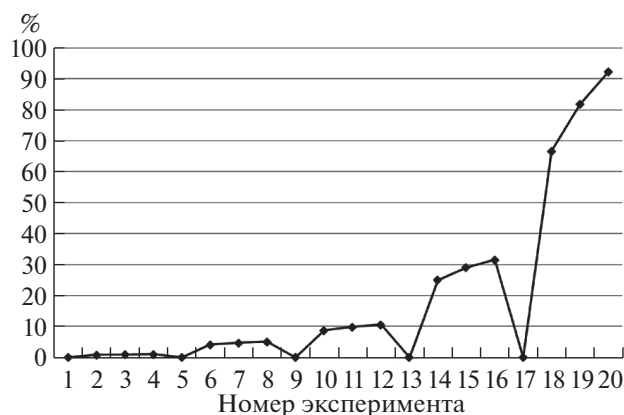


Рис. 10. Разница в точности способа (22) относительно способа (23).

формулы (22) целесообразно применять оценку (23):

$$(23) \quad p_{err} := \frac{\#(o_s) - \#(o)}{\#(\mathcal{D})},$$

где  $o_s$  — множество найденных объектов в результате сегментации.

Для определения оптимального способа расчета точности алгоритма было проведено 20 независимых экспериментов по распознаванию кругов на изображениях. Среди них на 8 изображениях объект интереса занимал 5 и менее процентов площади, а на 12 изображениях — более 5%. После обработки изображений разработанным алгоритмом были получены апостериорные значения ошибок распознавания  $p(o|b)$  и  $p(b|o)$ . Априорные и апостериорные данные экспериментов представлены в табл. 1.

Затем на основе полученных данных вычислялась точность согласно способам (22) и (23). Результаты представлены на рис. 7–10.

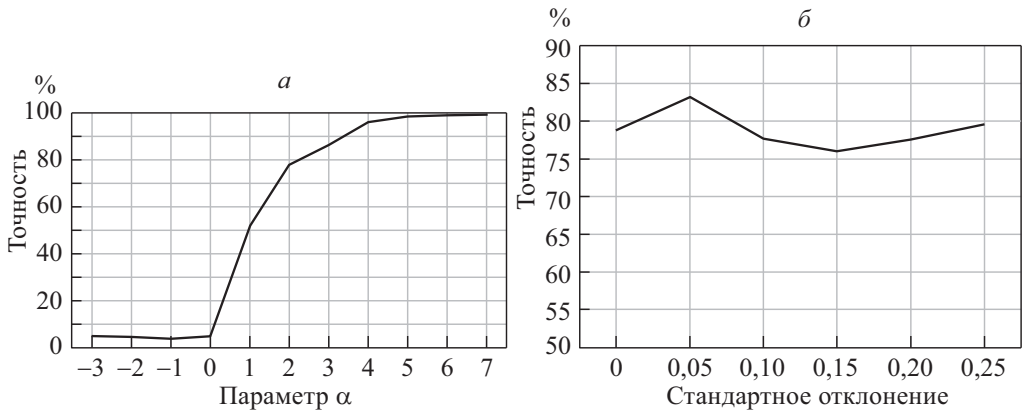


Рис. 11. Влияние параметров алгоритма на точность обнаружения окружностей: а) параметра радиальной жесткости; б) стандартного отклонения.

Результаты показывают, что для объектов с площадью до 5% от площади всего изображения включительно оценки корректности, рассчитанные по обоим способам при зафиксированном уровне ошибки 2-го рода на величине 0,01, в целом имеют небольшие различия. Однако если искомый объект занимает площадь более 5% от площади всего изображения, целесообразно применять оценку по формуле (22). Таким образом, способ (22) является предпочтительным.

Для всех дальнейших исследований точность вычислялась согласно (22).

#### 4.2. Влияние параметров преобразования радиальной симметрии

Исследована зависимость точности обнаружения окружностей от параметра радиальной жесткости  $\alpha$ . Было сгенерировано и исследовано 100 изображений окружностей с радиусами от 1 до 100 пикселей на исходном изображении  $2400 \times 8400$  пикселей. Были проведены эксперименты с изменением параметра жесткости от  $\alpha = -3$  до  $\alpha = +7$ . Результаты представлены на рис. 11,а. Видно, что при увеличении параметра радиальной жесткости  $\alpha$  до значения +2 точность обнаружения достигает 75% (0,75). При дальнейшем увеличении  $\alpha$  точность 95% достигается при значении +4 и далее асимптотически стремится к 100%. На рис. 11,б показано влияние стандартного отклонения на точность обнаружения окружностей при значении параметра радиальной жесткости  $\alpha$ , равном 2. Видно незначительное влияние этого параметра на результат при данных условиях.

#### 4.3. Влияние шумов на точность распознавания

Замутненность воды, наличие в ней мелких органических и неорганических включений, искажения, вызванные преломлением света между водой и объективом камеры, с математической точки зрения описываются как адди-

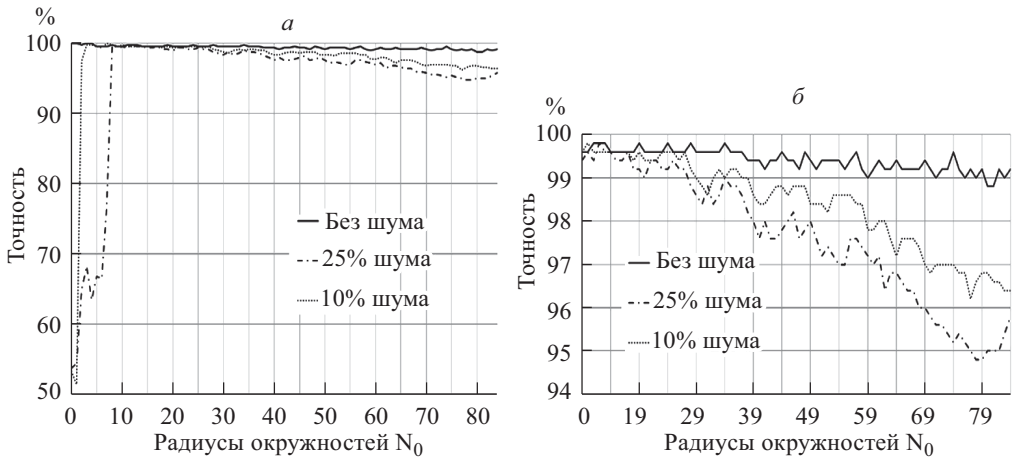


Рис. 12. Влияние импульсного шума на точность распознавания окружностей различного радиуса: а) для всех окружностей; б) для окружностей с радиусом более 10 пикселей.

тивные шумы. В частности, наличие мелкой твердой взвеси в воде моделируется как импульсный шум типа «соль» и «перец». Эффекты искажения и замутненности моделировались как гауссовы шумы и как шумы типа «размытие» с применением масок. Влияние на распознавание каждого типа шумов рассмотрено ниже.

#### 4.3.1. Влияние импульсного шума

Для определения влияния импульсного шума на работу алгоритма было исследовано 84 изображения окружностей с радиусами от 1 до 84 пикселей. Исходное изображение было зашумлено помехами с интенсивностью 25 и 10% площади изображения. Результаты работы алгоритма приведены на рис. 12.

Видно, что импульсный шум значительное влияние оказывает на объекты малых размеров, на результат распознавания более крупных объектов его влияние менее значимо.

#### 4.3.2. Влияние шума Гаусса

Для исследования влияния гауссового шума на результат работы алгоритма исходное изображение было зашумлено с помощью заполнения пикселей изображения нормально распределенными случайными числами с математическим ожиданием  $\mu = 0$  и стандартным отклонением  $\sigma = 5$ . Затем изображение было подвергнуто обработке детектором радиальной симметрии с параметром радиальной жесткости  $\alpha$ , равным 2. Результаты представлены на рис. 13.

Гауссов шум оказывает значительное влияние на объекты малых размеров, для крупных объектов влияние такого рода шума несущественно.

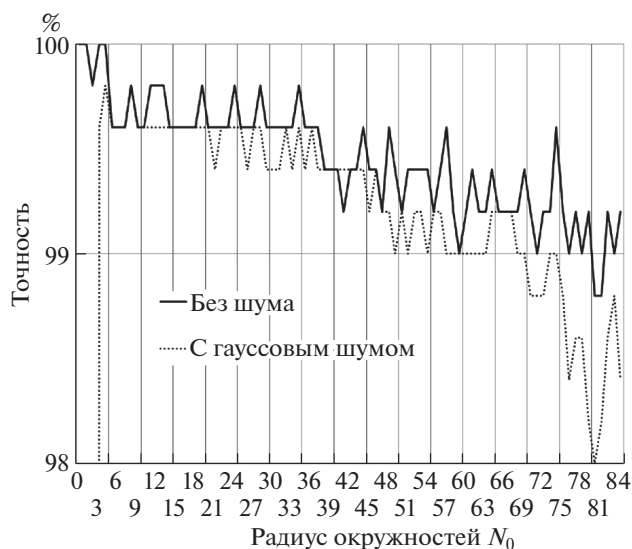


Рис. 13. Влияние гауссового шума на точность распознавания окружностей различного радиуса.

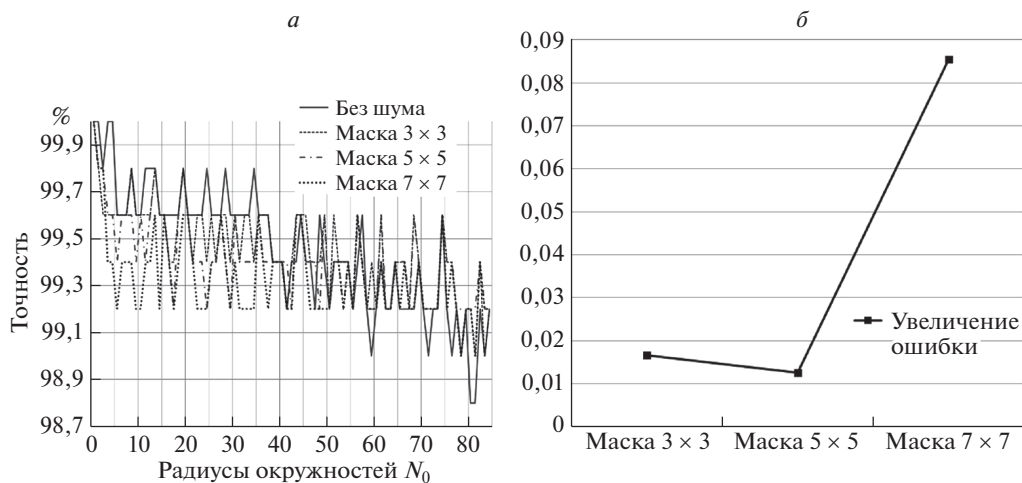


Рис. 14. Влияние размытия изображения масками на точность распознавания кругов: а) сравнительная точность для различных масок размытия; б) относительное среднее увеличение погрешности распознавания для различных масок по сравнению с неискаженным изображением.

#### 4.3.3. Влияние шума типа «размытие»

Для исследования влияния шума типа «размытие» исходное изображение было подвергнуто размытию масками размерности  $3 \times 3$ ,  $5 \times 5$  и  $7 \times 7$ . Результаты изменения точности распознавания окружностей представляет рис. 14.

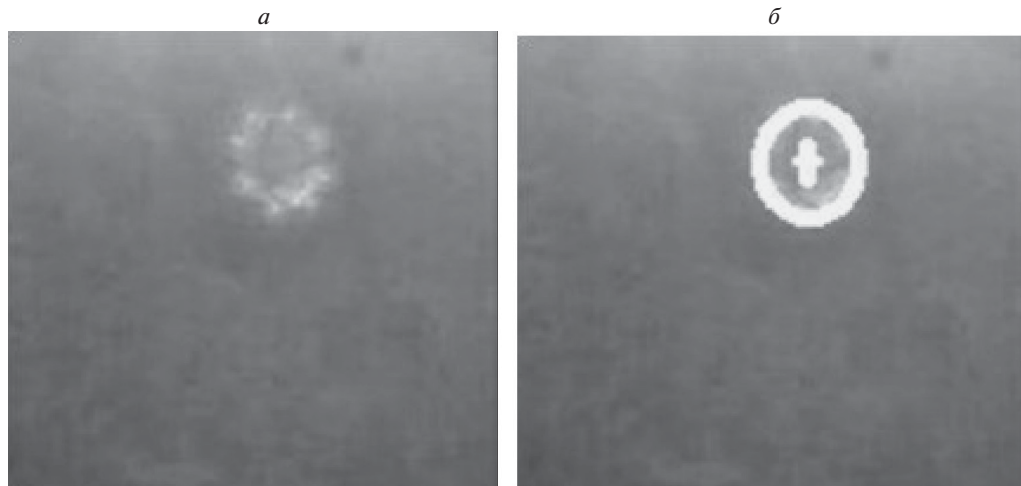


Рис. 15. Пример обнаружения светового маркера: *а)* исходное изображение маркированного кольца док-станции под углом; *б)* распознанная окружность с рассчитанным центром.

Размытие практически не влияет на результат обнаружения объектов, потеря точности составляет не более 0,09%.

Интересным фактом является повышение точности распознавания при наличии шума типа «размытие» по сравнению с незашумленным изображением в нескольких экспериментах. Это связано с проблемой ложного обнаружения, когда алгоритм ошибочно относит к найденным окружностям заданного радиуса также близко расположенные окружности других, близких по значению, радиусов. При наложении шума типа «размытие» вероятность такого ложноположительного обнаружения в некоторых случаях уменьшается.

Полученные результаты исследования влияния шумов различного типа на точность распознавания радиальных изображений показывают, что дополнительно применять алгоритмы повышения резкости не требуется, так как разработанный метод является достаточно устойчивым к шумам в рамках решаемой задачи.

#### *4.4. Детектирование изображения*

Изображение светового стыковочного кольца в замутненной воде относится к изображениям с высокой степенью искажения размытием, в меньшей степени — гауссовым и импульсным шумами. Таким образом, для объектов с радиусом более 10 пикселей (крупные объекты) разработанный алгоритм обеспечивает высокую устойчивость ко всем видам шумов, а по основному искажающему критерию — размытию — обеспечивает устойчивость для распознавания подводного стыковочного люминесцирующего кольца всех размеров.

Пример изображения светового маркера в замутненной воде под углом к камере  $16^\circ$  с распознанным центром для стыковки представлен на рис. 15.

В ходе численных экспериментов были распознаны как сами окружности, так и найдены их центры, координаты которых потенциально могут служить координатами конечного оптимального положения беспилотного аппарата при автоматическом управлении. Для изображения, представленного на рис. 15, в частности, имеем: параметры исходного изображения:

- а) цветное изображение размерностью  $448 \times 448$  пикселей в формате jpeg;
- б) количество искомых окружностей — 1, искомый радиус — 85 пикселей;
- в) множество допустимых цветов — оттенки желтого;
- г) допустимое рассогласование между положениями центров окружностей — 5 пикселей;

найдена окружность с центром в точке:

- $x_{target} = 282$ ;
- $y_{target} = 298$ .

#### *4.5. Сравнение эффективности алгоритмов*

В данном разделе проведено сравнение точности и быстродействия разработанного алгоритма с алгоритмами Хафа и быстрой радиальной симметрии, используемыми в отдельности. Такие методы, как, например, нейросетевое распознавание, требуют дополнительных вычислительных затрат и затрат памяти вычислительного устройства на этапе обучения. В первую очередь это связано с требованием наличия релевантного набора данных для обучения, включающих все возможные сочетания входных параметров для получения адекватной модели. Данное обстоятельство делает применение подобных алгоритмов машинного обучения малоприменимыми для низкопроизводительных вычислительных устройств, установленных на борт БПЛА или подобных беспилотных аппаратов.

##### *4.5.1. Сравнение точности обнаружения радиальных объектов*

Для иллюстрации эффективности метода обнаружения радиальных объектов на изображениях на основе предложенного алгоритма были проведены вычислительные эксперименты для двух типов изображений окружностей — с равномерным и случайным изменением радиусов.

В табл. 2 и 3 приведены результаты сравнения точности алгоритмов при варьировании минимального расстояния между центрами окружностей. Изображение считается правильно распознанным, если величина отклонения найденных центров от реальных не превосходит 15 пикселей.

Результаты сравнения показывают, что точность разработанного алгоритма в среднем на 35% выше точности классических методов, используемых в отдельности.



**Таблица 2.** Сравнение точности распознавания для радиальных объектов с равномерным увеличением радиусов

Минимальное расстояние между центрами окружн., пикс.	Реальное кол-во окружностей на изображении	Найдено по методу Хафа	Найдено по разработанному алгоритму
5	7	21	8
10	7	7	7
15	7	7	7
20	7	7	7
25	7	7	7
30	7	7	7
35	7	7	7
40	6	7	6
45	7	8	8
50	6	6	6

**Таблица 3.** Сравнение точности распознавания для радиальных объектов со случайным изменением радиусов

Минимальное расстояние между центрами окружн., пикс.	Реальное кол-во окружностей на изображении	Найдено по методу Хафа	Найдено по разработанному алгоритму
5	8	152	12
10	8	53	8
15	8	30	8
20	8	22	8
25	8	14	8
30	8	10	8
35	8	10	8
40	8	10	8
45	8	8	8
50	8	6	8

#### 4.5.2. Сравнение скорости обнаружения радиальных объектов

Для относительного сравнения скорости разработанного алгоритма был использован компьютер со следующими характеристиками:

- процессор: Intel Core i5-3230M 2.60 GHz;
- ОЗУ: 8 Гб;
- операционная система: Windows 10, разрядность 64.

Эксперименты проводились для трех тестовых наборов изображений: 2 полных датасета «LISA Traffic Light Dataset» и «Bosch Small Traffic Lights Dataset» набора датасетов открытого типа Kaggle, а также 200 выборочных изображений из различных датасетов набора Kaggle, релевантных проводи-

**Таблица 4.** Характеристики датасетов изображений для экспериментов по сравнению скорости алгоритмов

Имя датасета	Краткое описание
LISA Traffic Light Dataset	100 изображений из набора размером 1280×960 в формате JPG
Bosch Small Traffic Lights Dataset	200 изображений из набора различных размеров в формате PNG
Выборочный набор из Kaggle	200 изображений из набора различных размеров в формате PNG

**Таблица 5.** Сравнение скорости распознавания радиальных объектов различных диапазонов окружностей

Диапазон окружностей для распознавания	Время работы алгоритма FRST, мсек	Время работы разработанного алгоритма, мсек	Выигрыш разработанного алгоритма по скорости, раз
1–10	89 988	50 569	1,8
1–20	171 902	79 753	2,1
1–30	253 435	111 020	2,3
1–50	465 673	128 970	3,6

тому исследованию. Описание тестового набора изображений из этих баз приведено в табл. 4.

Были проведены вычислительные эксперименты по распознаванию окружностей в четырех диапазонах: от 1 до 10 пикселей, от 1 до 20 пикселей, от 1 до 30 пикселей и от 1 до 50 пикселей. В табл. 5 приведены результаты оценки быстродействия предложенного алгоритма многомасштабной модели изображения окружностей по сравнению с базовым методом на основе преобразования радиальной симметрии FRST.

Очевидно, что с ростом диапазона искомых окружностей возрастает преимущество в скорости работы предлагаемого метода по сравнению с базовым.

Необходимо заметить, что использованный для проведения экспериментов компьютер обладал избыточной мощностью для практической реализации предложенного алгоритма и был использован исключительно для целей сравнения авторского алгоритма с классическим. В реальных условиях алгоритму достаточно мощности бортовых вычислителей БПЛПА за счет простоты реализуемых вычислений.

## 5. Заключение

Предложенный подход может быть использован для детектирования и локализации объектов заданных формы и цвета на изображениях световых маркеров для обнаружения объекта интереса беспилотного подводного ли-

бо летательного подводного аппарата. Описанный метод прост в реализации, устойчив к воздействию помех и может быть дополнен различными операциями обработки изображений на всех этапах для повышения эффективности работы, что важно для обнаружения объекта в водной среде с пониженной видимостью.

Кроме того, следует отметить, что процедура формирования многомасштабной весовой модели изображения обладает свойствами естественного параллелизма. Это определяет возможность повышения скорости обработки за счет использования аппаратно-программных средств высокопроизводительных параллельных вычислений, что важно для эффективного управления беспилотным аппаратом в реальном времени и уменьшения его энергопотребления.

Ограничением применимости алгоритма в условиях замутненности воды является размер детектируемого радиального объекта. В частности, объекты радиусом менее 10 пикселей плохо поддаются детектированию разработанным алгоритмом. Более крупные объекты распознаются с высокой точностью свыше 95%. При этом следует задавать параметр радиальной жесткости  $\alpha$  равным +4. Для эффективного обнаружения объектов уровень импульсного шума на изображениях не должен превышать 25%, в то же время гауссов шум имеет малое влияние на обнаружение объектов, радиус которых превышает 5 пикселей. Разработанный алгоритм устойчив к шумам типа «размытие» с усредняющими фильтрами. В силу вышесказанного алгоритм не требует дополнительных методов для повышения четкости изображения, что является его неоспоримым преимуществом перед аналогами.

Данное преимущество получено за счет совместного применения нескольких базовых алгоритмов, таких как метод Хафа, преобразование быстрой радиальной симметрии и построение пирамиды Гаусса. Полученная в результате единая модель превосходит каждый из базовых алгоритмов в отдельности, а также при их попарном применении в быстродействии и в требованиях к вычислительным ресурсам.

## СПИСОК ЛИТЕРАТУРЫ

1. *Liquo T., Shenmin S., Xiaoyan Y., Jianwen S.* An overview of marine recovery methods of UAV for small ships // J. Harbin Institute Technol. 2019. V. 51. No. 10. P. 1–10.
2. *Chamola V., Kotesh P., Agarwal A., Naren, Gupta N., Guizani M.* A Comprehensive Review of Unmanned Aerial Vehicle Attacks and Neutralization Techniques // Ad Hoc Networks. 2021. V. 111. P. 102324. <https://doi.org/10.1016/j.adhoc.2020.102324>
3. ГОСТ Р 56829-2015 Интеллектуальные транспортные системы. Термины и определения, ГОСТ Р от 10 декабря 2015 года № 56829-2015.
4. *Albeaino G., Gheisari M., Franz B.* A Systematic Review of Unmanned Aerial Vehicle Application Areas and Technologies in the AEC Domain // J. of Inform. Technol. Construct. 2019. V. 24. P. 381–405.

5. *Hajiyev C., Soken H.E., Vural S.V.* Navigation Systems for Unmanned Aerial Vehicles // State Estimation and Control for Low-cost Unmanned Aerial Vehicles. 2015. P. 25–49. [https://doi.org/10.1007/978-3-319-16417-5\\_3](https://doi.org/10.1007/978-3-319-16417-5_3)
6. *Moiseev V.S., Shafigullin R.R., Gushchina D.S.* Rational Placement and Required Number of Information Unmanned Aerial Systems for On-Line Monitoring of Large Territories // Russian Aeronautics. 2012. V. 55. P. 223–229. <https://doi.org/10.3103/S1068799812030014>
7. *Neira J., Sequeiros C., Huaman R., Machaca E., Fonseca P., Nina W.* Review on Unmanned Underwater Robotics, Structure Designs, Materials, Sensors, Actuators, and Navigation Control // J. Robot. 2021. P. 1–26. <https://doi.org/10.1155/2021/5542920>
8. *Колесников М.П., Мартынова Л.А., Пашкевич И.В., Шелест П.С.* Метод позиционирования автономного обитаемого подводного аппарата в процессе приведения к причальному устройству // Изв. ТулГУ. Технические науки. 2015. № 11-2. С. 38–49.
9. *Qiu S., Cui W.* An Overview on Aquatic Unmanned Aerial Vehicles // Ann. Rev. Res. 2019. V. 5. No. 3. P. 555663. <https://doi.org/10.19080/ARR.2019.05.555663>
10. *Popescu D., Ichim L.* Image Recognition in UAV Application Based on Texture Analysis // Advanced Concepts for Intelligent Vision Systems. ACIVS 2015. Lecture Notes in Computer Science. 2015. V. 9386. [https://doi.org/10.1007/978-3-319-25903-1\\_60](https://doi.org/10.1007/978-3-319-25903-1_60)
11. *Samadzadegan F., Dadrass Javan F., Ashtari Mahini F., Gholamshahi M.* Detection and Recognition of Drones Based on a Deep Convolutional Neural Network Using Visible Imagery // Aerospace. 2022. V. 9. No. 1. P. 31. <https://doi.org/10.3390/aerospace9010031>
12. *Fujiyoshi H., Hirakawa T., Yamashita T.* Deep Learning-Based Image Recognition for Autonomous Driving // IATSS Res. 2019. V. 43. No. 1. P. 244–252. <https://doi.org/10.1016/j.iatssr.2019.11.008>
13. *Севостьянов И.Е., Девитт Д.В.* Система визуального позиционирования многороторных беспилотников для совершения высокоточной автономной посадки // Science Time. 2021. № 90. С. 38–42.
14. *Степанов Д.Н.* Методы и алгоритмы определения положения и ориентации беспилотного летательного аппарата с применением бортовых видеокамер // Программные продукты и системы. 2014. № 1. С. 150–157.
15. *Deltheil C., Didier L., Hospital E., Brutzman D.P.* Simulating an Optical Guidance System for the Recovery of an Unmanned Underwater Vehicle // IEEE J. Ocean. Engineer. 2000. V. 25. No. 4. P. 568–574. <https://doi.org/10.1109/48.895364>
16. *Guo D., Bacciaglia A., Simpson M., Bil C., Marzocca P.* Design and Development a Bimodal Unmanned System // AIAA Scitech 2019 Forum. 2019. P. 1–7. <https://doi.org/10.2514/6.2019-2096>
17. *Pinheiro P.M., Neto A.A., Grando R.B., Silva C.B. da, Aoki Vivian M., Cardoso D.S., Horn A.C., Drews P.L.J.* Trajectory Planning for Hybrid Unmanned Aerial Underwater Vehicles with Smooth Media Transition // J. Intelligent Robot. Syst. 2022. V. 104. No. 46. <https://doi.org/10.1007/s10846-021-01567-z>

18. *Lock R.J., Vaidyanathan R., Burgess S.C., Loveless J.* Development of a Biologically Inspired Multi-Modal Wing Model for Aerial-Aquatic Robotic Vehicles through Empirical and Numerical Modelling of the Common Guillemot, *Uria Aalge* // *Bioinspirat. Biomimetics*. 2010. V. 5. No. 4. P. 1–15.  
<https://doi.org/10.1088/1748-3182/5/4/046001>
19. *Wu Y., Li L., Su X., Gao B.* Dynamics Modeling and Trajectory Optimization for Unmanned Aerial-Aquatic Vehicle Diving into the Water // *Aerospace Sci. Technol.* 2019. V. 89. P. 220–229. <https://doi.org/10.1016/j.ast.2019.04.004>
20. *Liu S., Ozay M., Okatani T., Xu H., Sun K., Lin Y.* Detection and Pose Estimation for Short-Range Vision-Based Underwater Docking // *IEEE Access*. 2019. V. 30. No. 7. P. 2720–2749. <https://doi.org/10.1109/ACCESS.2018.2885537>
21. *Cowen S., Briest S., Dombrowski J.* Underwater Docking of Autonomous Undersea Vehicles Using Optical Terminal Guidance // *Oceans '97. MTS/IEEE Conference Proceedings, Halifax, NS, Canada*. 1997. V. 2. P. 1143–1147.  
<https://doi.org/10.1109/OCEANS.1997.624153>
22. *Negre A., Pradalier C., Dunbabin M.* Robust Vision-Based Underwater Homing Using Self-Similar Landmarks // *J. Field Robot.* 2008. V. 25. No. 6–7. P. 360–377.  
<https://doi.org/10.1002/rob.20246>
23. *Ghosh S., Ray R., Vadali S.R.K., Shome S.N., Nandy S.* Reliable Pose Estimation of Underwater Dock Using Single Camera: A Scene Invariant Approach // *Machine Vision Appl.* 2016. V. 27. No. 2. P. 221–236.  
<https://doi.org/10.1007/s00138-015-0736-4>
24. *Li Y., Jiang Y., Cao J., Wang B., Li Y.* AUV Docking Experiments Based on Vision Positioning Using Two Cameras // *Ocean Engineer.* 2015. V. 110. P. 163–173.
25. *Чичкарев Е., Сергиенко А., Балалаева Е.* Использование моделей машинного обучения и сетей глубокого обучения для распознавания рукописных чисел и букв русского и латинского алфавитов // *InterConf. Prague, Czech Republic*. 2021. P. 363–380. <https://doi.org/10.51582/interconf.21-22.11.2021.044>
26. *Сирота А.А., Митрофанова Е.Ю., Милованова А.И.* Анализ алгоритмов поиска объектов на изображениях с использованием различных модификаций сверточных нейронных сетей // *Вестник ВГУ. Серия: Системный анализ и информационные технологии*. 2019. № 3. С. 123–137.  
<https://doi.org/10.17308/sait.2019.3/1313>
27. *Zou Z., Shi Z., Guo Y., Ye J.* Object Detection in 20 Years: A Survey // *Proceedings of the IEEE*. 2019. No. 111. P. 257–276.
28. *Клетеник Д.В.* Сборник задач по аналитической геометрии. М.: Наука, Физматлит, 1998.
29. *Gonzalez R.C., Woods R.E.* *Digital Image Processing*. Third Edition. London: Pearson, 2007.
30. *Hough P.V.C.* Machine Analysis of Bubble Chamber Pictures // *2nd International Conference on High-Energy Accelerators and Instrumentation, HEACC 1959*. CERN, Geneva, Switzerland. 1959. P. 554–558.
31. *Lyasheva S., Shleymovich M., Shakirzyanov R.* The Image Analysis Using Fast Radial Symmetry Transform in Control Systems Base on the Computer Vision // *International Multi-Conference on Industrial Engineering and Modern Technologies, FarEastCon*. 2019. P. 1–6. <https://doi.org/10.1109/FarEastCon.2019.8934298>

32. *Shakirzyanov R.M.* Detection of Traffic Signals Using Color Segmentation and a Radial Symmetry Detector // Bulletin of the Voronezh State Technical University. 2020. V. 16. No. 6. P. 25–33.
33. *Adelson E., Burt P., Anderson C., Ogden J.M., Bergen J.* Pyramid Methods in Image Processing // RCA Engineer. 1984. V. 29. No. 6. P. 33–41.
34. *O'Malley R., Jones E., Glavin M.* Rear-Lamp Vehicle Detection and Tracking in Low-Exposure Color Video for Night Conditions // IEEE Transactions on Intelligent Transportation Systems. 2010. V. 11. No. 2. P. 453–462.  
<https://doi.org/10.1109/TITS.2010.2045375>
35. *Друки А.А.* Алгоритмы выделения лиц на статических RGB изображениях и в видеопотоке // Изв. ТПУ. 2012. № 5. С. 65–69.
36. *Darge A., Rajendran R.S., Zerihum D., Chung P.Y.K.* Multi Color Image Segmentation using L\*A\*B\* Color Space // Int. J. Advanced Engineer., Management Sci. 2019. V. 5. P. 346–352. <https://doi.org/10.22161/ijaems.5.5.8>
37. *Forsyth D., Ponce J.* Computer Vision: A Modern Approach. London: Pearson, 2012.
38. *Droogenbroeck V.M., Barnich O.* Design of Statistical Measures for the Assessment of Image Segmentation Schemes // Proceedings of 11th International Conference on Computer Analysis of Images and Patterns (CAIP2005), Lecture Notes in Computer Science. Rocancourt, France, 2005. V. 3691. P. 280–287.
39. *Кольцов П.П., Осипов А.С., Куцаев А.С., Кравченко А.А., Котович Н.В., Захаров А.В.* О количественной оценке эффективности алгоритмов анализа изображений // Компьютерная оптика. 2015. Т. 39, № 4. С. 542–556.

*Статья представлена к публикации членом редколлегии О.А. Степановым.*

Поступила в редакцию 19.11.2022

После доработки 16.03.2023

Принята к публикации 28.04.2023