

ОНТОЛОГИЧЕСКАЯ МОДЕЛЬ ПОСТРОЕНИЯ КОНТУРОВ ОБЪЕКТОВ НА ИЗОБРАЖЕНИИ

М. В. Бобырь¹ [0000-0002-5400-6817], В. П. Добрица² [0000-0001-7533-3684],
А. С. Сизов³ [0000-0001-9658-0318], А. А. Дородных⁴ [0000-0003-0292-3127]

^{1, 2, 3}Юго-Западный государственный университет, Курская обл., Курск, 305040
Россия

⁴Научно-исследовательский институт органических полупродуктов
и красителей, Московская обл., Долгопрудный, 141701, Россия

¹maxbobyrg@gmail.com, ²dobritsa@mail.ru, ³sizov@mail.ru, ⁴alex.dorodnych@mail.ru

Аннотация

В настоящее время разработка онтологических моделей построения границ и их контуров по движущимся объектам в реальном времени или близком к нему является актуальной задачей. В связи с этим в статье представлена онтологическая модель реализации такого процесса. Рассмотрены основные алгоритмы детекции границ объектов на изображении, а также представлены программные коды для их реализации. Отмечено, что для распознавания контуров лучшим является алгоритм Канни. Вместе с этим определён и его серьёзный недостаток, заключающийся в том, что при незначительном движении объектов более 50% информации о контурах теряется.

Ключевые слова: границы объектов, контура объектов, Канни, Собель, Прюитт, Робертс, Лапласиан

ВВЕДЕНИЕ

Детектирование границ и контуров объектов на изображении в режиме реального времени и/или близком к нему является одной из ключевых задач компьютерного зрения и обработки изображений. Алгоритмы построения контуров применяются в различных областях, таких как: медицинская визуализация с целью анализа медицинских изображений, например рентгенография, МРТ и ис-

пользование в виртуальных биологических лабораториях [1]; робототехника с целью навигации и взаимодействия роботов с окружающей средой [2]; геоинформационные системы для анализа и визуализации географических данных [3].

В целом технология построения контуров включает две вычислительные процедуры: детектирование границ и построение контуров по найденным границам. Наиболее часто используемыми алгоритмами детекции контуров являются следующие модели. Детектор Канни наиболее часто используется в библиотеке OpenCV для обнаружения границ и включает несколько шагов: сглаживание изображения с помощью гауссова фильтра, вычисление градиентов, подавление немаксимальных значений и двойной порог для выделения потенциальных границ [4, 5]. Детектор Собеля применяет сверточные матрицы для вычисления градиентов интенсивности изображения в горизонтальном и вертикальном направлениях. Полученные градиенты применяются для вычисления угла градиента и построения границ объектов по ним [6, 7]. Детектор Прюитта аналогичен алгоритму Собеля, но использует другие коэффициенты в сверточных матрицах для вычисления градиентов и их углов [8, 9]. Детектор Робертса основан на двухмерных диагональных сверточных масках, за счёт чего является наиболее быстродействующим фильтром [10, 11]. Детектор Лапласиана использует вторые производные для вычисления изменения интенсивности пикселей на изображении с целью определения точек, в которых изменение градиентов очень резкое [12, 13].

К алгоритмам построения контуров относятся следующие модели. Алгоритм Сузуки–Абе является одним из наиболее распространённых алгоритмов выделения контуров, он реализован в OpenCV [14, 15]. Алгоритм Дугласа–Пекера упрощает процесс построения полилиний, так как запоминает только координаты пикселей, в которых изгибается контур, тем самым уменьшая количество вершин в распознанном контуре объекта [16, 17]. Алгоритм Чана–Весе используется для сегментации изображений и выделения контуров путем их минимизации [18, 19].

Следует отметить, что методы обнаружения границ и выделения контуров играют важную роль в задачах компьютерной обработки изображений. Выбор конкретного метода зависит от задачи и характеристик изображений, таких как уровень шума и сложность объектов.

МЕТОДЫ И МАТЕРИАЛЫ

Онтологическая модель построения границ и контуров объектов (ОМПГКО) на изображении в реальном времени или близком к нему представляет собой кортеж, состоящий из набора понятий, их атрибутов и связей между ними:

$$O = \left\langle \begin{matrix} 5 & 5 & 4 \\ O_k, & O_a, & O_c \\ n=1 & a=1 & c=1 \end{matrix} \right\rangle, \quad (1)$$

где O_k – онтологическая модель классов объектов; O_a – онтологическая модель свойств атрибутов; O_c – связей между понятиями и атрибутами.

ОМПГКО в виде семантической сети представлена на рис. 1.

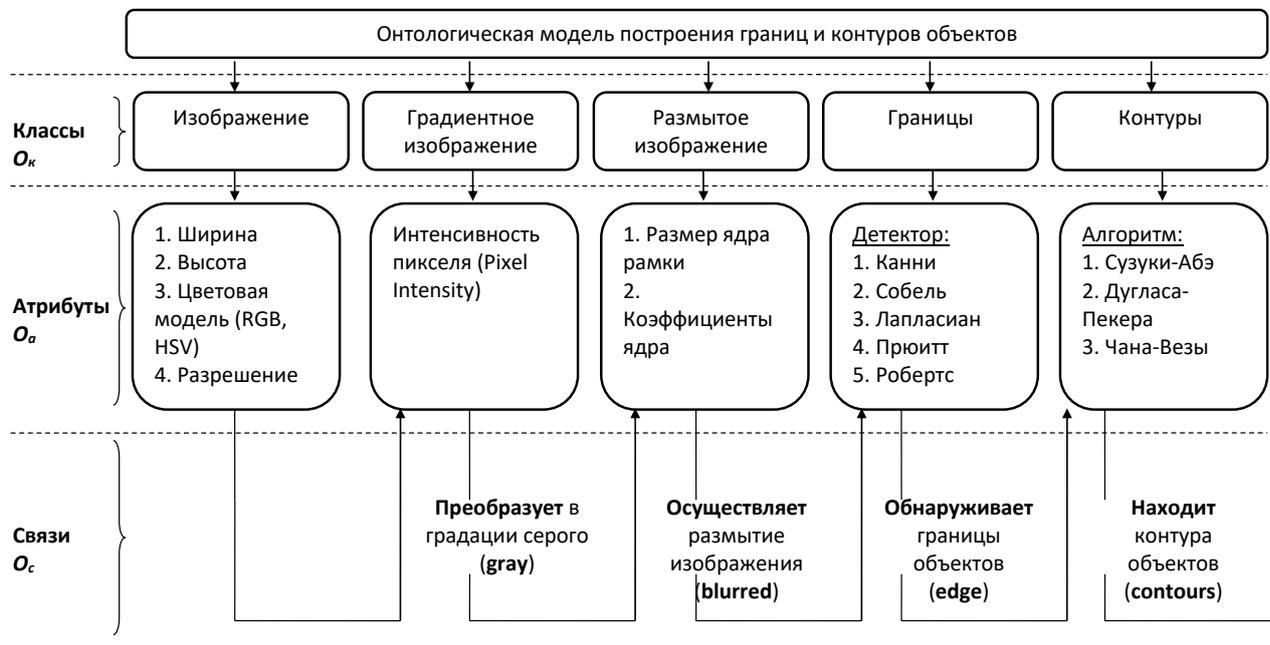


Рис. 1. Онтологическая модель построения границ объектов

Процесс обработки изображения с помощью ОМПГКО осуществляют в следующей последовательности. Сначала исходное изображение преобразуют в градиенты серого с целью уменьшения количества каналов с четырех (RGBA) до одного. Затем осуществляют размытие изображения с помощью одного из фильтров (линейного, гауссова, билатерального) с целью уменьшения шума на изображении и более четкого выделения границ на нём. Далее с помощью одного из методов нахождения границ (например, Канни, Собеля, Приютта, Робертса или

Лапласа) выделяют границы на изображении. После этого, используя один из алгоритмов обнаружения контуров (например, Сузуки–Абе, Дугласа–Пекара, Чана–Весы), строят набор контуров, представляющий замкнутые линии, которые описывают формы объектов на изображении.

Одним из эффективных алгоритмов построения контуров является модель Сузуки–Абе, содержащая следующую последовательность вычислительных операций.

1. Бинаризация изображения – преобразование изображения в бинарный формат, где пиксели объекта помечают как 1 (белые), а фон – как 0 (черные).

2. Инициализация. Алгоритм начинает работу поиском первого белого пикселя (запись координат стартового пиксель-объекта), причём с целью удаления ложных границ рекомендуется пользоваться логическим фильтром. Алгоритм начинает работу с поиска первого белого пикселя (записи его координат как стартовой точки контура). Для исключения ложных границ рекомендуется использовать фильтр связности: если сумма белых пикселей (значение 1) восьми соседних пикселей относительно центрального пикселя в окне 3×3 меньше или равна 1, этот (центральный) пиксель считается шумом, пропускается, и осуществляется поиск следующего белого пикселя. С целью повышения быстродействия на данном шаге рекомендуется пользоваться моделью оптимизации подсчёта суммы значений в рамке 3×3 [20].

3. Поиск контуров. Алгоритм обходит пиксели объекта по часовой стрелке, сохраняя координаты точек, которые образуют замкнутую линию вокруг объекта с выделенными границами. Для реализации этого способа рекомендуется пользоваться рекурсивным алгоритмом закрашивания выделенных областей [21].

4. Сохранение контура. Как только замкнутый контур найден, алгоритм дошёл до стартового пиксель-объекта (Шаг 2), найденные координаты пикселей сохраняются, тем самым формируя распознанный контур.

5. Поиск всех объектов. Алгоритм продолжает выполнять шаги 2–4, пока все контуры распознанных объектов не будут найдены на изображении.

Таким образом, ОМПГКО возвращает список всех найденных контуров, каждый из которых представляет собой замкнутую линию вокруг распознанных объектов на изображении.

РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНЫХ ИССЛЕДОВАНИЙ

Для тестирования ОМПГКО было разработано специализированное программное обеспечение на языке высокого уровня Python в среде Anaconda 3. Программный код алгоритмов построения контуров с помощью различных операторов детектирования границ представлен на листингах 1–5.

```
import cv2
import numpy as np

cap = cv2.VideoCapture(0)// захват видео с веб-камеры
cv2.namedWindow("frame")

while True:
    ret, frame = cap.read()
    gray = cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY) // преобразование в серое
    blurred = cv2.GaussianBlur(gray, (5, 5), 0) // размытие
    edge = cv2.Canny(blurred,100,200)// детектор Канни
contours,h = cv2.findContours(edge,cv2.RETR_TREE,cv2.CHAIN_APPROX_SIMPLE)
    cv2.drawContours(frame,contours,-1,(0,0,255),3) // нахождение контуров
    cv2.imshow("frame", frame) // вывод на экран
    cv2.imshow("edge", edge)

    k = cv2.waitKey(1) & 0xFF
    if k == 27:
        break
cap.release()
cv2.destroyAllWindows()
```

Листинг 1. Детектирование границ с помощью оператора Канни

```
sobelx = cv2.Sobel(blurred, cv2.CV_64F, 1, 0, ksize=3)
sobely = cv2.Sobel(blurred, cv2.CV_64F, 0, 1, ksize=3)
edge = np.sqrt(sobelx**2 + sobely**2)
edge = np.uint8(edges)
```

Листинг 2. Фрагмент программного кода для детектирования границ с помощью оператора Собеля

```
laplacian = cv2.Laplacian(gray, cv2.CV_64F, ksize=3)
edge = np.uint8(np.absolute(laplacian))
```

Листинг 3. Фрагмент программного кода для детектирования границ с помощью Лапласиана

```
kernel_x = np.array([[1, 0, -1], [1, 0, -1], [1, 0, -1]])
kernel_y = np.array([[1, 1, 1], [0, 0, 0], [-1, -1, -1]])
prewitt_x = cv2.filter2D(blurred, -1, kernel_x)
prewitt_y = cv2.filter2D(blurred, -1, kernel_y)
edge = (prewitt_x + prewitt_y)
```

Листинг 4. Фрагмент программного кода для детектирования границ с помощью оператора Прюитта

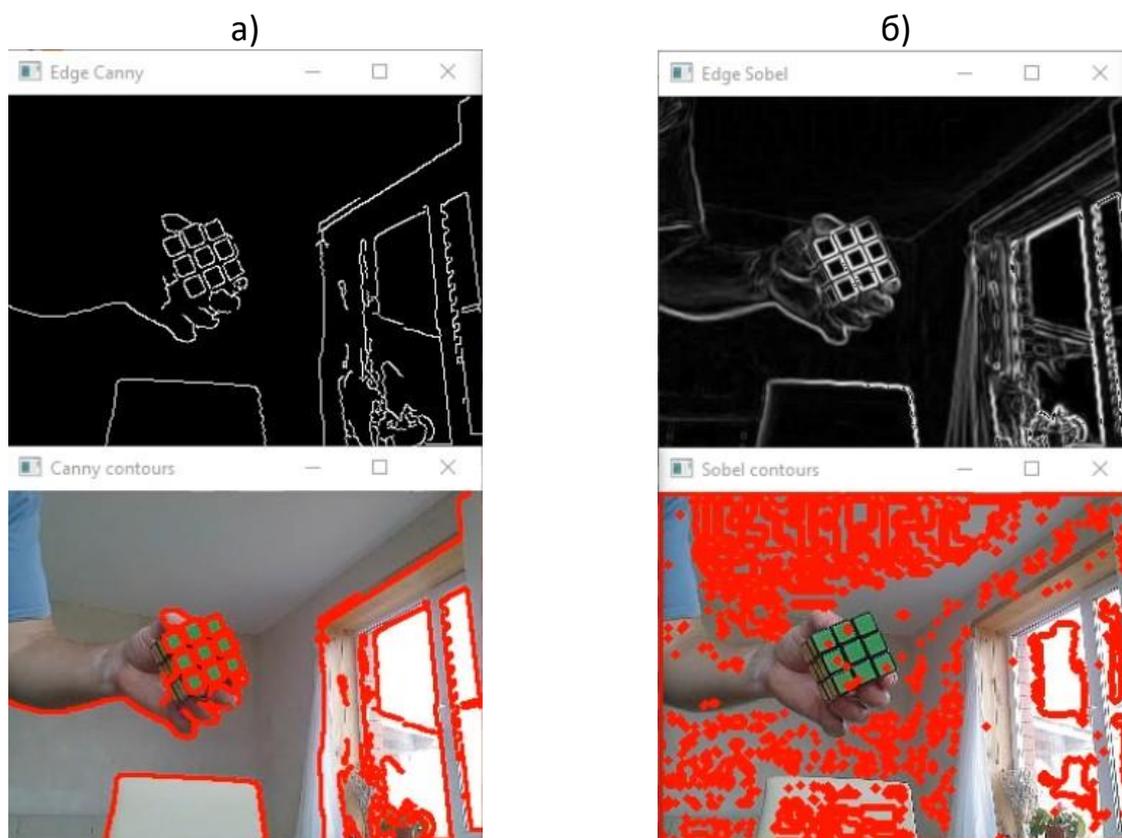
```
kernel_x = np.array([[1, 0], [0, -1]])
kernel_y = np.array([[0, 1], [-1, 0]])
roberts_x = cv2.filter2D(gray, -1, kernel_x)
roberts_y = cv2.filter2D(gray, -1, kernel_y)
edge = (roberts_x + roberts_y)
```

Листинг 5. Фрагмент программного кода для детектирования границ с помощью оператора Робертса

В первом эксперименте осуществлялся визуальный контроль качества детектирования границ и построения по ним контуров. В ходе эксперимента вращался кубик-рубик и программы должны были построить по девять контуров на каждой из его граней. Результат работы алгоритмов представлен на рис. 2.

В ходе первого эксперимента использовался видеоряд, захваченный с веб-камеры, причем для каждого из алгоритмов (листинги 1–5) строились два изображения. На верхнем изображении представлены результаты получения границ объектов с помощью одного из операторов: Канни, Собеля, Лапласиан, Прюитта, Робертса. На нижнем изображении показаны выделенные контуры, найденные

по границам объектов. Анализ результатов работы перечисленных алгоритмов, представленных на рис. 2, показал, что лучшее построение контуров осуществляется при применении алгоритма Канни. При использовании остальных фильтров на исходном изображении появляется много шума. Укажем наиболее предпочтительные детекторы границ в порядке их убывания: Канни → Собель → Лапласиан → Прюитт → Робертс.



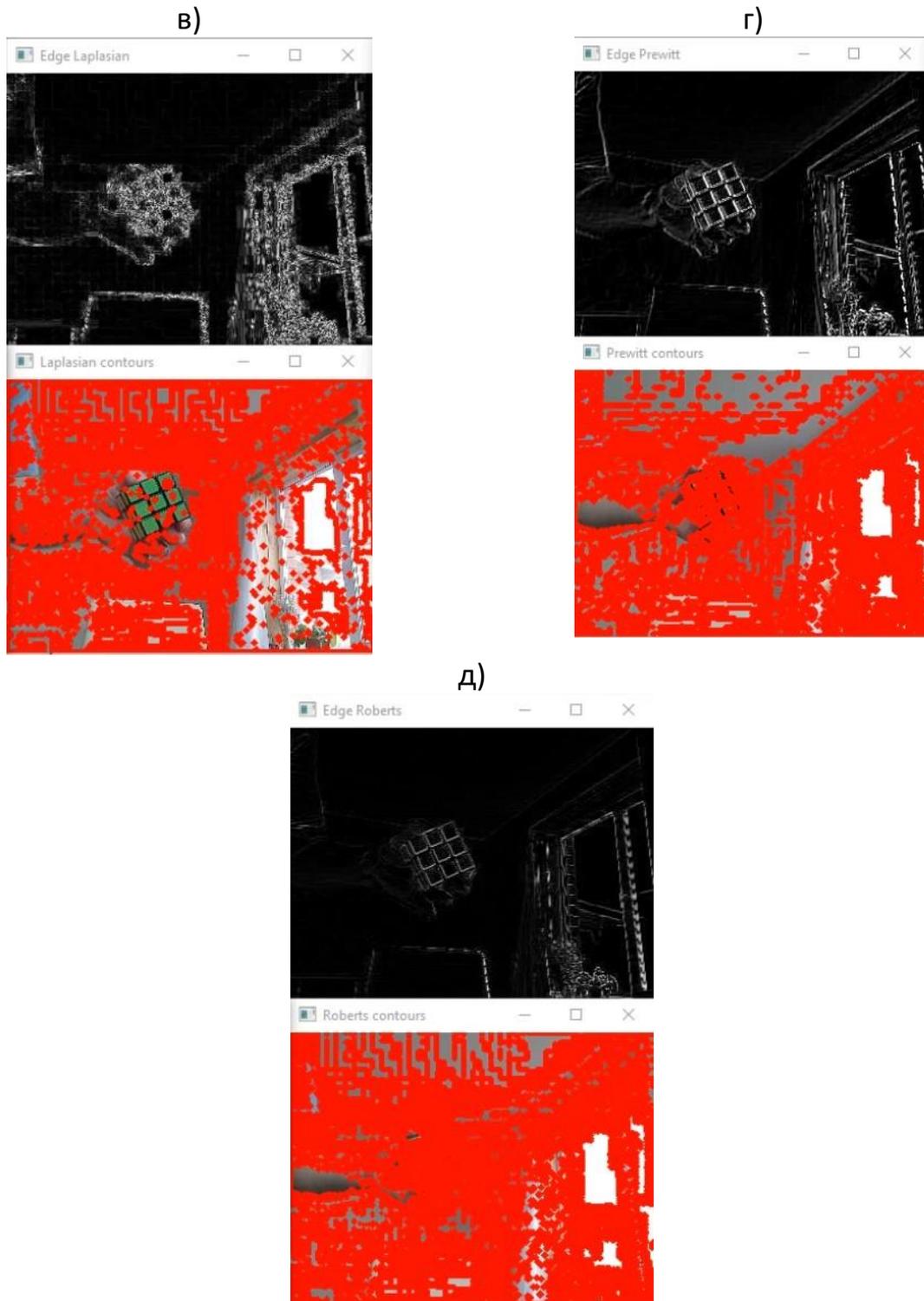


Рис. 2. Детектирование контуров на основе ОМПКО, причем на верхнем изображении представлен один из детекторов, а на нижнем – выделенные контуры на исходном изображении: а – детектор Канни; б – детектор Собеля; в – детектор Лапласа; г – детектор Прюитта; а – детектор Робертса

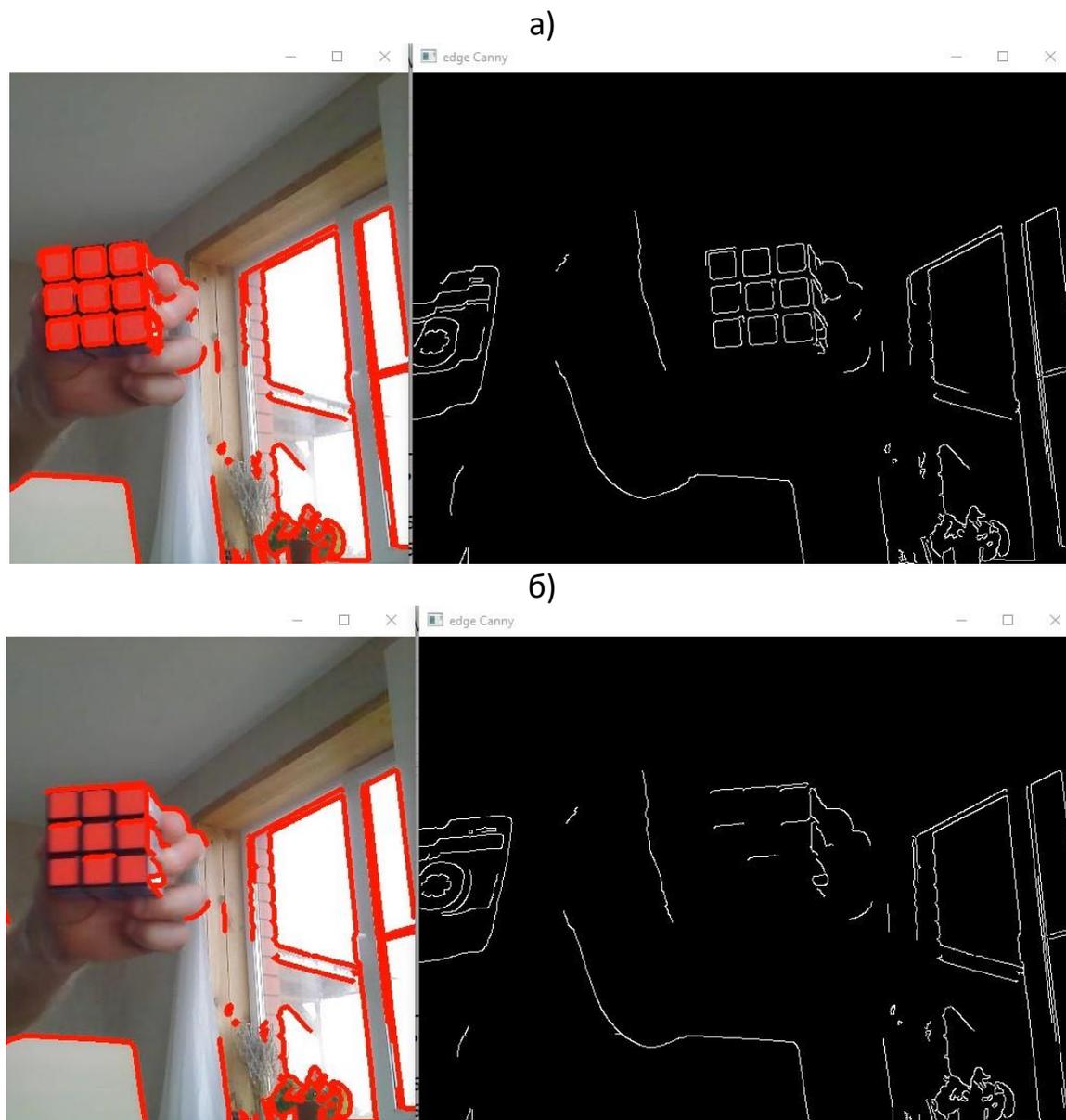


Рис. 3. Детектирование границ объектов с помощью детектора Канни:

а – без движения; б – в движении

Следует отметить, что при определении границ на всех изображениях (кроме детектора Канни) границы представлены градациями серого и значения их интенсивности находятся в диапазоне от 0 до 255, в то время как на выходе алгоритма Канни — бинарное изображение.

В ходе второго эксперимента также использовался видеоряд, захваченный с веб-камеры, и для каждого из алгоритмов (листинги 1–5) анализировалось качество выделения границ при движении объектов, причём использовались два наилучших фильтра: Канни и Собеля. Результаты представлены на рис. 3 и 4.

В ходе второго эксперимента было установлено, что без движения детектор Канни при распознавании границ кубика-рубика достаточно точно их определяет (рис. 3 а), однако при относительно небольшом движении теряет более 50% контуров (рис. 3 б).

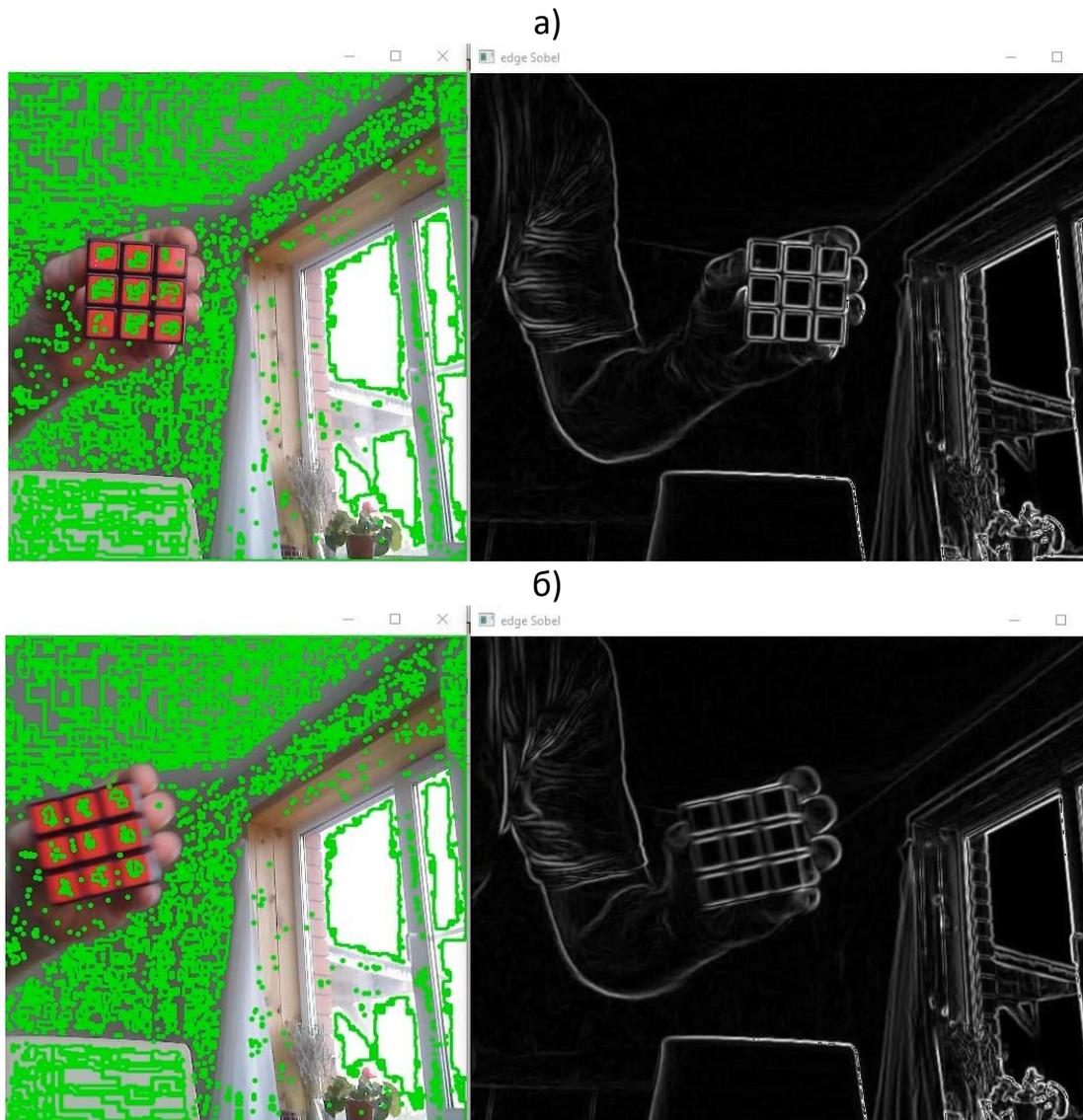


Рис. 4. Детектирование границ объектов с помощью детектора Собеля:
а – без движения; б – в движении

В то же время оператор Собеля недостаточно корректно выделяет контуры как на статичном (рис. 4 а), так и на подвижном (рис. 5 а) изображениях, при этом также происходит смазывание границ объекта. Для повышения качества распознавания границ объектов рекомендуется использовать нечеткую логику [22, 23], включающую новый метод детекции краёв [24].

ЗАКЛЮЧЕНИЕ

Таким образом, рассмотрены пять методов детектирования границ объектов на изображении и построение по ним контуров на основе детекторов: Канни, Собеля, Лапласиана, Прюитта и Робертса. Результаты экспериментального исследования, в ходе которого проводилось детектирование границ и контуров на изображениях, показали превосходство детектора Канни по отношению к другим алгоритмам. Однако детектор Канни имеет серьезный недостаток, заключающийся в том, что при движении более 50% контуров, видимых на статичном изображении, теряются. Результаты эксперимента также показали, что точность распознавания границ и контуров объектов с использованием алгоритма Канни напрямую зависит от скорости движения объектов – при очень высокой скорости потери при детектировании контуров доходят до 90%.

Благодарности

Работа выполнена при финансовой поддержке Государственного задания (проект № 0851-2020-0032).

СПИСОК ЛИТЕРАТУРЫ

1. *Абрамов В.Д., Кугуракова В.В., Ризванов А.А. [и др.]* Виртуальные лаборатории как средство обучения биомедицинским технологиям // Электронные библиотеки. 2016. Т. 19, № 3. С. 129–148.
 2. *Sanz P.* Robotics: Modeling, Planning, and Control. IEEE Robotics and Automation Magazine. 2009. <https://doi.org/10.1109/MRA.2009.934833>
 3. *Reddy G.P.O.* Geographic Information System: Principles and Applications. 2018. P. 45–62. https://doi.org/10.1007/978-3-319-78711-4_3
 4. *Canny J.* A Computational Approach to Edge Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1986. PAMI-8(6). P. 679–698. <https://doi.org/10.1109/TPAMI.1986.4767851>
-

5. *Qin X.* A modified Canny edge detector based on weighted least squares. *Computational Statistics.* (2021). Vol. 36. No. 1. P. 641–659. <https://doi.org/10.1007/s00180-020-01017-8>.
6. *Бо Я., Неусыпин К.А.* Совершенствование метода предварительной обработки звездной карты с использованием оператора Собеля // *Автоматизация. Современные технологии.* 2023. Т. 77. № 7. С. 308–314. <https://doi.org/10.36652/0869-4931-2023-77-7-308-314>.
7. *Tian R., Sun G., Liu X., Zheng B.* Sobel edge detection based on weighted nuclear norm minimization image denoising // *Electronics.* 2021. Vol. 10. No. 6. P. 1–15. <https://doi.org/10.3390/electronics10060655>
8. *Adlakha D., Tanwar R.* Analytical Comparison between Sobel and Prewitt Edge Detection Techniques. *International Journal of Scientific & Engineering Research.* 2016. Vol. 7. No. 1. P. 1482–1485.
9. *Karthick C.N., Nirmala P.* Smart Edge Detection Technique in X-ray Images for Improving PSNR using Prewitt Edge Detection Algorithm with Gaussian Filter in Comparison with Laplacian Algorithm // *Cardiometry.* 2023. №25. P.1758–1762. <https://doi.org/10.18137/cardiometry.2022.25.17581762>
10. *Gong H.X., Hao L.* Roberts edge detection algorithm based on GPU. *Journal of Chemical and Pharmaceutical Research.* 2014. Vol. 6. No. 7. P. 1308–1314.
11. *Utama K.M.R.A., Umar R., Yudhana A.* Edge detection comparative analysis using Roberts, Sobel, Prewitt, and Canny methods// *Jurnal Teknologi Dan Sistem Komputer.* 2022. Vol. 10. No. 2. P. 67–71. <https://doi.org/10.14710/jtsiskom.2021.14209>
12. *Qu Y., Tang W., Su, Cheng S.* Web Inspection Algorithm of Low Contrast Paper Defects Using Gabor Filter // *In Lecture Notes in Electrical Engineering.* 2022. Vol. 920. P. 371–378. Springer Science and Business Media Deutschland GmbH. https://doi.org/10.1007/978-981-19-3927-3_36
13. *Darwis D., Fernando Y., Trisnawati F., Marzuki D.H., Setiawansyah, S.* Comparison of edge detection methods using Roberts and Laplacian operators on mango leaf objects. *Barekeng* // *Jurnal Ilmu Matematika Dan Terapan.* 2023. Vol. 17. No. 3. P. 1815–1824. <https://doi.org/10.30598/barekengvol17iss3pp1815-1824>

14. *Suzuki S., Abe K.* Topological structural analysis of digitized binary images by border following // *Computer Vision, Graphics and Image Processing*. 1985. Vol. 30. P. 32–46. [https://doi.org/10.1016/0734-189X\(85\)90016-7](https://doi.org/10.1016/0734-189X(85)90016-7)
 15. *Mei L., Wang C., Zhao Y., Wei W., Li Y.* Real-time detection method of landmark in UAV autonomous landing // *Systems Engineering and Electronics*. 2019. Vol. 41. No. 10. P. 2157–2162. <https://doi.org/10.3969/j.issn.1001-506X.2019.10.01>
 16. *Zhou Z., Zhang Y., Yuan X., Wang H.* Compressing AIS Trajectory Data Based on the Multi-Objective Peak Douglas-Peucker Algorithm // *IEEE Access*. 2023. No. 11. P. 6802–6821. <https://doi.org/10.1109/ACCESS.2023.3234121>
 17. *Liu J., Li H., Yang Z., Wu K., Liu Y., Liu R.W.* Adaptive Douglas–Peucker Algorithm with Automatic Thresholding for AIS-Based Vessel Trajectory Compression // *IEEE Access*. 2019. No. 7. P. 150677–150692. <https://doi.org/10.1109/ACCESS.2019.2947111>
 18. *Gunasekara S.R., Kaldera K., Dissanayake M.B.* A Systematic Approach for MRI Brain Tumor Localization and Segmentation Using Deep Learning and Active Contouring // *Journal of Healthcare Engineering*. 2021. <https://doi.org/10.1155/2021/6695108>
 19. *Çataloluk H., Çelebi F.V.* A novel hybrid model for two-phase image segmentation: GSA based Chan–Vese algorithm // *Engineering Applications of Artificial Intelligence*. 2018. No. 73. P. 22–30. <https://doi.org/10.1016/j.engappai.2018.04.027>
 20. *Бобырь М.В., Храпова Н.И., Супрунова О.Г., Дородных А.А.* Рекурсивный алгоритм закрашивания областей распознанных объектов // *Известия Юго-Западного государственного университета*. 2023. Т. 27. № 1. С. 126–139.
 21. *Бобырь М.В., Емельянов С.Г., Милостная Н.А.* Оптимизация числа проходов в задаче логической фильтрации изображений // *Искусственный интеллект и принятие решений*. 2023. № 2. С. 98–107. <https://doi.org/10.14357/20718594230208>.
 22. *Бобырь М.В., Кулабухов С.А.* Моделирование процесса управления температурным режимом в зоне резания на основе нечеткой логики // *Проблемы машиностроения и надежности машин*. 2017. № 3. С. 76–82. <https://doi.org/10.3103/S1052618817030049>.
-

23. Бобырь М.В., Храпова Н.И. Двухуровневая информационно-аналитическая система управления интеллектуальным светофором// Электронные библиотеки. 2024. №5. С. 696–717.

24. Бобырь М.В., Архипов А.Е., Горбачев С.В. Нечетко-логические методы в задаче детектирования границ объектов// Информатика и автоматизация. 2022. Т. 21, № 2. С. 376–404.

ONTOLOGICAL MODEL FOR CREATING OBJECT CONTOURS IN AN IMAGE

M. V. Bobyr¹ [0000-0002-5400-6817], V. P. Dobritsa² [0000-0001-7533-3684],

A. S. Sizov³ [0000-0001-9658-0318], A. A. Dorodnykh⁴ [0000-0003-0292-3127]

^{1, 2, 3}Southwest State University, Kursk, Russia

⁴Research Institute of Organic Semi-Products and Dyes, Dolgoprudny, Russia

¹maxbobyr@gmail.com, ²dobritsa@mail.ru, ³sizov@mail.ru, ⁴alex.dorodnych@mail.ru

Abstract

Now days, the development of ontological models for creating edges and their contours for moving objects in real time or close to it is an urgent task. An ontological model for implementing this process is shown in the article. The main algorithms for detecting object edges and constructing contours in an image and program codes for their implementation are considered in the article. It is noted that the Canny algorithm is the best for recognizing edges. At the same time, its serious drawback is determined, which consists in the fact that with insignificant movement of objects, more than 50% of information about the contours is lost.

Keywords: *object edges, object contours, Canny, Sobel, Prewitt, Roberts, Laplacian.*

REFERENCES

1. Abramov V.D., Kugurakova V.V., Rizvanov A.A. [et al.] Virtual laboratories as a means of teaching biomedical technologies // Electronic libraries. 2016. Vol. 19. No. 3. P. 129–148.

2. Sanz P. Robotics: Modeling, Planning, and Control. IEEE Robotics and Automation Magazine. 2009. <https://doi.org/10.1109/MRA.2009.934833>

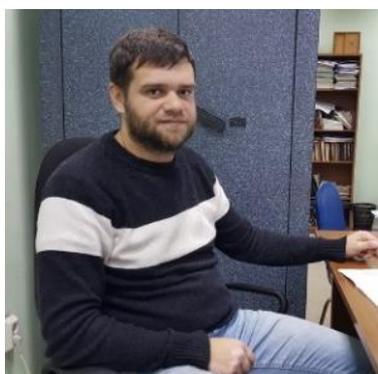
3. *Reddy G.P.O.* Geographic Information System: Principles and Applications. 2018. P. 45–62. https://doi.org/10.1007/978-3-319-78711-4_3
4. *Canny J.* A Computational Approach to Edge Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1986. PAMI-8(6). P. 679–698. <https://doi.org/10.1109/TPAMI.1986.4767851>
5. *Qin X.* A modified Canny edge detector based on weighted least squares. Computational Statistics. (2021). Vol. 36. No. 1. P. 641–659. <https://doi.org/10.1007/s00180-020-01017-8>.
6. *Bo Ya., Neusybin K.A.* Improving the method of preprocessing a star map using the Sobel operator // Automation. Modern technologies. 2023. Vol. 77. No. 7. P. 308–314. <https://doi.org/10.36652/0869-4931-2023-77-7-308-314>.
7. *Tian R., Sun G., Liu X., Zheng B.* Sobel edge detection based on weighted nuclear norm minimization image denoising // Electronics. 2021. Vol. 10. No. 6. P. 1–15. <https://doi.org/10.3390/electronics10060655>
8. *Adlakha D., Tanwar R.* Analytical Comparison between Sobel and Prewitt Edge Detection Techniques. International Journal of Scientific & Engineering Research. 2016. Vol. 7. No. 1. P. 1482–1485.
9. *Karthick C.N., Nirmala P.* Smart Edge Detection Technique in X-ray Images for Improving PSNR using Prewitt Edge Detection Algorithm with Gaussian Filter in Comparison with Laplacian Algorithm // Cardiometry. 2023. No. 25. P. 1758–1762. <https://doi.org/10.18137/cardiometry.2022.25.17581762>
10. *Gong H.X., Hao L.* Roberts edge detection algorithm based on GPU. Journal of Chemical and Pharmaceutical Research. 2014. Vol. 6. No. 7. P. 1308–1314.
11. *Utama K.M.R.A., Umar R., Yudhana A.* Edge detection comparative analysis using Roberts, Sobel, Prewitt, and Canny methods// Jurnal Teknologi Dan Sistem Komputer. 2022. Vol. 10. No. 2. P. 67–71. <https://doi.org/10.14710/jtsiskom.2021.14209>
12. *Qu Y., Tang W., Su, Cheng S.* Web Inspection Algorithm of Low Contrast Paper Defects Using Gabor Filter// In Lecture Notes in Electrical Engineering. 2022. Vol. 920. P. 371–378. Springer Science and Business Media Deutschland GmbH. https://doi.org/10.1007/978-981-19-3927-3_36

13. *Darwis D., Fernando Y., Trisnawati F., Marzuki D.H., Setiawansyah, S.* Comparison of edge detection methods using Roberts and Laplacian operators on mango leaf objects. *Barekeng // Jurnal Ilmu Matematika Dan Terapan*. 2023. Vol. 17. No. 3. P. 1815–1824. <https://doi.org/10.30598/barekengvol17iss3pp1815-1824>
 14. *Suzuki S., Abe K.* Topological structural analysis of digitized binary images by border following // *Computer Vision, Graphics and Image Processing*. 1985. Vol. 30. P. 32–46. [https://doi.org/10.1016/0734-189X\(85\)90016-7](https://doi.org/10.1016/0734-189X(85)90016-7)
 15. *Mei L., Wang C., Zhao Y., Wei W., Li Y.* Real-time detection method of landmark in UAV autonomous landing // *Systems Engineering and Electronics*. 2019. Vol. 41. No. 10. P. 2157–2162. <https://doi.org/10.3969/j.issn.1001-506X.2019.10.01>
 16. *Zhou Z., Zhang Y., Yuan X., Wang H.* Compressing AIS Trajectory Data Based on the Multi-Objective Peak Douglas-Peucker Algorithm // *IEEE Access*. 2023. No. 11. P. 6802–6821. <https://doi.org/10.1109/ACCESS.2023.3234121>
 17. *Liu J., Li H., Yang Z., Wu K., Liu Y., Liu R.W.* Adaptive Douglas–Peucker Algorithm with Automatic Thresholding for AIS-Based Vessel Trajectory Compression // *IEEE Access*. 2019. No. 7. P. 150677–150692. <https://doi.org/10.1109/ACCESS.2019.2947111>
 18. *Gunasekara S.R., Kaldera K., Dissanayake M.B.* A Systematic Approach for MRI Brain Tumor Localization and Segmentation Using Deep Learning and Active Contouring // *Journal of Healthcare Engineering*. 2021. <https://doi.org/10.1155/2021/6695108>
 19. *Çataloluk H., Çelebi F.V.* A novel hybrid model for two-phase image segmentation: GSA based Chan–Vese algorithm // *Engineering Applications of Artificial Intelligence*. 2018. No. 73. P. 22–30. <https://doi.org/10.1016/j.engappai.2018.04.027>
 20. *Bobyry M.V., Khrapova N.I., Suprunova O.G., Dorodnykh A.A.* The recursive algorithm for filling areas of recognized objects // *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta*. 2023. T. 27, № 1. S. 126-139.
 21. *Bobyry M.V., Emelyanov S.G., Milostnaya N.A.* Optimization of the number of passes in the problem of logical image filtering // *Artificial Intelligence and Decision Making*. 2023. № 2. S. 98–107. <https://doi.org/10.14357/20718594230208>.
 22. *Bobyry M.V., Kulabukhov S.A.* Simulation of control of temperature mode in cutting area on the basis of fuzzy logic // *Journal of Machinery Manufacture and Reliability*. 2017. Vol. 46. P. 288–295. <https://doi.org/10.3103/S1052618817030049>.
-

23. *Bobyр M.V., Khrapova N.I.* Two-level information and analytical control system for intelligent traffic lights // *Electroniclibraries*. 2024. №5. S. 696–717.

24. *Bobyр M.V., Arkhipov A.E., Gorbachev S.V.* Fuzzy logic methods in the problem of detecting object boundaries // *Informatics and Automation*. 2022. Vol. 21, No. 2. P. 376–404.

СВЕДЕНИЯ ОБ АВТОРАХ



БОБЫРЬ Максим Владимирович. Доктор технических наук, профессор. Профессор кафедры «Программная инженерия», ФГБОУ ВО «Юго-Западный государственный университет». Области исследований: интеллектуальные системы управления, адаптивные нейро-нечеткие системы вывода, распознавание и обработка изображения, робототехнические системы.

Maxim V. BOBYR. Doctor of technical sciences, professor. Professor of the Department of Software Engineering, Southwest State University. Research areas: intelligent control systems, adaptive neuro-fuzzy inference systems, image recognition and processing, robotic systems.

email: maxbobyр@gmail.com

ORCID: 0000-0002-5400-6817



ДОБРИЦА Вячеслав Порфирьевич. Доктор технических наук, профессор. Профессор кафедры «Информационная безопасность», ФГБОУ ВО «Юго-Западный государственный университет». Области исследований: системы искусственного интеллекта, нейронные сети для обнаружения и локализации кибератак.

Vyacheslav P. DOBRITSA. Doctor of technical sciences, professor. Professor of the Department of Software Engineering, Southwest State University. Research areas: artificial intelligence systems, neural networks for detection and localization of cyber attacks.

email: dobritsa@mail.ru

ORCID: 0000-0001-7533-3684



СИЗОВ Александр Семенович. Доктор технических наук, профессор. Профессор кафедры «Программная инженерия», ФГБОУ ВО «Юго-Западный государственный университет». Области исследований: нейронные сети и нечёткие системы, интеллектуальные системы управления сложными техническими объектами.

Alexander S. SIZOV. Doctor of technical sciences, professor. Professor of the Department of Software Engineering, Southwest State University. Research areas: neural networks and fuzzy systems, intelligent control systems for complex technical objects.

email: sizov@mail.ru

ORCID: 0000-0001-9658-0318



ДОРОДНЫХ Александр Алексеевич. АО «НИОПИК (Научно-исследовательский институт органических полупродуктов и красителей)». Область исследования: вычислительное моделирование для интерфейсов мозг–компьютер, глаз–компьютер и глаз–мозг–компьютер, решение задач в режиме, близком к реальному времени. Системы технического зрения для распознавания объектов с использованием нечеткой логики, нейронных сетей.

Alexander A. DORODNYKH. JSC "NIOPIK (Research Institute of Organic Semi-finished Products and Dyes)". Research area: Research area: near-real-time problem solving and computational modeling for brain–computer, eye–computer and eye–brain–computer interfaces. Machine vision systems for object recognition using fuzzy logic, neural networks.

email: alex.dorodnych@mail.ru

ORCID: 0000-0003-0292-3127

Материал поступил в редакцию 25 ноября 2024 года