

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
(наименование института полностью)

Кафедра «Прикладная математика и информатика»
(наименование кафедры)

02.03.03 Математическое обеспечение и администрирование
информационных систем

(код и наименование направления подготовки, специальности)

Технология программирования

(направленность (профиль)/специализация)

БАКАЛАВРСКАЯ РАБОТА

на тему: «Локализация объектов на изображении на основе анализа их формы»

Студент

Б.М. Элчибекова

(И.О. Фамилия)

(личная подпись)

Руководитель

В.С. Климов

(И.О. Фамилия)

(личная подпись)

Консультанты

К.А. Селиверстова

(И.О. Фамилия)

(личная подпись)

Допустить к защите

Заведующий кафедрой к.т.н., доцент, А.В. Очеповский

(ученая степень, звание, И.О. Фамилия)

(личная подпись)

« _____ » _____ 20 _____ г.

Тольятти 2019

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
(наименование института полностью)

Кафедра «Прикладная математика и информатика»
(наименование кафедры)

УТВЕРЖДАЮ
Завкафедрой «Прикладная
математика и информатика»

_____ А.В. Очеповский
(подпись) (И.О. Фамилия)
« ____ » _____ 20__ г.

ЗАДАНИЕ
на выполнение бакалаврской работы

Студент Элчибекова Боиста Маъруфшоевна

1. Тема Локализация объектов на изображении на основе анализа их формы
2. Срок сдачи студентом законченной выпускной квалификационной работы
21 июня 2019 г.
3. Исходные данные к выпускной квалификационной работе: загрузка изображений из файла, детектор границ Канни, алгоритм поиска контуров на изображении.
4. Содержание выпускной квалификационной работы (перечень подлежащих разработке вопросов, разделов)

Введение

1 Анализ состояния вопроса

2 Разработка алгоритма локализации объектов на изображении на основе анализа их формы

3 Практическая реализация предложенных решений

Заключение

Список используемой литературы

5. Ориентировочный перечень графического и иллюстративного материала: набор формул, объясняющих математический аппарат алгоритма; результаты тестирования работы алгоритма на реальном наборе данных; диаграммы, поясняющие результат работы алгоритма; формы окна программы.
6. Консультанты по разделам _____
7. Дата выдачи задания «15» декабря 2018 г.

Руководитель выпускной
квалификационной работы

_____ В.С. Климов

Задание принял к исполнению

_____ Б.М. Элчибекова

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
(наименование института полностью)

Кафедра «Прикладная математика и информатика»
(наименование кафедры)

УТВЕРЖДАЮ
Завкафедрой «Прикладная
математика и информатика»

_____ А.В. Очеповский
(подпись) (И.О. Фамилия)
« ____ » _____ 20__ г.

КАЛЕНДАРНЫЙ ПЛАН
выполнения бакалаврской работы

Студента Элчибекова Боиста Маъруфшоевна

по теме «Локализация объектов на изображении на основе анализа их формы»

Наименование раздела работы	Плановый срок выполнения раздела	Фактически й срок выполнения раздела	Отметка о выполнении	Подпись руководителя
Анализ предметной области	1.10.2018	1.10.2018		
Разработка алгоритма локализации объектов на изображении на основе анализа их формы	12.12.2018	12.12.2018		
Практическая реализация	17.01.2019	17.01.2019		

предложенных решений				
Подготовка пояснительной записки	25.02.2019	25.02.2019		
Подготовка презентации к защите	25.02.2019	25.02.2019		
Предзащита дипломной работы	30.05.2019	30.05.2019		
Проверка ВКР в системе «Антиплагиат. ВУЗ»	06.06.2019	06.06.2019		
Сдача пояснительной записки и отзыва на кафедру	17.06.2019	17.06.2019		
Защита дипломной работы				

Руководитель выпускной
квалификационной работы

В.С. Климов

Задание принял к исполнению

Б.М. Элчибекова

АННОТАЦИЯ

Тема бакалаврской работы: «Локализация объектов на изображении на основе анализа их формы».

В данной бакалаврской работе исследуются способы локализации объектов на изображении путем анализа их формы.

В работе предложен алгоритм локализации объектов на изображении, основанный на выделении границ методом Канни; эвристическом замыкании всех разомкнутых контуров; сглаживании контуров; сохранение всех контуров в коллекцию; поиск в коллекции контуров, удовлетворяющих заданным критериям; отображение найденных контуров поверх изображения.

При подготовке исследований, представленных в рамках данной бакалаврской работы, была опубликована 1 статья.

Бакалаврской работа состоит из введения, трёх глав, заключения и списка литературы.

Во введении описывается актуальность проводимого исследования, дается краткая характеристика проделанной работы.

В первой главе приводится обзор методов локализации объектов на изображении.

Во второй главе описывается предложенный алгоритм локализации объектов на изображении на основе анализа их формы.

В третьей главе описывается программная реализация предложенных подходов. Приводятся примеры работы предложенного алгоритма.

В заключении представлены выводы по проделанной работе.

В работе присутствуют 88 таблиц, 88 рисунков. Список литературы состоит из 20 литературных источников. Общий объем выпускной квалификационной работы составляет 40 страниц.

ABSTRACT

The subject of bachelor's work: "Localization of objects in the image based on the analysis of their forms."

In this bachelor's work, methods of localizing objects in an image are investigated by analyzing their forms.

Algorithm of localized objects in the image, based on the selection of the Canny method; heuristic closure of all open circuits; smoothing contours; saving all contours to the collection; search in the collection of contours that meet the specified standard; display of the found contours of the image surfaces.

In the preparation of studies was published 1 article.

Bachelor's work consists of introduction, three chapters, conclusion and list of references.

The introduction describes the relevance of the study, provides a brief description of the work done.

The first chapter reviews methods for localizing objects in an image.

The second chapter describes the proposed algorithm for localizing objects in an image based on an analysis of their shape.

The third chapter describes the software implementation of the proposed approaches. Examples of the operation of the proposed algorithm are given.

In the conclusion presents conclusions on the work done.

In the work there are 88 tables, 88 figures. References consists of 20 references. The total amount of final qualifying work is 40 pages.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	5
1 АНАЛИЗ СОСТОЯНИЯ ВОПРОСА.....	7
2 РАЗРАБОТКА АЛГОРИТМА ЛОКАЛИЗАЦИИ ОБЪЕКТОВ НА ИЗОБРАЖЕНИИ НА ОСНОВЕ АНАЛИЗА ИХ ФОРМЫ	10
2.1 Описание алгоритма	10
2.2 Математическая модель алгоритма.....	12
3 ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ ПРЕДЛОЖЕННЫХ РЕШЕНИЙ	27
3.1 Программная реализация алгоритма.....	27
3.2 Тестирование алгоритма	28
ЗАКЛЮЧЕНИЕ	32
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ	36

ВВЕДЕНИЕ

Системы компьютерного зрения используются в различных областях науки и техники (анализ медицинских изображений, создание беспилотных транспортных средств, в составе графических редакторов изображений, считывание QR и штрих-кодов и т.д.) при решении задач локализации объектов на изображении.

Большинство современных алгоритмов локализации объектов на изображении, применяемых на практике, основано на использовании технологий машинного обучения (сверточных нейронных сетей, деревьев решений, индуктивных алгоритмов построения классификаторов). Однако для применения данных технологий требуется накопление большой коллекции изображений, на которых алгоритмы будут обучаться. Это не всегда возможно реализовать на практике, поэтому актуальной задачей остается развитие алгоритмов компьютерного зрения основанных на классическом (без использования машинного обучения) анализе решаемой задачи.

В настоящее время вопросами локализации объектов на изображениях занимаются такие современники, как Ferdinand Hahmann, Heike Ruppertshofen, Gordon Böer, Ralf Stannarius, Hauke Schramm, Vladimir Reilly, Berkan Solmaz, Mubarak Shah, Dipankar Das, Yoshinori Kobayashi, Yoshinori Kuno, Alexander Oliver Mader, Cristian Lorenz, Martin Bergtholdt, Jens Berg, Hauke Schramm, Jan Modersitzki, Carsten Meyer, Patrick Sudowe, Bastian Leibe и др. [1-20].

В ходе исследований был разработан алгоритм локализации объектов на изображении на основе анализа их формы. Алгоритм включает в себя подготовку изображения к анализу. Сюда входит преобразование цветовой модели из цветовой модели RGB в Gray, что позволяет снизить размерность признакового пространства и увеличить скорость работы предложенного алгоритма. Далее следует применение фильтра Гаусса, необходимое для сглаживания цифровых шумов на изображении. Для выделения границ

объектов в алгоритме предусмотрено расчет градиента каждой группы пикселей изображения. Направление градиента квантируется по 45 градусов. Затем, с учетом наиболее часто встречаемого направления вектора градиента осуществляется подавление пикселей, определенных как границы объектов. Путем выполнения двойной пороговой фильтрации осуществляется подавление границ с низкой величиной градиента и усиление границ со средними и высокими значениям величин градиента.

Также алгоритм осуществляет замыкание незамкнутых контуров за счет использования операций морфологического преобразования (дилатации и эрозии). Затем осуществляется упрощение замкнутых контуров, путем снижения количества опорных точек с применением алгоритма Рамера-Дугласа-Пекера. На заключительном этапе работы алгоритма оценивается форма замкнутых упрощенных контуров объектов. Осуществляется поиск тех контуров, которые удовлетворяют заданным критериям отбора.

Целью работы является разработка алгоритма локализация объектов на изображении на основе анализа их формы.

В ходе выполнения исследований была разработана программа, реализующая предложенный алгоритм. Предложенные подходы были протестированы на тестовых изображениях.

В ходе выполнения бакалаврской работы было опубликовано статья по теме разработки алгоритма локализация объектов на изображении на основе анализа их формы.

Основные результаты работы были доложены на V Международной научно-практической конференции (школы-семинара) молодых ученых «Прикладная математика и информатика: современные исследования в области естественных и технических наук»

1 АНАЛИЗ СОСТОЯНИЯ ВОПРОСА

Локализация объектов с точки зрения систем компьютерного зрения – это процесс поиска всех искомым объектов и определение места их расположения на изображении.

Решение задачи локализации является одной из самой часто решаемых задач разработчиками систем компьютерного зрения. Локализация объектов является первым этапом анализа изображений в системах распознавания образов. Так, например, локализация автомобильных номеров необходима перед началом процесса распознавания, содержащихся на нем символов. Перед декодированием QR кода, необходимо определить, где в какой области изображения, полученной с камеры телефона он находится. В программном обеспечении камер мобильных телефонов встроена возможность автофокусировки изображения на лицах людей (т.е. решается задача локализации лиц на изображениях).

Развитие технологий искусственного интеллекта позволило разработчикам систем компьютерного зрения разрабатывать алгоритмы локализации объектов на изображении, основанные на машинном обучении. Такой подход позволил снизить трудоемкость создания алгоритмов за счет автоматизации настройки параметров математических моделей под решаемую задачу.




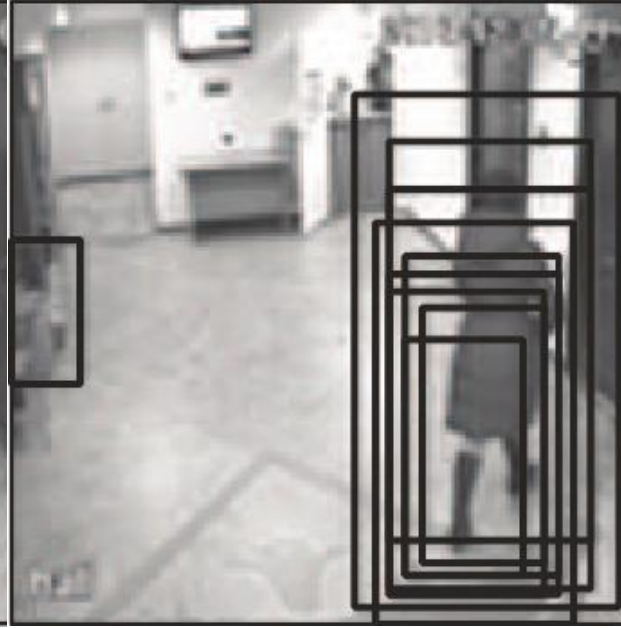
К технологиям машинного обучения, применяемым для локализации объектов на изображении, относятся:

- сверточные нейронные сети;
- метод Виолы-Джонса (поиск каскада слабых классификаторов, основанных на признаках Хаара);
- метод HOG SVM (признаковое пространство HOG и метод опорных векторов);
- метод HOG+Decision Tree (признаковое пространство HOG и дерево принятия решений);

- метод GNG-FIS (нейронный газ и нечеткий вывод).

Пример сравнения работы разных алгоритмов в задаче локализации людей на изображении представлен в таблице 1.

Таблица 1 – Сравнение разных методов машинного обучения в задаче локализации людей на изображениях

Сверточные нейронные сети	Метод Виолы-Джонса
	
Метод HOG SVM	Метод GNG-FIS
	

Как видно из таблицы 1, различные алгоритмы локализации объектов основанные на машинном обучении по разному справляются с задачей

локализации людей. Точность работы алгоритмов во многом зависит от успешности настройки математической модели на этапе обучения. Принцип обучения заключается в накоплении большого количества изображений (обучающую выборку) содержащих искомый объект и изображений где искомый объект отсутствует. При этом изображения должны быть дополнены метаданными, указывающими, на каком изображении и в каком месте находится искомый объект. Так результаты представленные на в таблице 1 получены на обучающей выборке размерностью 2100 изображений. Точность работы алгоритмов локализации можно повысить, увеличив размерность обучающей выборки.

Однако для применения данных технологий требуется накопление большой коллекции изображений, на которых алгоритмы будут обучаться. Это не всегда возможно реализовать на практике, поэтому актуальной задачей остается развитие алгоритмов компьютерного зрения основанных на классическом (без использования машинного обучения) анализе решаемой задачи. При этом во многих случаях идентифицировать объекты на изображении можно путем оценивания их формы.

Поэтому актуальной стоит признать цель исследования – разработка алгоритма локализация объектов на изображении на основе анализа их формы.

Данную цель предполагается достигнуть путем решения следующих задач:

1. Провести анализ текущего развития алгоритмов локализации объектов на изображении.
2. Разработать алгоритм локализации объектов на изображении, основанных на анализе их формы
3. Разработать программное обеспечение, реализующее данный алгоритм.
4. Протестировать программную реализацию на реальных изображениях.

2 РАЗРАБОТКА АЛГОРИТМА ЛОКАЛИЗАЦИИ ОБЪЕКТОВ НА ИЗОБРАЖЕНИИ НА ОСНОВЕ АНАЛИЗА ИХ ФОРМЫ

2.1 Описание алгоритма

Предложенный алгоритм локализация объектов на изображении на основе анализа их формы состоит из четырех этапов (рисунок 2.1).

Первый этап заключается в подготовке изображения к анализу. Сюда входит преобразование цветовой модели из цветовой модели RGB в Gray, что позволяет снизить размерность признакового пространства и увеличить скорость работы предложенного алгоритма. Также к этому этапу относится применение фильтра Гаусса, необходимое для сглаживания цифровых шумов на изображении, вносимых алгоритмами сжатия изображения (например, JPEG) и сенсорами (матрицей веб-камеры).

Вторым этапом алгоритма является выделение границ объектов на изображении. Данный этап состоит из последовательного выполнения следующих шагов. Сначала осуществляет поиск для каждого пикселя изображения величины и направления градиента. Направление градиента квантируется по 45 градусов. Затем, с учетом наиболее часто встречаемого направления вектора градиента осуществляется подавление пикселей, определенных как границы объектов. Путем выполнения двойной пороговой фильтрации осуществляется подавление границ с низкой величиной градиента и усиление границ со средними и высокими значениям величин градиента.

На третьем этапе осуществляет замыкание незамкнутых контуров за счет использования операций морфологического преобразования (дилатации и эрозии). Затем осуществляет упрощение замкнутых контуров, путем снижения количества опорных точек с применением алгоритма Рамера-Дугласа-Пекера.

На заключительном этапе оценивается форма замкнутых упрощенных контуров объектов. Осуществляется поиск тех контуров, которые удовлетворяют заданным критериям отбора.

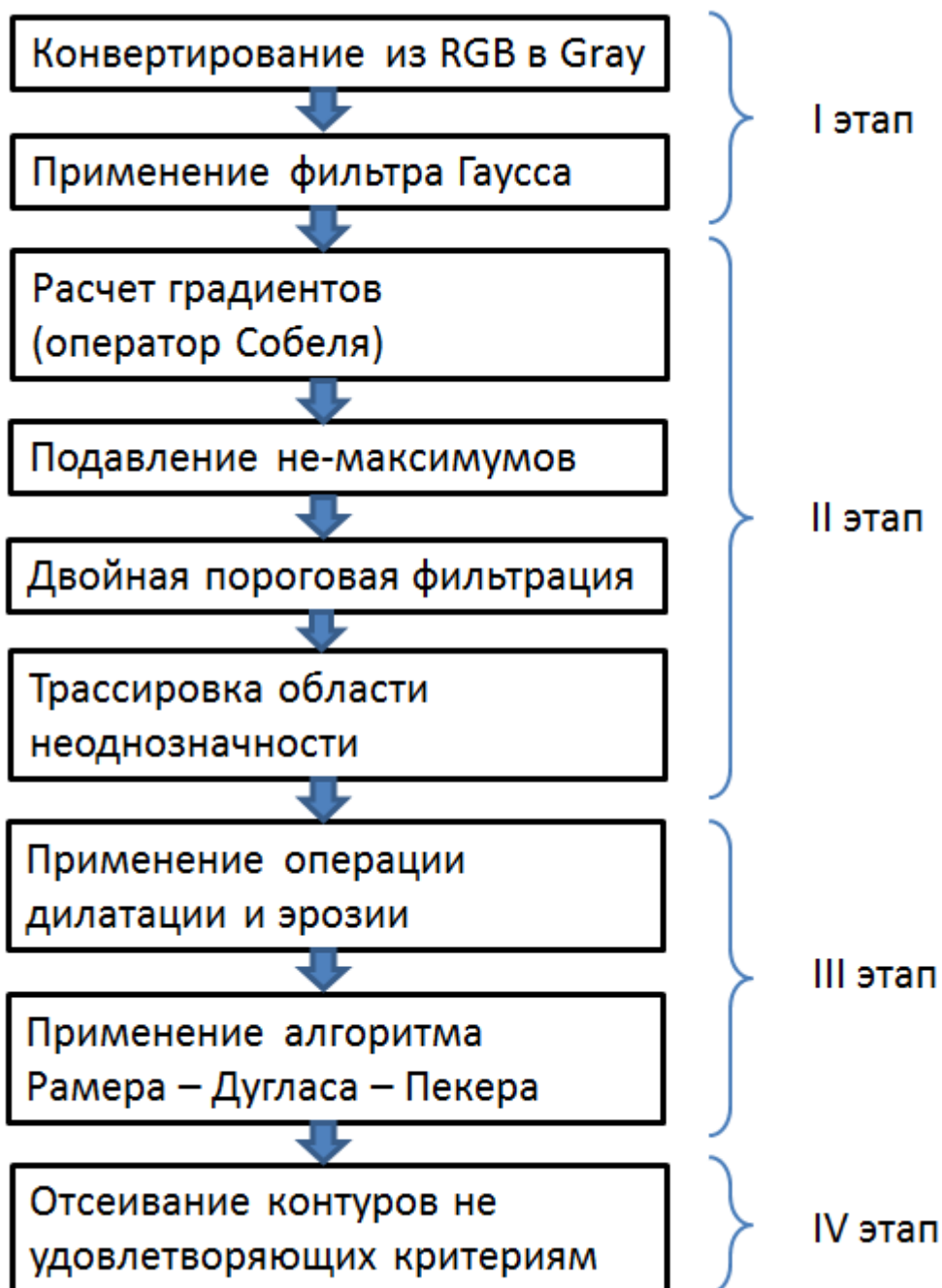


Рисунок 2.1 – Алгоритм локализация объектов на изображении на основе анализа их формы

2.2 Математическая модель алгоритма

Для простоты изложения, принципы предложенного метода будут рассматриваться на примере задачи поиска и локализации книг на изображении.

На рисунке 1 представлен пример изображения, на котором будет проводиться апробация предложенного метода. Как видно, на изображении и присутствуют две книги (прямоугольной формы), которые необходимо локализовать, а также другие объекты различных форм (геймпад, мышь и т.д.).

Для того что бы оценить форму объектов присутствующих на изображении необходимо выполнить два основных действия: выделить контуры объектов, провести анализ найденных контуров для оценки соответствия их геометрии заданным критериям.

Изображения могут браться из файлов или с веб-камеры. При этом изображении состоит из пикселей, цвет которых закодирован в цветовой модели RGB. В соответствии с данной цветовой моделью цвет состоит из трех каналов – красного, зеленого и синего. Тогда анализируемое изображение можно представить в виде массива $pix(x,y)$ (2.1)

$$pix(x, y) = \begin{pmatrix} (R_{1,1}; G_{1,1}; B_{1,1}) & \dots & (R_{m,1}; G_{m,1}; B_{m,1}) \\ \dots & \dots & \dots \\ (R_{1,n}; G_{1,n}; B_{1,n}) & \dots & (R_{m,n}; G_{m,n}; B_{m,n}) \end{pmatrix}, \quad (2.1)$$

где x и y – координаты пикселей изображения ($x=1 \dots m$, $y=1 \dots n$), m – ширина изображения в пикселях, n – высота изображения в пикселях, R , G , B – компоненты цвета пикселей ($R \in [0; 255]$, $G \in [0; 255]$, $B \in [0; 255]$).

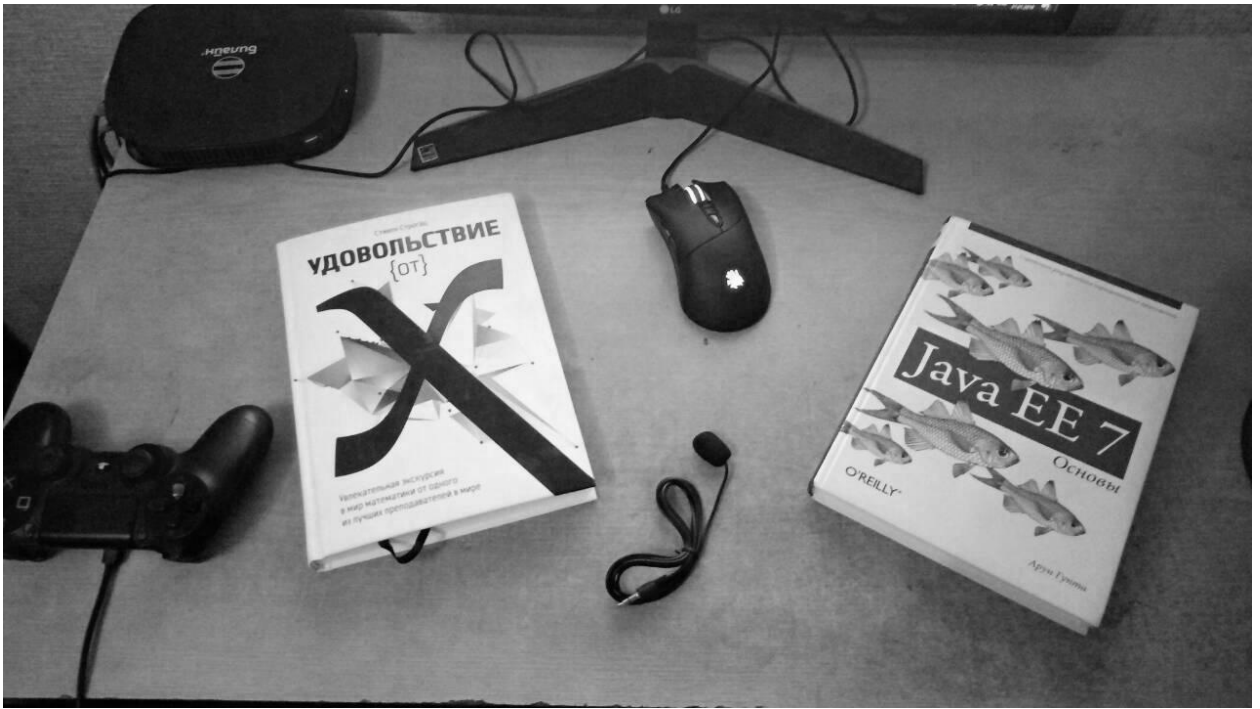


Рисунок 2.3 – Изображение $pix_gray(x,y)$

Стоит отметить, что обычно цифровые изображения получают с помощью фотосенсоров, которыми оснащены web-камеры и фотокамеры. Фотосенсоры искажают изображения, внося в них дефекты, называемые цифровым шумом. При анализе изображений цифровой шум вносит значительную погрешность в результаты работы алгоритмов машинного зрения. Поэтому цифровые шумы необходимо сглаживать, используя, например, фильтр размытия по Гауссу.

Применение фильтра заключается в выполнении следующих вычислений (2.3)

$$pix_gauss(x, y) = w * pix_gray(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b [w(s, t) \times pix_gray(x - s, y - t)] \quad (2.3)$$

где w – матрица свертки, $pix_gray(x,y)$ – исходное изображение к которому применяется матрица свертки, $pix_gauss(x,y)$ – изображение с результатом применения матрицы свертки, a, b – коэффициенты, которые зависят от

размеров матрицы ядра свертки (для матрицы свертки 3 на 3: $a=1, b=1$, для матрицы 5 на 5: $a=2, b=2$ и т.д.)

Для ускорения расчётов используют матрицы, значения которых заранее рассчитанные по формуле нормального распределения Гаусса (2.4):

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad (2.4)$$

где μ – математическое ожидание, σ – среднеквадратическое отклонение.

Таким образом, матрица свертки фильтра Гаусса с размерами 3 на 3 выглядит следующим образом (2.5):

$$w = \frac{1}{16} \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix} \quad (2.5)$$

Для более сильного размытия изображения можно использовать фильтр Гаусса большего размера, например 5 на 5 (2.6):

$$w = \frac{1}{256} \begin{pmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{pmatrix} \quad (2.6)$$

Пример применения фильтра Гаусса к изображению представлен на рисунке 2.4.

Так как необходимо оценивать геометрическую форму объектов необходимо выделить и подчеркнуть края объектов на изображении. Самым популярным методом выделения границ является – детектор границ Канни (Canny).

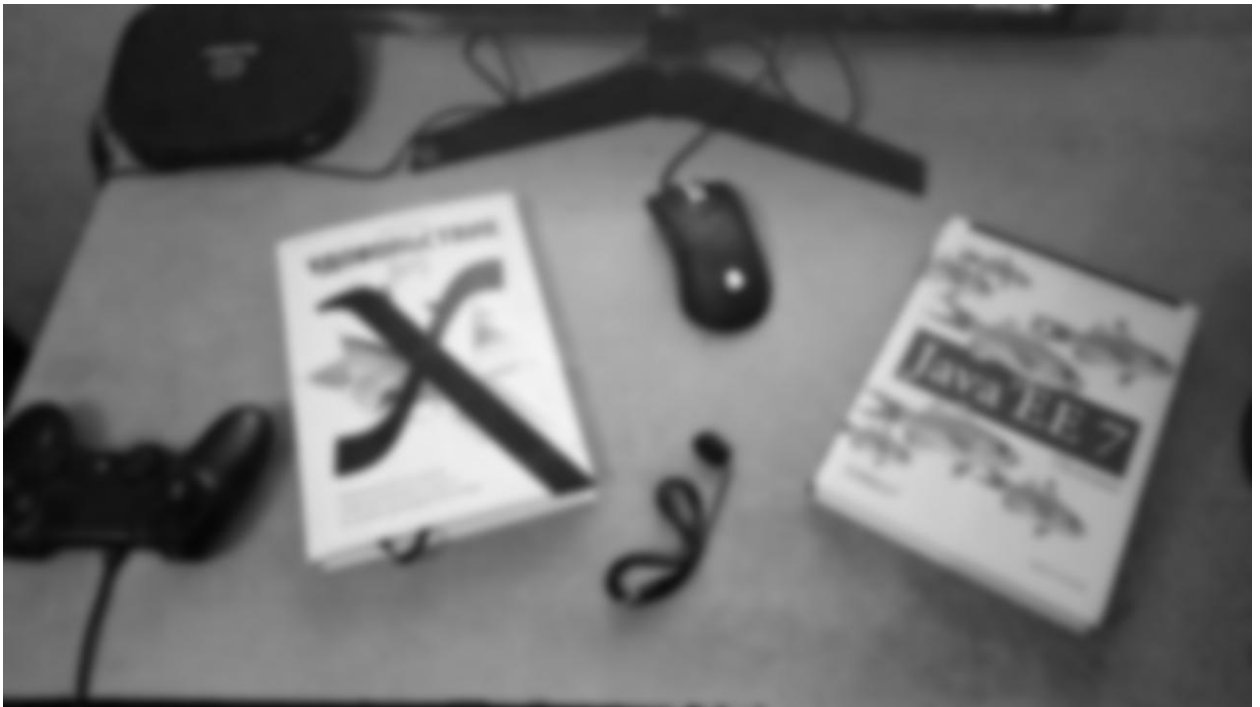


Рисунок 2.4 – Изображение $pix_gauss(x,y)$

Работа метода заключается в поиске кривых на изображениях вдоль которых происходит резкое изменение яркости пикселей. Причинами резкого изменения яркостей пикселей являются:

- изменение условий освещенности на изображении;
- разный цвет пикселей находящихся рядом с друг другом;
- изменение глубины сцены (разная ориентация поверхности, на которой находятся объекты изображения).

Детектор границ Канни определяет места резкого изменения яркости пикселей с помощью градиента. При программной реализации расчета градиента с точки зрения оптимизации применяют дискретный дифференциальный оператор Собеля, позволяющий вычислять приближенное значение градиента.

При этом оператор Собеля основан на свертке изображения целочисленными фильтрами в двух направлениях – вертикальном и горизонтальном.

Ядро горизонтального фильтра MG_x выглядит как (2.7):

$$MGx = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \quad (2.7)$$

При этом ядро вертикального фильтра MGy будет таким (2.8):

$$MGy = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} \quad (2.8)$$

Расчет значений градиента осуществляется для каждой позиции пикселя с помощью двух вложенных циклов. Первый цикл с индексом iX управляет за перемещение по оси X, второй цикл iY отвечает за перемещение по оси Y (2.9):

$$\begin{aligned} & \text{for } iY \in 2 \dots \text{rows}(\text{pix_gauss}) - 2 \text{ do} \\ & \quad \text{for } iX \in 2 \dots \text{cols}(\text{pix_gauss}) - 2 \text{ do ,} \\ & \quad \quad \langle \text{расчет градиента} \rangle \end{aligned} \quad (2.9)$$

где функция $\text{rows}()$ определяет высоту исходного изображения в пикселях, а функция $\text{cols}()$ – ширину изображения в пикселях.

Внутри цикла $\langle \text{расчет градиента} \rangle$ для расчета выполняются следующие действия. На первом этапе происходит выделение фрагмента part_pix_gauss изображения pix_gauss размером 3 на 3 пикселя. Для этого используется функция $\text{submatrix}()$ (2.10):

$$\text{part_pix_gauss} = \text{submatrix}(\text{pix_gauss}, iY - 1, iY + 1, iX - 1, iX + 1) \quad (2.10)$$

Затем рассчитывается значение градиента GX в горизонтальном направлении, так как это показано на (2.11):

$$GX = \sum_{y=1}^3 \sum_{x=1}^3 (\text{Part_pix_gauss}(x, y) \cdot MGx(x, y)) \quad (2.11)$$

Аналогичным образом рассчитывается значение градиента GY в вертикальном направлении (2.12)

$$GY = \sum_{y=1}^3 \sum_{x=1}^3 (\text{Part_pix_gauss}(x, y) \cdot MGy(x, y)) \quad (2.12)$$

Теперь с учетом (2.11) и (2.12) можно рассчитать значение градиента с учетом горизонтальной и вертикальной составляющей следующим образом (2.13):

$$G = \sqrt{Gx^2 + Gy^2} \quad (2.13)$$

Угол вектора градиента квантируется по 45 градусов, т.е. все значение углов округляются до ближайшего значения из ряда (0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°). Это необходимо для последующей фильтрации изображения с целью более точного выделения границ объектов (2.14):

$$\theta = \text{round}\left(\frac{\arctan 2(Gx, Gy)}{\pi / 4}\right) \cdot \frac{\pi}{4} - \frac{\pi}{2} \quad (2.14)$$

На этом заканчивается текущая итерация цикла *<расчет градиента>* и осуществляется переход к следующей итерации с другой парой значений *iX* и *iY*.

После завершения выполнения цикла мы получаем две матрицы: первая содержит значения градиента для каждого пикселя изображения (2.15), а вторая угол вектора градиента для каждого пикселя изображения (2.16).

$$\text{pix_edge}(iX, iY) = G \quad (2.15)$$

$$\text{pix_G_angle}(iX, iY) = \theta \quad (2.16)$$

Следующим этапом фильтрации изображения является подавление не-максимумов. Его идея заключается в том, что границами объявляются только те пиксели, в которых достигается максимум градиента в направлении вектора градиента.

На рисунке 2.5 показан принцип действия подавления не максимумов. При этом сначала определяется наиболее часто встречающееся направление градиента (на показанном примере это угол 0°). При данном направлении вектора осуществляется поиск максимальных значений градиента в вертикальном направлении (в каждом столбце пикселей). В примере пиксели

с максимальным значением градиента показаны белой рамкой. Эти максимальные значения градиента остаются, а остальные подавляются.

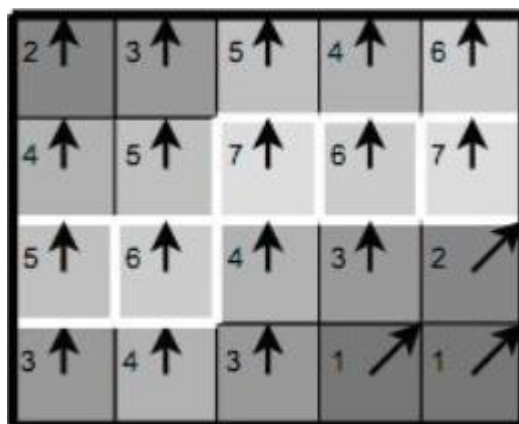


Рисунок 2.5 – Подавление не-максимумов (стрелками показано направление векторов), цифры – значение градиента.

Следующим шагом выделения границ является двойная пороговая фильтрация. Верхний порог предназначен для маркировки пикселей, у которых значение градиента выше значения порога, как пиксель границы. Нижний порог предназначен для маркировки пикселей, у которых значение градиента ниже значения данного порога, как пиксель не являющейся границей. Пиксели, значения которых лежат между двумя порогами маркируются, как неоднозначные и, впоследствии, будут подвергнуты дополнительному анализу.

Заключительным этапом выделения границ объектов является трассировка области неоднозначности. Те пиксели, которые на предыдущем этапе были промаркированы как неоднозначные перемаркируются, как пиксели границы, если они касаются в одном из восьми направлений пикселей границы. В противном случае они подавляются.

Результат использования детектора границ Канни представлен на рисунке 2.6

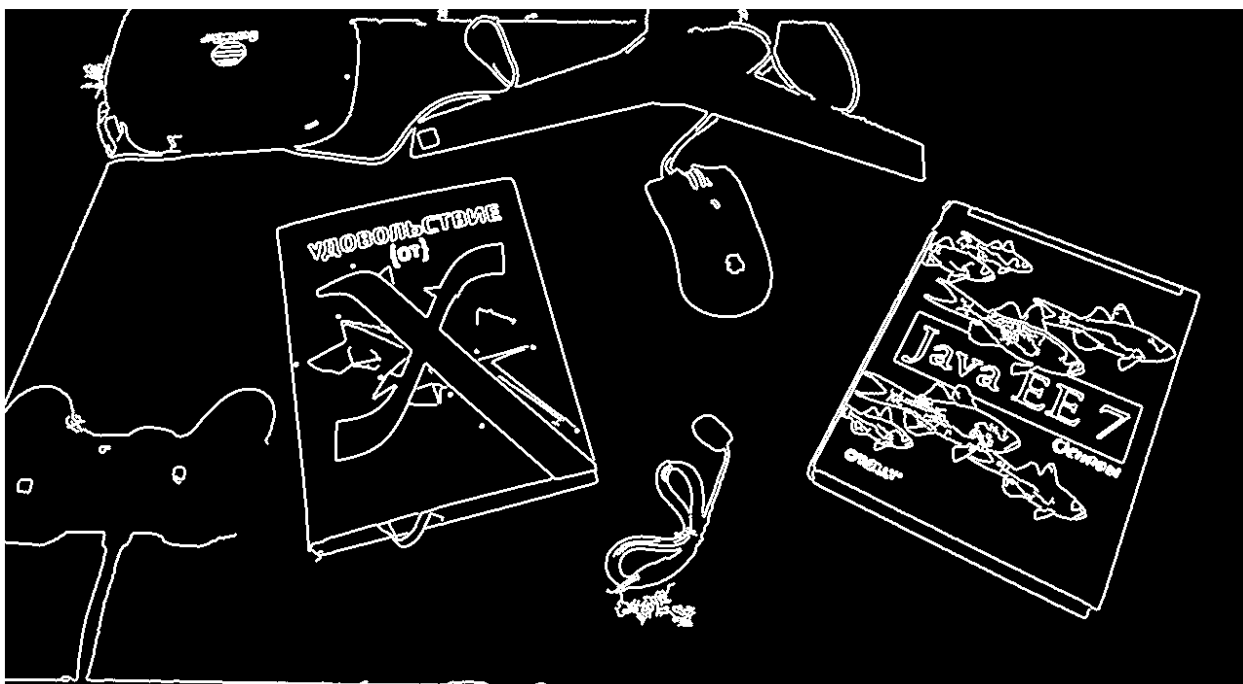


Рисунок 2.6 – Изображение $pix_edge(x,y)$

Как видно из рисунка 2.6, границы объектов представляют собой в основном разомкнутые контуры. Так как планируется анализировать замкнутые контуры, то необходимо выполнить процедуру замыкания контуров.

Проблему наличия разомкнутых контуров можно решить двумя способами:

1. Подобрать такие параметры детектора границ, при которых количество разомкнутых контуров минимально. На это влияют в основном значения верхнего и нижнего порогов на этапе двойной фильтрации границ.
2. Сформировать закрытые контуры с использованием морфологических преобразований.

Так как использование первого способа более трудоемко из-за необходимости ручного подбора параметров, остановимся на втором способе.

Морфологические преобразования - это простые операции, основанные на форме изображения. Обычно это выполняется на двоичных изображениях. Он требует двух входов, один из которых является нашим исходным изображением, второй называется структурирующим элементом или ядром, которое определяет характер операции. Два основных морфологических оператора - это эрозия и дилатация. Другие операции – открытие, закрытие, градиент и др. являются, по сути, вариациями использования эрозии и дилатации.

В нашем случае, в качестве ядра для операции закрытия будет использована матрица *kernel* следующего вида:

$$kernel = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \quad (2.17)$$

Операция закрытия является – применение операции дилатации с последующим применением к результату операции эрозии с тем же ядром *kernel*. После выполнения данных действий промежутки в контурах удаляются.

Результат выполнения операции закрытия представлен на рисунке 2.7.

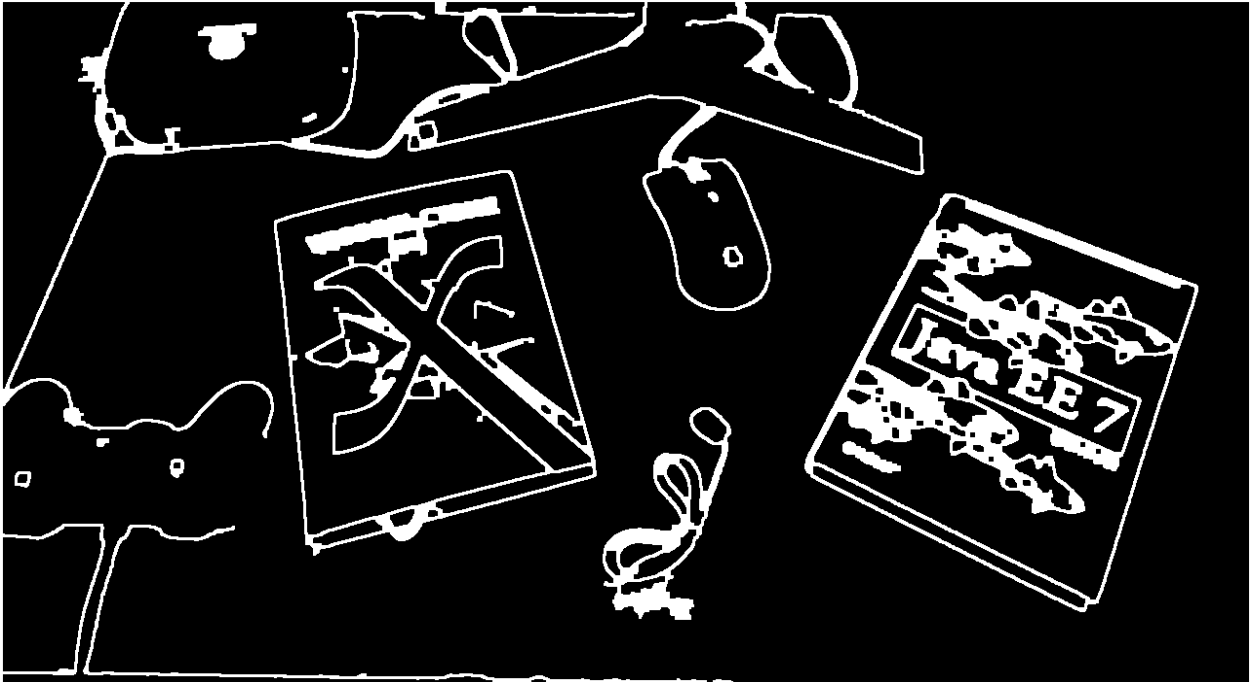


Рисунок 2.7 – Изображение $pix_closed(x,y)$ как результат морфологического преобразования (операция закрытия)

Следующим этапом анализа изображения будет процедурный поиск всех замкнутых контуров и формирование массива обнаруженных замкнутых контуров. На языке Python запуск данной процедуры осуществляется методом `findContours()`: `(cnts, _) = cv2.findContours(...)`. При этом результатом выполнения данной операции будет коллекция контуров, хранящаяся в переменной `cnts`.

Теперь, когда у нас в `cnts` хранятся все найденные контуры можно переходить к их анализу.

Например, мы хотим найти на изображении все книги. Очевидно, что книги имеют прямоугольную форму. Тогда задача поиска таких объектов сводится к поиску контуров с четырьмя углами.

Таким образом, в цикле производим проверку каждого контура из массива `cnts` на наличие четырех углов:

```
for c in cnts:
```

```
<действия, выполняемые над каждым контуром>
```

Очень редко контур представляет собой идеальный прямоугольник. Это связано с тем, что на изображении могут присутствовать шумы и тени. По этой причине перед анализом формы контура его необходимо сглаживать, причем сила *epsilon* сглаживания должна зависеть от масштаба контура (2.18):

$$\epsilon(per_i) = 0,02 \cdot per_i \quad (2.18)$$

Таким образом, сначала производится расчет длины *peri* контура, а затем применяется процедура по его сглаживания.

Под процедурой сглаживания понимается применение алгоритма Рамера – Дугласа – Пекера. Суть алгоритма состоит в том, чтобы по данной ломаной, аппроксимирующей кривую, построить ломаную с меньшим числом точек. Алгоритм определяет расхождение, которое вычисляется по максимальному расстоянию между исходной и упрощённой кривыми. Упрощенная кривая состоит из подмножества точек, которые определяются из исходной кривой.

Начальная кривая представляет собой упорядоченный набор точек или линий, и заданное расстояние $\epsilon > 0$.

Алгоритм рекурсивно делит линию. Входом алгоритма служат координаты всех точек между первой и последней. Первая и последняя точка сохраняются неизменными. После чего алгоритм находит точку, наиболее удалённую от отрезка, соединяющего первую и последнюю. Если точка находится на расстоянии, меньшем s , то все точки, которые ещё не были отмечены к сохранению, могут быть выброшены из набора и получившаяся прямая сглаживает кривую с точностью не ниже *epsilon*.

Если же расстояние больше *epsilon*, то алгоритм рекурсивно вызывает себя на наборе от начальной до данной и от данной до конечной точек (что означает, что данная точка будет отмечена к сохранению).

По окончании всех рекурсивных вызовов выходная ломаная строится только из тех точек, что были отмечены к сохранению.

Псевдокод применение алгоритма Рамера – Дугласа – Пекера представлен на рисунке 2.8.

```
function DouglasPeucker(PointList[], epsilon)
  //находим точку с максимальным расстоянием между
  //первой и последней точками набора
  dmax = 0
  index = 0
  for i = 2 to (length(PointList) - 1)
    d = PerpendicularDistance(PointList[i], Line(PointList[1], PointList[end]))
    if d > dmax
      index = i
      dmax = d
    end
  end
  //если максимальная дистанция больше, чем epsilon, то рекурсивно
  //вызываем ее на участках
  if dmax >= epsilon
    //Recursive call
    recResults1[] = DouglasPeucker(PointList[1...index], epsilon)
    recResults2[] = DouglasPeucker(PointList[index...end], epsilon)
    // Строим итоговый набор точек
    ResultList[] = {recResults1[1...end-1] recResults2[1...end]}
  else
    ResultList[] = {PointList[1], PointList[end]}
  end
  // Возвращаем результат
  return ResultList[]
end
```

Рисунок 2.8 – Псевдокод алгоритма Рамера – Дугласа – Пекера.

Пример работы алгоритма Рамера – Дугласа – Пекера представлен на рисунке 2.9.

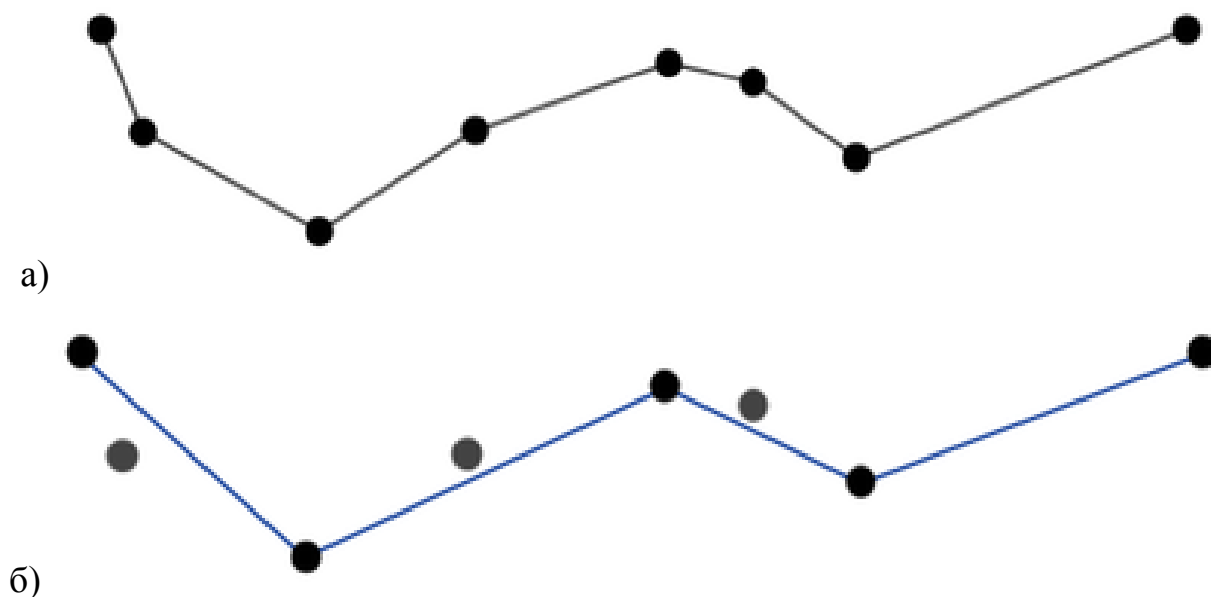


Рисунок 2.9 – Пример сглаживания контура алгоритмом Рамера – Дугласа – Пекера: а – исходный контур; б – сравнение исходного контура и результирующего контура

Так как в результате работы алгоритма Рамера – Дугласа – Пекера количество опорных точек уменьшается, то, для определения является ли контур прямоугольником достаточно подсчитать их количество. Если опорных точек у контура равно 4, то данный контур является искомым.

Для наглядности все контуры, которые соответствуют заданным критериям можно отображать поверх исходного изображения. Результат анализа изображения представлен на рисунке 2.10.

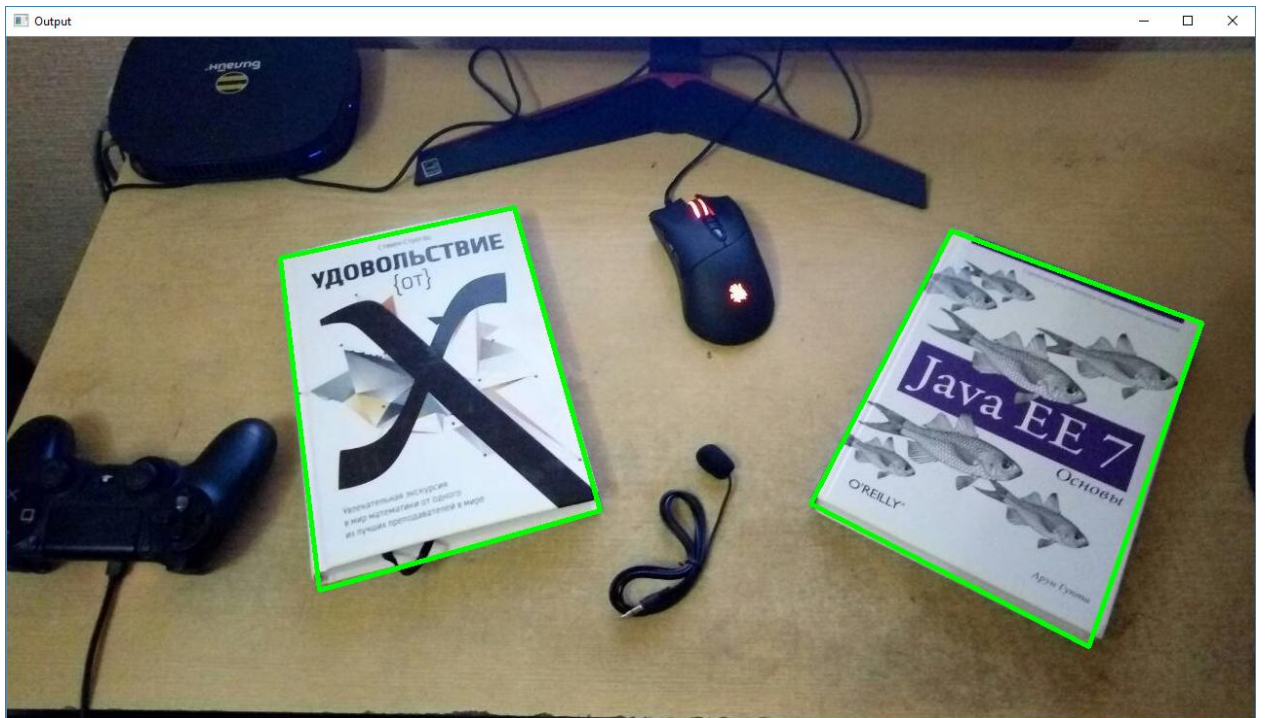


Рисунок 2.10 – Результат анализа исходного изображения

Как видно из рисунка 2.10, предложенный в рамках данного исследования подход позволяет решать задачу локализации объектов на изображении за счет анализа их формы.

В заключении стоит отметить, что предложенный алгоритм позволяет использовать другие параметры отбора контуров объектов или их сочетание.

Например, можно учитывать:

- размер объект в виде (в виде площади и периметра);
- соотношение длин сторон;
- расположение контура на изображении;
- поворот контура относительно вертикали или горизонтали;
- другие параметры и их сочетания.

3 ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ ПРЕДЛОЖЕННЫХ РЕШЕНИЙ

3.1 Программная реализация алгоритма

При выполнении бакалаврской работы на языке Python было разработано программное обеспечение, реализующее предложенный алгоритм локализации объектов на изображении (рисунок 3.1).

Функциональные особенности программного обеспечения:

- наличия графического интерфейса;
- возможность загрузки изображений из файлов формата png и jpg;
- программная реализация всех шагов предложенного алгоритма: конвертирование цветовой модели в формат Gray, применения фильтра Гаусса, выделение границ объектов, замыкание контуров на изображении, анализ контуров на соответствие заданным критериям, выделение границ найденных объектов;
- Отображение результатов выполнения каждого шага алгоритма.

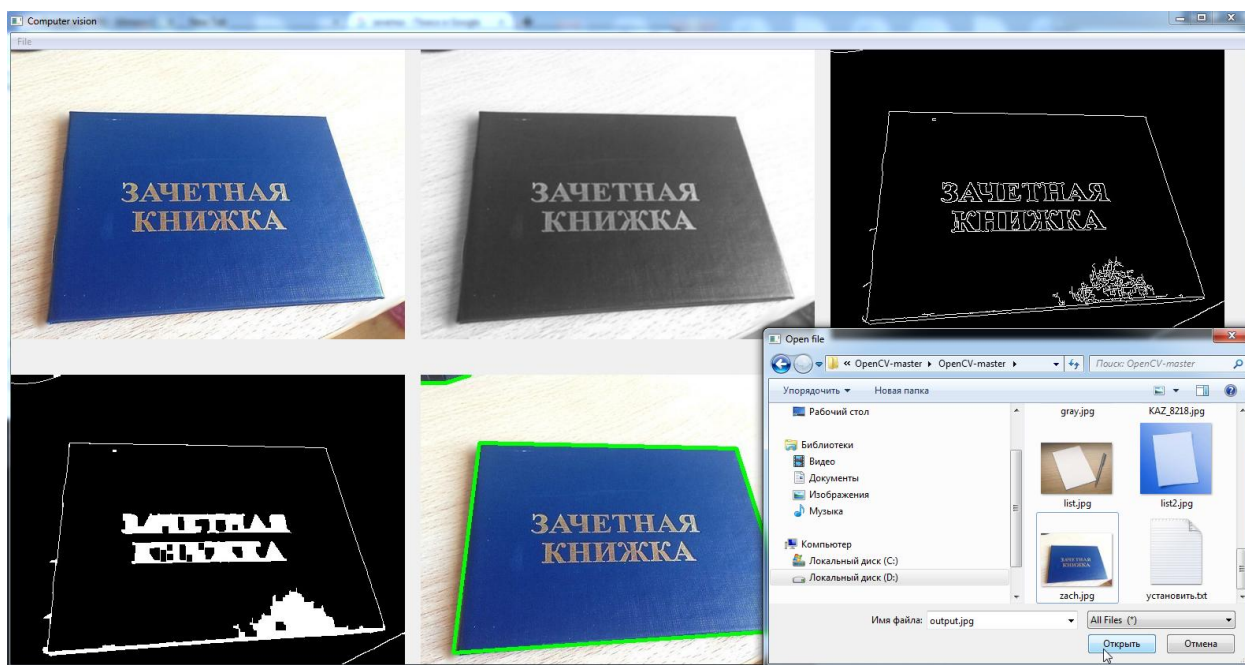


Рисунок 3.1 – Главное окно программы и окно для выбора анализируемого файла

3.2 Тестирование алгоритма

Проверка работы алгоритма проводилась на трех изображениях. Первое изображение содержит в себе фрагмент стола, на котором расположены книги, кофейные чашки, подстаканники и конфета. Это изображение очень контрастное и четкое, поэтому алгоритм локализации успешно справляется с задачей поиска прямоугольных объектов (таблица 2).

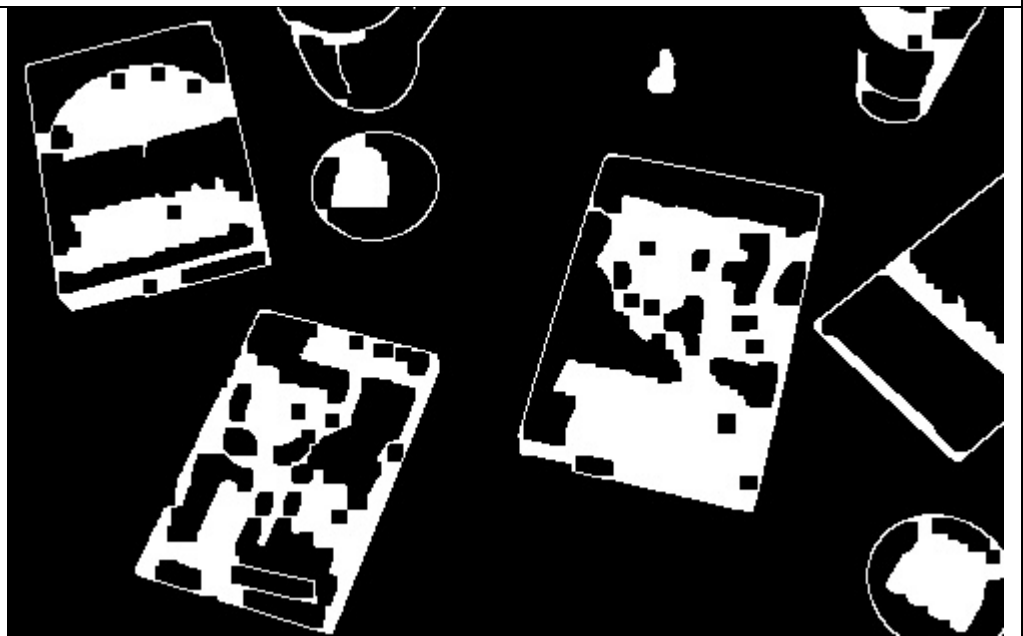
Таблица 2 – Тестовое изображение «Книги на контрастном фоне, высокое качество фото»

Название этапа	Изображение
Загрузка изображения	 The image shows a top-down view of a dark table. On the table are several items: a white mug, a Starbucks coffee cup, two brown coasters with the 'WLANOS' logo, and four books. The books are 'ZONGA' by Michael Crichton, 'POST OFFICE' by Charles Bukowski, 'PREY' by Michael Crichton, and 'BEING TEA' by The Sunday Morning Post. The image is sharp and high-contrast.
Перевод цветовой модели в Gray и применение фильтра Гаусса	 This image is a grayscale version of the one above. It has been processed with a Gaussian blur filter, which softens the edges and reduces the sharpness of the objects on the table. The text on the books and coasters is still legible but less crisp than in the original image.

Выделение
границ
объектов



Замыкание
контуров



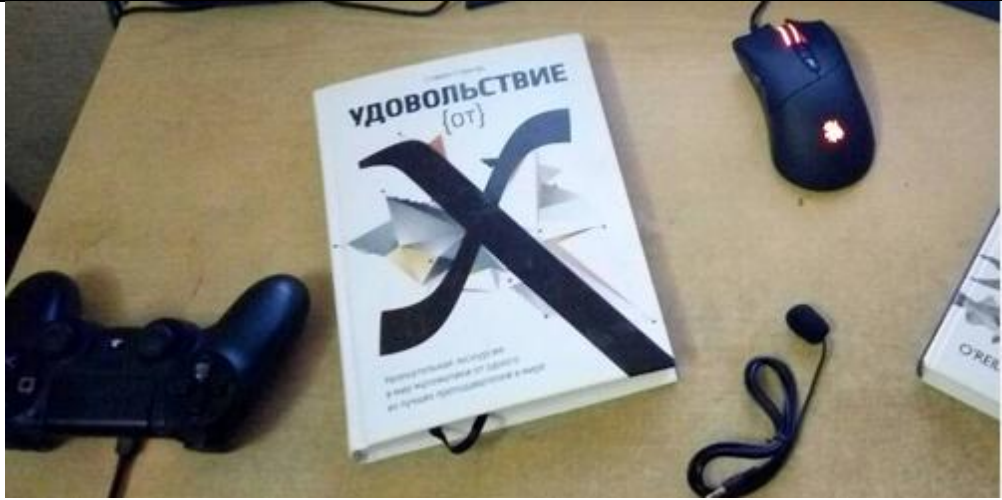
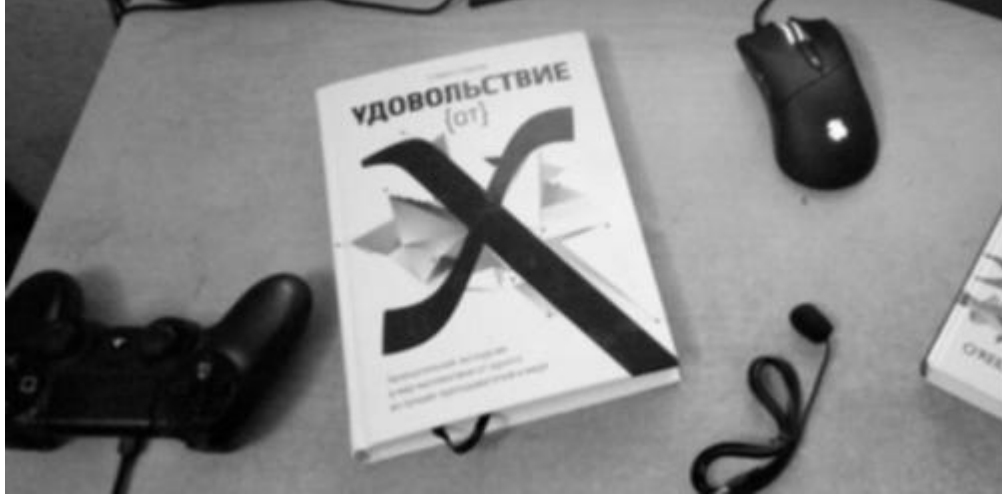
Выделение
найденных
объектов



Следующее изображение снято на камеру телефона с низким разрешением, и оно содержит в себе помимо книг более сложные объекты, такие как компьютерная мышь, геймад, подставка для монитора, микрофон.

Работа алгоритма в данном случае осложнена меньшей контрастностью изображения (светлый стол), слабым освещением и наличием мягких теней. Однако и с этой задачей алгоритм справился успешно, найдя все объекты прямоугольной формы (таблица 3).

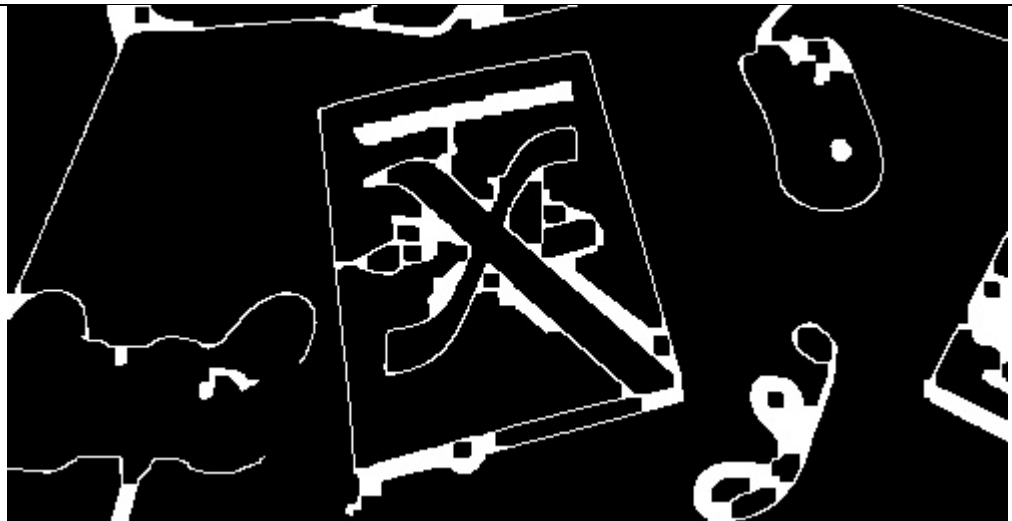
Таблица 3 – Тестовое изображение «Книги на контрастном фоне, низкое качество фото, наличие теней»

Название этапа	Изображение
Загрузка изображения	
Перевод цветовой модели в Gray и применение фильтра Гаусса	

Выделение
границ
объектов



Замыкание
контуров





Выделение
найденных
объектов

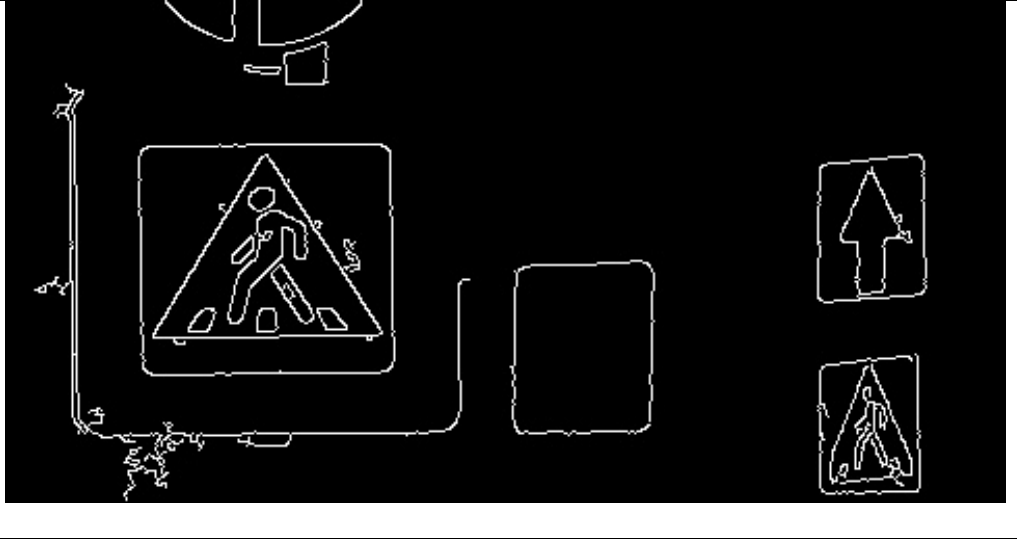
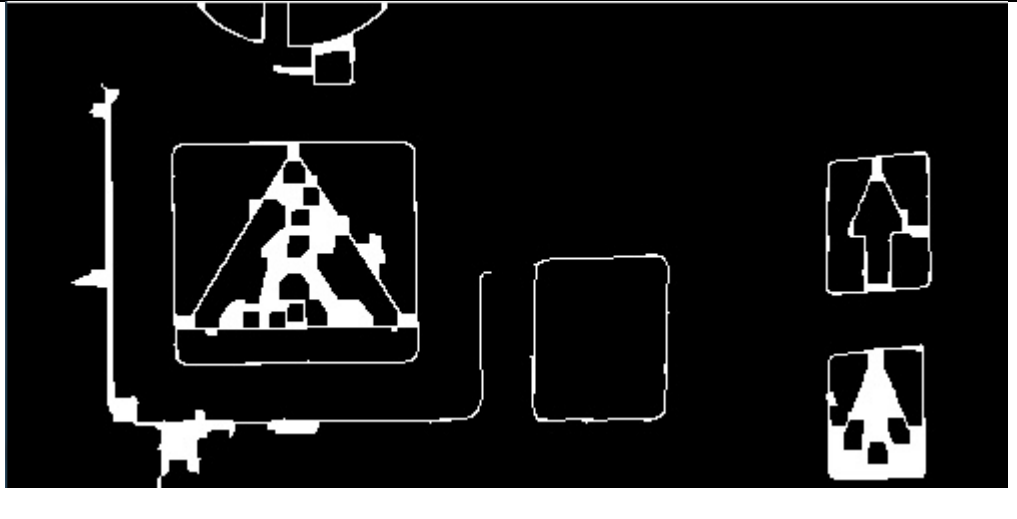



Последним тестовым изображением является фотография дорожных знаков, сделанная во время снегопада. Работа алгоритма осложнена низкой контрастностью изображения знака на переднем плане, размытое изображение знака на заднем плане, а также расположение под углом знаков на среднем плане. Также на изображении присутствует “шум” в виде падающего снега.

В этом случае алгоритм локализации также справился с поставленной задачей и обнаружил все дорожные знаки (Таблица 4).

Таблица 4 – Тестовое изображение «Знаки дорожного движения во время снегопада, низкое качество фото, низкое разрешение»

Название этапа	Изображение
Загрузка изображения	
Перевод цветовой модели в Gray и применение фильтра Гаусса	

<p>Выделение границ объектов</p>	 <p>The image shows the edge detection result of the traffic signs from the original image. The signs are represented by white outlines on a black background. The pedestrian crossing sign is the largest and most prominent, with a smaller square sign to its right and two smaller triangular signs to the right of that. The edges are sharp and clearly defined.</p>
<p>Замыкание контуров</p>	 <p>The image shows the result of closing the contours from the edge detection step. The white outlines are now filled with a checkerboard pattern, representing the closed binary regions of the signs. The shapes are more solid and complete, with the gaps between pixels filled in.</p>
<p>Выделение найденных объектов</p>	 <p>The image shows the original photograph of the traffic signs with green bounding boxes drawn around each of them. The largest bounding box is around the pedestrian crossing sign, and smaller ones are around the other signs, demonstrating the successful localization of the objects.</p>

Как видно из результатов тестирования, предложенный алгоритм локализации объектов на изображении успешно справляется с поставленной задачей. Это на практике доказывает эффективность предложенных решений.

ЗАКЛЮЧЕНИЕ

В ходе выполнения исследований представленных в данной бакалаврской работе были сделаны следующие выводы:

1. Обзор научных литературных источников по теме исследования показал, что современные системы компьютерного зрения используются в различных областях науки и техники (анализ медицинских изображений, создание беспилотных транспортных средств, в составе графических редакторов изображений, считывание QR и штрих-кодов и т.д.) при решении задач локализации объектов на изображении.

2. Обзор научных источников показал, что большинство современных алгоритмов применяемых на практике, основано на использовании таких технологий машинного обучения, как сверточные нейронные сети, деревья решений, индуктивные алгоритмы построения классификаторов (такие как метод Виолы-Джонса). Однако для применения технологий машинного обучения при решении задачи локализации объектов на изображении требуется накопление большой коллекции изображений, на которых алгоритмы будут обучаться. Это не всегда возможно реализовать на практике, поэтому актуальной задачей остается развитие алгоритмов компьютерного зрения основанных на классическом анализе решаемой задачи (без использования машинного обучения).

3. В ходе исследований был разработан алгоритм локализации объектов на изображении на основе анализа их формы. Алгоритм включает в себя подготовку изображения к анализу. Сюда входит преобразование цветовой модели из цветовой модели RGB в Gray, что позволяет снизить размерность признакового пространства и увеличить скорость работы предложенного алгоритма. Далее следует применение фильтра Гаусса, необходимое для сглаживания цифровых шумов на изображении, вносимых алгоритмами сжатия изображения (например, JPEG) и сенсорами (матрицей веб-камеры).

Для выделения границ объектов в алгоритме предусмотрено последовательное выполнение следующих шагов. Сначала осуществляется поиск для каждого пикселя изображения величины и направления градиента. Направление градиента квантируется по 45 градусов. Затем, с учетом наиболее часто встречаемого направления вектора градиента осуществляется подавление пикселей, определенных как границы объектов. Путем выполнения двойной пороговой фильтрации осуществляется подавление границ с низкой величиной градиента и усиление границ со средними и высокими значениями величин градиента.

Также алгоритм осуществляет замыкание незамкнутых контуров за счет использования операций морфологического преобразования (дилатации и эрозии). Затем осуществляется упрощение замкнутых контуров, путем снижения количества опорных точек с применением алгоритма Рамера-Дугласа-Пекера. На заключительном этапе работы алгоритма оценивается форма замкнутых упрощенных контуров объектов. Осуществляется поиск тех контуров, которые удовлетворяют заданным критериям отбора.

4. На языке программирования Python разработана программная реализация предложенного алгоритма, которая была протестирована на различных изображениях (см. главу 3)

Основные результаты работы были доложены на V Международной научно-практической конференции (школы-семинара) молодых ученых «Прикладная математика и информатика: современные исследования в области естественных и технических наук»

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Hahmann, F. Eye Localization Using the Discriminative Generalized Hough Transform [Text] / Ferdinand Hahmann, Heike Ruppertshofen, Gordon Böer, Ralf Stannarius, Hauke Schramm // Joint DAGM (German Association for Pattern Recognition) and OAGM Symposium, DAGM/OAGM 2012: Pattern Recognition - Joint 34th DAGM and 36th OAGM Symposium, Graz, Austria, August 28-31, 2012. Proceedings – Springer-Verlag Berlin Heidelberg 2012. – pp. 155-164
2. Hamouz, M. Hypotheses-Driven Affine Invariant Localization of Faces in Verification Systems [Text] / M. Hamouz, J. Kittler, J. K. Kamarainen, H. Kälviäinen // International Conference on Audio- and Video-Based Biometric Person Authentication, AVBPA 2003: Audio- and Video-Based Biometric Person Authentication – 4th International Conference, AVBPA 2003 Guildford, UK, June 9–11, 2003. Proceedings. – Springer-Verlag Berlin Heidelberg 2003. – pp. 276-284
3. Reilly, V. Geometric Constraints for Human Detection in Aerial Imagery [Text] / Vladimir Reilly, Berkan Solmaz, Mubarak Shah // European Conference on Computer Vision ECCV 2010: Computer Vision – ECCV 2010 – 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part VI. – Springer-Verlag Berlin Heidelberg 2010. pp. 252-265
4. Das, D. Object Detection and Localization in Clutter Range Images Using Edge Features [Text] / Dipankar Das, Yoshinori Kobayashi, Yoshinori Kuno // International Symposium on Visual Computing, ISVC 2009: Advances in Visual Computing – 5th International Symposium, ISVC 2009, Las Vegas, NV, USA, November 30-December 2, 2009. Proceedings, Part II. – Springer-Verlag Berlin Heidelberg 2009. pp. 172-183
5. Mader, A. Detection and Localization of Landmarks in the Lower Extremities Using an Automatically Learned Conditional Random Field [Text] / Alexander Oliver Mader, Cristian Lorenz, Martin Bergtholdt, Jens von Berg,

Hauke Schramm, Jan Modersitzki, Carsten Meyer // International Workshop on Graphs in Biomedical Image Analysis, International Workshop on Imaging Genetics, International Workshop on Mathematical Foundations of Computational Anatomy, GRAIL 2017, MICGen 2017, MFCA 2017: Graphs in Biomedical Image Analysis, Computational Anatomy and Imaging Genetics – First International Workshop, GRAIL 2017, 6th International Workshop, MFCA 2017, and Third International Workshop, MICGen 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 10–14, 2017, Proceedings. – Springer International Publishing AG 2017. pp. 64-75

6. Sudowe, P. Efficient Use of Geometric Constraints for Sliding-Window Object Detection in Video [Text] / Patrick Sudowe, Bastian Leibe // International Conference on Computer Vision Systems, ICVS 2011: Computer Vision Systems – 8th International Conference, ICVS 2011, Sophia Antipolis, France, September 20-22, 2011. Proceedings. – Springer-Verlag Berlin Heidelberg 2011. pp. 11-20

7. Pallawala, P. Automated Optic Disc Localization and Contour Detection Using Ellipse Fitting and Wavelet Transform [Text] / P. M. D. S. Pallawala, Wynne Hsu, Mong Li Lee, Kah-Guan Au Eong // European Conference on Computer Vision, ECCV 2004: Computer Vision - ECCV 2004 – 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part II. – Springer-Verlag Berlin Heidelberg 2004. pp. 139-151

8. Akram, M. Retinal Images: Optic Disk Localization and Detection [Text] / M. Usman Akram, Aftab Khan, Khalid Iqbal, Wasi Haider Butt // International Conference Image Analysis and Recognition, ICIAR 2010: Image Analysis and Recognition – 7th International Conference, ICIAR 2010, Póvoa de Varzin, Portugal, June 21-23, 2010, Proceedings, Part II. – Springer-Verlag Berlin Heidelberg 2010. pp. 40-49

9. Zheng, Q. Text Localization and Recognition in Complex Scenes Using Local Features [Text] / Qi Zheng, Kai Chen, Yi Zhou, Congcong Gu, Haibing Guan // Asian Conference on Computer Vision, ACCV 2010: Computer

Vision – ACCV 2010 – 10th Asian Conference on Computer Vision, Queenstown, New Zealand, November 8-12, 2010, Revised Selected Papers, Part III. – Springer-Verlag Berlin Heidelberg 2011. pp. 121-132

10. Abdelrahman, M. Solving Geometric Co-registration Problem of Multi-spectral Remote Sensing Imagery Using SIFT-Based Features toward Precise Change Detection [Text] / Mostafa Abdelrahman, Asem Ali, Shireen Elhabian, Aly A. Farag // International Symposium on Visual Computing, ISVC 2011: Advances in Visual Computing – 7th International Symposium, ISVC 2011, Las Vegas, NV, USA, September 26-28, 2011. Proceedings, Part II. – Springer-Verlag Berlin Heidelberg 2011. pp. 607-616

11. Samei, G. Detection and Registration of Ribs in MRI Using Geometric and Appearance Models [Text] / Golnoosh Samei, Gábor Székely, Christine Tanner // International Conference on Medical Image Computing and Computer-Assisted Intervention, MICCAI 2014: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2014 – 17th International Conference, Boston, MA, USA, September 14-18, 2014, Proceedings, Part I. – Springer International Publishing Switzerland 2014. pp. 706-713

12. Zhang, J. Geometric Constrained Joint Lane Segmentation and Lane Boundary Detection [Text] / Jie Zhang, Yi Xu, Bingbing Ni, Zhenyu Duan // European Conference on Computer Vision, ECCV 2018: Computer Vision – ECCV 2018 – 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part I. – Springer Nature Switzerland AG 2018. pp. 502-518

13. Maigrot, C. Tampering Detection and Localization in Images from Social Networks: A CBIR Approach [Text] / Cedric Maigrot, Ewa Kijak, Ronan Sicre, Vincent Claveau // International Conference on Image Analysis and Processing, ICIAP 2017: Image Analysis and Processing - ICIAP 2017 – 19th International Conference, Catania, Italy, September 11-15, 2017, Proceedings, Part I. – Springer International Publishing AG 2017. pp. 750-761

14. Kasar, T. Multi-script and Multi-oriented Text Localization from Scene Images [Text] / Thotreingam Kasar, Angarai G. Ramakrishnan //

International Workshop on Camera-Based Document Analysis and Recognition, CBDAR 2011: Camera-Based Document Analysis and Recognition – 4th International Workshop, CBDAR 2011, Beijing, China, September 22, 2011, Revised Selected Papers. – Springer-Verlag Berlin Heidelberg 2012. pp. 1-14

15. Helmer, S. Multiple Viewpoint Recognition and Localization [Text] / Scott Helmer, David Meger, Marius Muja, James J. Little, David G. Lowe // Asian Conference on Computer Vision, ACCV 2010: Computer Vision – ACCV 2010 – 10th Asian Conference on Computer Vision, Queenstown, New Zealand, November 8-12, 2010, Revised Selected Papers, Part I. – Springer-Verlag Berlin Heidelberg 2011. pp. 464-477

16. Delalandre, M. A Performance Characterization Algorithm for Symbol Localization [Text] / Mathieu Delalandre, Jean-Yves Ramel, Ernest Valveny, Muhammad Muzzamil Luqman // International Workshop on Graphics Recognition, GREC 2009: Graphics Recognition. Achievements, Challenges, and Evolution – 8th International Workshop, GREC 2009, La Rochelle, France, July 22-23, 2009. Selected Papers. – Springer-Verlag Berlin Heidelberg 2010. pp. 260-271

17. Figueiredo, I. Automatic Optic Disc Detection in Retinal Fundus Images Based on Geometric Features [Text] / Isabel N. Figueiredo, Sunil Kumar // International Conference Image Analysis and Recognition, ICIAR 2014: Image Analysis and Recognition – 11th International Conference, ICIAR 2014, Vilamoura, Portugal, October 22-24, 2014, Proceedings, Part II. – Springer International Publishing Switzerland 2014. pp. 285-292

18. Ardeshir, S. GIS-Assisted Object Detection and Geospatial Localization [Text] / Shervin Ardeshir, Amir Roshan Zamir, Alejandro Torroella, Mubarak Shah // European Conference on Computer Vision, ECCV 2014: Computer Vision – ECCV 2014 – 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI. – Springer International Publishing Switzerland 2014. pp. 602-617

19. Heyvaert, M. Combining Geometric Edge Detectors for Feature Detection [Text] / Michaël Heyvaert, David Van Hamme, Jonas Coppens, Peter Veelaert // International Conference on Advanced Concepts for Intelligent Vision Systems, ACIVS 2010: Advanced Concepts for Intelligent Vision Systems – 12th International Conference, ACIVS 2010, Sydney, Australia, December 13-16, 2010, Proceedings, Part I. – Springer-Verlag Berlin Heidelberg 2010. pp. 221-232

20. Tajbakhsh, N. Automatic Polyp Detection Using Global Geometric Constraints and Local Intensity Variation Patterns [Text] / Nima Tajbakhsh, Suryakanth R. Gurudu, Jianming Liang // International Conference on Medical Image Computing and Computer-Assisted Intervention, MICCAI 2014: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2014 – 17th International Conference, Boston, MA, USA, September 14-18, 2014, Proceedings, Part II. – Springer International Publishing Switzerland 2014. pp. 179-18