

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 Агапов Константин Олегович

1. Тема Разработка алгоритма распознавания жестов кисти руки
2. Срок сдачи студентом законченной выпускной квалификационной работы
21 июня 2019 г.
3. Исходные данные к выпускной квалификационной работе: получение видео посредством веб-камеры, применение алгоритма вычитания фона MOG2, детекторы границ объектов, алгоритмы поиска контуров на изображении.
4. Содержание выпускной квалификационной работы (перечень подлежащих разработке вопросов, разделов)

Введение

1 Анализ состояния вопроса

2 Разработка алгоритма распознавания жестов кисти руки

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.

Заказчик, директор IT Students

(подпись)

А.В. Очеповский

(И.О. Фамилия)

Руководитель выпускной
квалификационной работы

(подпись)

В.С. Климов

(И.О. Фамилия)

Задание принял к исполнению

(подпись)

К.О. Агапов

(И.О. Фамилия)

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 Агапов Константин Олегович

по теме «Разработка алгоритма распознавания жестов кисти руки»

Наименование раздела работы	Плановый срок выполнения раздела	Фактический срок выполнения раздела	Отметка о выполнении и	Подпись руководителя
Анализ предметной области	1.10.2018	1.10.2018		
Разработка алгоритма распознавания жестов кисти руки	12.12.2018	12.12.2018		
Практическая реализация предложенных решений	17.01.2019	17.01.2019		
Подготовка пояснительной	25.02.2019	25.02.2019		

записки				
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

К.О. Агапов

АННОТАЦИЯ

Тема бакалаврской работы: «Разработка алгоритма распознавания жестов кисти руки».

В данной бакалаврской работе исследуются процесс синтеза алгоритма для распознавания жестов кисти руки.

В работе предложен алгоритм, позволяющий распознавать до 5 жестов кисти руки. Предложенный подход основан на использовании алгоритма вычитания фона MOG2, детекторе границ объектов, алгоритме поиска контуров и алгоритме подсчета углов контура. Разработанное программное обеспечение позволяет анализировать потоковое видео, получаемое с веб-камеры.

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

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

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

В первой главе проводится обзор методов распознавания образов, используемых в различных областях науки для анализа изображений.

Во второй главе описывается предложенный алгоритм для распознавания жестов кисти руки.

В третьей главе описывается программная реализация предложенного алгоритма, демонстрируются примеры работы программы на изображениях полученных с веб-камеры.

В заключении представлены выводы по проделанной работе.

В работе присутствуют 1 таблица, 25 рисунков. Список литературы состоит из 20 литературных источников. Общий объем выпускной квалификационной работы составляет 40 страниц.

ABSTRACT

The topic of the given graduation project is Development of hand gesture recognition algorithm.

In this bachelor's work, hand gesture recognition algorithms are investigated.

The paper proposes an algorithm to recognize up to 5 hand gestures. The proposed approach is based on the use of the MOG2 lantern subtraction algorithm, the contour search algorithm and the contour angle calculation algorithm. The developed software allows analyzing streaming video received from a webcam.

In preparing the studies presented in this bachelor's work, 1 article was published.

Undergraduate work consists of introduction, three chapters, conclusion and list of references.

The introduction describes the relevance of the research, provides a brief description of the bachelor's work.

The first chapter reviews methods of pattern recognition used in various fields of science for image analysis.

The second chapter describes the proposed algorithm for recognizing hand gestures.

In the third chapter describes the software implementation of the proposed algorithm, demonstrates examples of the work of the program on images obtained from a webcam.

In the conclusion presents conclusions on the work done. This graduation work consist of an introduction on 2 pages, including 25 figures, 1 table, conclusion on 2 pages and the list of 20 references. The total amount of final qualifying work is 40 pages.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	5
1 АНАЛИЗ ТЕХНОЛОГИЙ РАСПОЗНАВАНИЯ ОБРАЗОВ	7
2 РАЗРАБОТКА АЛГОРИТМА РАСПОЗНАВАНИЯ ЖЕСТОВ КИСТИ РУКИ	10
2.1 Описание предложенного алгоритма	10
2.2 Локализация области распознавания жестов и преобразование в цветовую модель Gray	12
2.3 Применение фильтра Гаусса	13
2.3 Применение субтрактора фона	15
2.3 Применение детектора границ Canny	22
2.4 Алгоритм анализа контура кисти руки	27
3 ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ ПРЕДЛОЖЕННЫХ РЕШЕНИЙ	29
3.1 Описание разработанного программного обеспечения	29
3.2 Тестирование программного обеспечения	31
ЗАКЛЮЧЕНИЕ	36
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ	38

ВВЕДЕНИЕ

Распознавание образов – раздел информатики и смежных дисциплин, направленный на создание методов классификации и идентификации различных предметов, явлений, процессов, сигналов, ситуаций, объектов, которые характеризуются конечным набором некоторых свойств и признаков.

Технологии распознавания образов уже уверенно используются при решении таких задач, как распознавание символов текста, распознавание речи на основе анализа ауди записи, постановка диагноза на основе анализа медицинских данных, распознавание лиц, на основе анализа изображений, прогноз погоды, классификация документов при организации систем информационного поиска.

Развитие технологий направленных на создание человеко-машинных интерфейсов, отвечающих за взаимодействия человека с управляемыми машинами, является одним из перспективных направлений исследований, связанных с практическим применением алгоритмов распознавания образов. Поэтому актуальной можно признать цель данной бакалаврской работы – разработка алгоритма распознавания жестов кисти руки.

Вопросами исследования алгоритмов распознавания образов занимаются такие современники, как Guanlin Li, Zhanhong Ma, Haiguang Wang, Masato Yonekawa, Hiroaki Kurokawa, Byungjun Son, Sung-Hyuk Cha, Yillbyung Lee, Congcong Li, Guangda Su, Yan Shang, Yingchun Li, Yan Xiang, и Mohamed Abdel-Mottaleb, Mohammad H. Mahoor и др. [2-18].

В ходе выполнения бакалаврской работы разработан алгоритм распознавания жестов кисти руки. Данный алгоритм основан на преобразовании изображения в цветовую модель Gray, применении фильтра Гаусса для очистки изображения от цифровых шумов (посредством размытия), применении субтрактора фона MOG2 (для удаления фона), применения детектора границ Canny (для выделения конура кисти руки),

применении алгоритма для анализа контура кисти руки (для определения демонстрируемого жеста).

При выполнении бакалаврской работы было разработано и протестировано программное обеспечение, реализующее предложенный алгоритм распознавания жестов кисти руки. Результаты испытаний алгоритма показали, что он удовлетворительно справляется с задачей распознавания жестов, в том числе в условиях слабой освещенности и при наличии сложного фона (содержащего большое количество посторонних объектов)

В ходе выполнения бакалаврской работы была опубликована статья «Сегментация изображения с использованием алгоритма k-means» на тему практического применения алгоритмов анализа изображений.

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 ТЕХНОЛОГИЙ РАСПОЗНАВАНИЯ ОБРАЗОВ

Распознавание образов является научным направлением, связанным с разработкой принципов и построения систем, предназначенных для определения принадлежности объектов (фрагментов изображений, звуковых файлов, различных сигналов, наборов данных) к одному из заранее известных классов.

Технологии распознавания образов успешно используются при решении следующих задач:

- распознавание символов;
- распознавание речи;
- установление медицинского диагноза;
- прогноз погоды;
- распознавание лиц
- классификация документов и др.

Под процессом распознавания образов понимается отнесение исходных данных к определенному классу с помощью выделения существенных признаков характеризующих данный класс образов.

Обычно, постановка задачи, требующая для решения технологий распознавания образов, формулируется следующим образом. Дано множество объектов, относительно которых необходимо провести процедуру классификацию для объектов, у которых отсутствует метка класса. Исходными данными в этом случае информация о классах, описание всего множества и описание информации об объектах, классовая принадлежность которых неизвестна. Требуется по имеющейся информации о классах и описании объектов установить, к каким классам они относятся.

При решении задач распознавания образов в рамках анализа изображений чаще всего рассматриваются монохромные изображения, что дает возможность рассматривать изображение как функцию на плоскости.

Если рассмотреть точечное множество на плоскости T , где функция $f(x, y)$ выражает в каждой точке изображения его характеристику – яркость, прозрачность (значение в цветовой модели Gray), то такая функция и будет являться математической записью изображения изображением.

Оценивая сходство между изображениями конечным множеством различных изображений $f_1(x, y), f_2(x, y), \dots, f_m(x, y)$ становится возможным решение постановка задачи распознавания образов.

Стоит отметить, что конкретный вид постановки задачи зависит от подходов, применяемых в последующем для ее решения.

При решении задач оптического распознавания образов при классификации объектов могут учитываться аналитическому описанию формы, положение объекта, геометрические шаблоны для сравнения с элементами объекта, цвет пикселей и т.д.

Классическая структура системы распознавания образов представлена на рисунке 1.1.

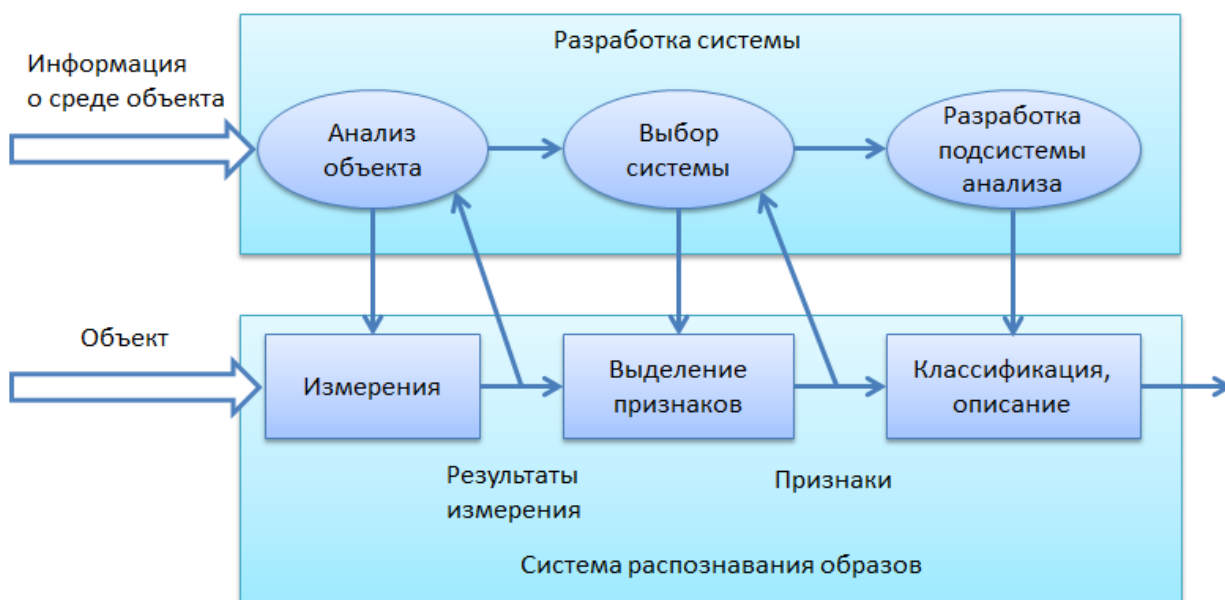


Рисунок 1.1 – Классическая структура системы распознавания образов

Таким образом, задачи распознавания образов с математической точки зрения рассматриваются, как решение задачи классификации. С развитием искусственного интеллекта, при решении задач распознавания образов стали

применяться технологии машинного обучения, способные решать задачи классификации, такие как:

- нейронные сети прямого распространения;
- рекуррентные нейронные сети;
- деревья принятия решений;
- метод Виолы-Джонса;
- метод HOG-SVM;
- метод k ближайших соседей;
- другие методы классификации.

Развитие технологий направленных на создание человеко-машинных интерфейсов, отвечающих за взаимодействия человека с управляемыми машинами, является одним из перспективных направлений исследований, связанных с практическим применением алгоритмов распознавания образов.

Развитие человеко-машинных интерфейсов направлено повышения простоты и удобства взаимодействия с управляемыми устройствами, которое возможно, например, с помощью жестов.

Поэтому актуальной можно признать цель данной бакалаврской работы – разработка алгоритма распознавания жестов кисти руки.

Цель работы – разработка алгоритма распознавания жестов кисти руки.

Поставленную цель предполагается достигнуть последовательным решением ряда задач:

1. Провести анализ существующих исследований в области распознавания образов.
2. Разработать алгоритм распознавания жестов кисти руки.
3. Разработать программное обеспечение, реализующее предложенный алгоритм
4. Протестировать работу алгоритма и оценить его эффективность.

2 РАЗРАБОТКА АЛГОРИТМА РАСПОЗНАВАНИЯ ЖЕСТОВ КИСТИ РУКИ

2.1 Описание предложенного алгоритма

Предложенный алгоритм распознавания жестов кисти руки состоит из четырех основных этапов:

- первый этап – подготовка изображения к анализу;
- второй этап – очистка заднего фона изображения и упрощение изображения путем извлечения контура кисти руки;
- третий этап – анализ контура кисти руки для определения демонстрируемого жеста;
- четвертый этап – заключительный этап, в рамках которого осуществляется вывод результата распознавания жеста.

Первый этап алгоритма направлен на подготовку изображения к анализу и включает в себя: снижение размерности признакового пространства за счет смены цветовой модели из RGB в Gray, а также сглаживание цифровых шумов, вносимых сенсором веб-камеры за счет применения фильтра Гаусса (размытие изображение).

Второй этап алгоритма предназначен для отделения фона от изображения руки за счет применения субтрактора MOG2 и последующем преобразовании изображения в набор контуров с помощью детектора границ Canny.

Третий этап обработки изображения направлен на извлечение одного единственного контура (имеющего максимальную длину) – контура кисти руки. Также здесь осуществляет анализ контура путем подсчета количества отогнутых пальцев. По количеству отогнутых пальцев определяется демонстрируемый жест изображения.

Четвертым этапом является вывод результатов распознавания жеста кисти руки.

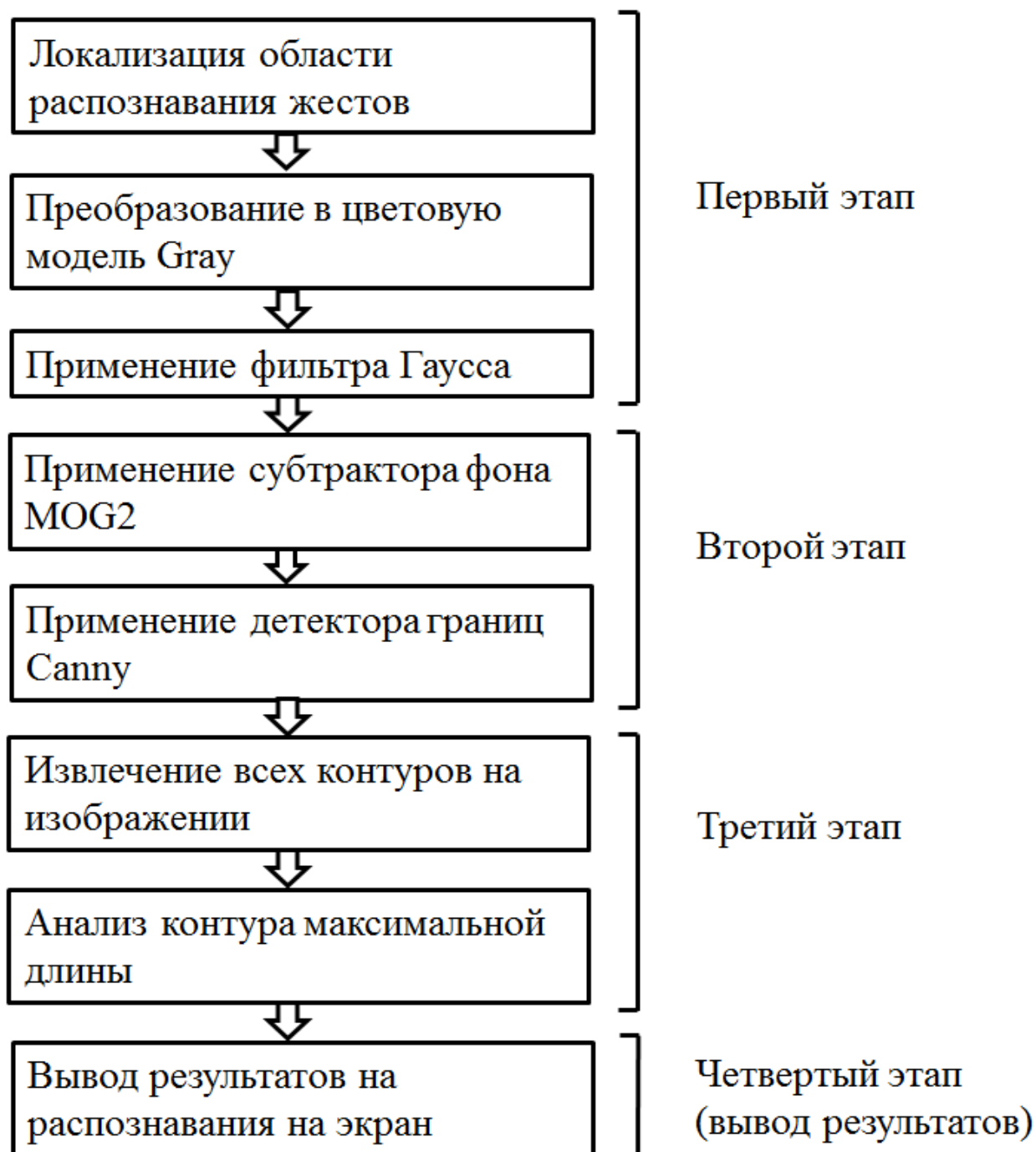


Рисунок 2.1 – Схема работы алгоритма

Перейдем к математическому описанию каждого из представленных этапов. Для каждого этапа приведем примеры получаемых промежуточных изображений.

2.2 Локализация области распознавания жестов и преобразование в цветовую модель Gray

Изображения получаемые с фотосенсоров, кодируются в цветовой модели RGB. Это означает что, цвет каждого пикселя определяются в зависимости от пропорций трех цветовых компонентов – красного цвета, зеленого и синего. Тогда анализируемое изображение можно представить в виде функции $frame(x,y)$ (2.1)

$$frame(x, y) = (R_{1,1}; G_{1,1}; B_{1,1}), \dots, (R_{m,n}; G_{m,n}; B_{m,n}), \quad (2.1)$$

где x и y – координаты пикселей изображения по осям x и y ($x=1 \dots m$, $y=1 \dots n$), m и n – соответственно ширина и высота изображения в пикселях, а R , G , B – компоненты цвета пикселей каждый из которых лежит в диапазоне от 0 до 255.

При решении задач распознавания образов в рамках анализа изображений чаще всего рассматриваются монохромные изображения, что дает возможность рассматривать изображение как функцию на плоскости. Если рассмотреть точечное множество на плоскости T , где функция $f(x, y)$ выражает в каждой точке изображения его характеристику – яркость, прозрачность (значение в цветовой модели Gray), то такая функция и будет являться математической записью изображения изображением.

Существуют множество способов перевода изображения в цветовую модель Gray, например, это можно сделать, воспользовавшись методом средних. В этом случае расчет яркости каждого пикселя будет производиться по формуле (2.2):

$$frame_gray(x, y) = (R_{x,y} + G_{x,y} + B_{x,y}) / 3 \quad (2.2)$$

Мы будем пользоваться рекомендацией BT.601 международного союза электросвязи, в которой приведена следующая формула преобразования цветовой модели RGB в Gray (2.3)

$$frame_gray(x, y) = 0,299 \cdot R_{x,y} + 0,587 \cdot G_{x,y} + 0,114 \cdot B_{x,y}, \quad (2.3)$$

где $frame_gray(x,y)$ – изображение, переведенное в оттенки серого.

2.3 Применение фильтра Гаусса

Практически всегда, на изображениях, получаемых с фотосенсоров камер из-за несовершенства технологии, появляются дефекты в виде цифровых шумов. Цифровой шум представляет собой наложение на изображение разноцветных зерен - пикселей случайного цвета и яркости. Такой шум обычно очень заметен на однотонных участках изображения.

Цифровой шум, помимо неэстетической составляющей вносит и практические проблемы, связанные со снижением эффективности алгоритмов анализа изображений.

Стандартной практикой у специалистов, занимающихся компьютерным зрением, является сглаживание цифрового шума на изображении с помощью фильтра Гаусса. Визуально применение фильтра Гаусса выглядит как размытие изображения.

Под применением фильтра к изображению понимается выполнение следующих расчетов, представленных на (2.4)

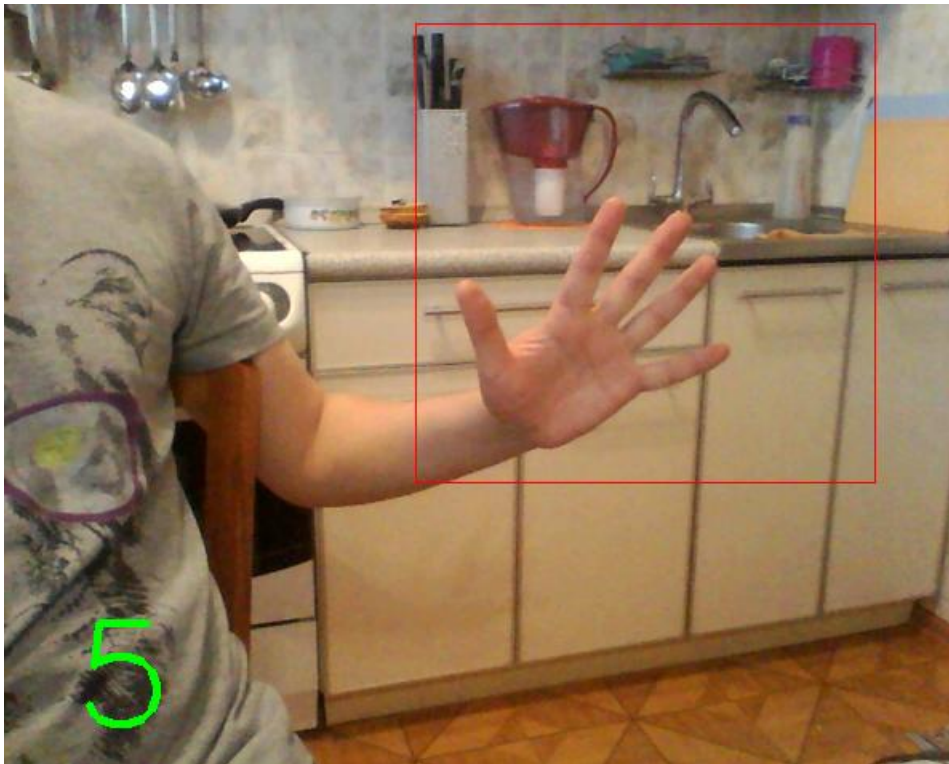
$$\begin{aligned} frame_gauss(x, y) &= frame_gray(x, y) * w(s, t) = \\ &= \sum_{s=-a}^a \sum_{t=-b}^b [frame_gray(x-s, y-t) \times w(s, t)] \end{aligned}, \quad (2.4)$$

где $frame_gauss(x,y)$ – изображение с результатом применения матрицы свертки, w – матрица свертки, $frame_gray(x,y)$ – исходное изображение к которому применяется матрица свертки, a, b – коэффициенты, которые зависят от размеров матрицы ядра свертки. Для матрицы (2.4) эти коэффициенты равны: $a=2, b=2$.

В нашем случае, для размытия изображения применяется матрица свертки размером 5 на 5, имеющей следующий вид (2.5):

$$w = \frac{1}{256} \begin{pmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{pmatrix} \quad (2.5)$$

Пример применения фильтра Гаусса к изображению представлен на рисунке 2.2.



а)



б)



в)

Рисунок 2.2 – Изображение $frame_gauss(x,y)$

2.3 Применение субтрактора фона

Субтрактор фона – это алгоритм, анализирующий несколько последовательных кадров видеоряда с целью выделения модели и фона и применение ее к текущему кадру с целью выделения маски переднего плана.

Маска переднего плана (маска объекта) – это матрица размерностью как исходное изображение, компоненты матрицы могут принимать только два значения – 0 или 1. Значение 0 компонента матрицы маски означает, что пиксель текущего кадра, расположенный по этим же координатам относится к заднему плану (фону), а значение 1 компонента матрицы маски означает, что пиксель относится к переднему плану (к объекту).

Обычно матрицу маски объекта визуализируют с помощью черно-белого изображения, обозначая белым цветом компоненты, принимающие значение 1, а черным цветом – компоненты принимающие значение 0.

Маска переднего плана рассчитывается на основе изображения текущего кадра и модели фона путем попиксельного вычитания одного из другого. Затем полученные по модули разности сравниваются с фиксированным значением (значение порога T) и если разность больше чем значение порога, то пиксель по данным координатам считается относящимся к объекту, если разность меньше значения порога, то пиксель считается относящимся к фону (рисунок 2.3) (2.6).

$$mask(x, y) = \begin{cases} 1 & \text{if } |frame_gauss(x, y) - fon(x, y)| \geq T \\ 0 & \text{if } |frame_gauss(x, y) - fon(x, y)| < T \end{cases}, \quad (2.6)$$

где $mask(x, y)$ – маска объекта, $frame_gauss(x, y)$ – текущий кадр, $fon(x, y)$ – модель фона, рассчитанная субтрактором.

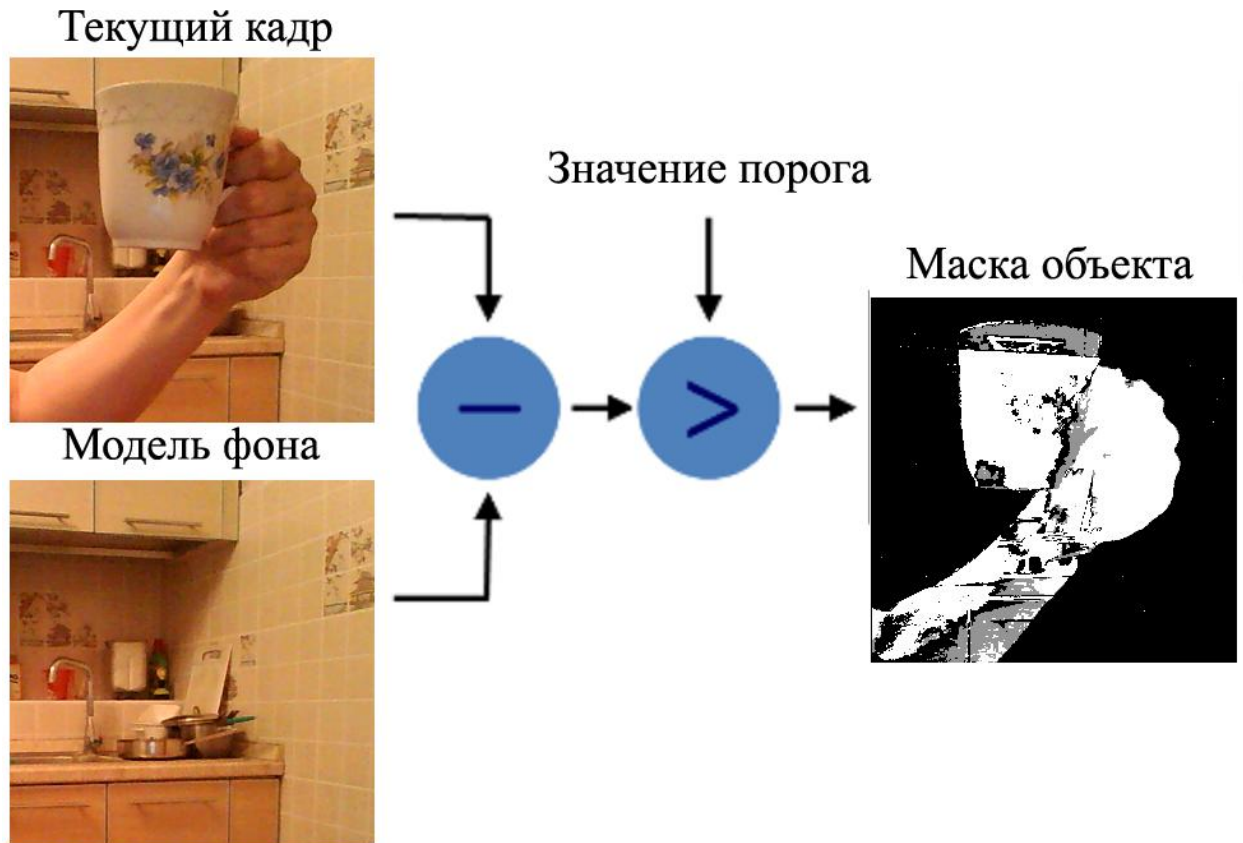


Рисунок 2.3 – Получение изображения $mask(x,y)$

Отличительной особенностью функционирования субтракторов фона является динамический пересчет модели фона на каждом кадре изображения. Это означает, что если движущийся объект, присутствующий в видео ряде, остановится и будет находиться в неподвижном состоянии, то через некоторое время субтрактор фона будет считать данный объект элементом фона.

Наиболее известные субтракторы фона – MOG, MOG2 и GNG. Они отличаются друг от друга способами расчета модели фона.

Работа алгоритма MOG основана на использовании смеси Гауссовских распределений. Данный алгоритм был в 2001 году P. KadewTraKuPong и R. Bowden. Он использует метод моделирования каждого фонового пикселя с помощью смеси K -гауссовых распределений (при $K = 3 - 5$). Весы смеси представляют временные пропорции, которые от того, как долго цвета

пикселей остаются на изображении. Вероятные цвета пикселей фона - это те, которые остаются дольше всех и более статичны.

Программная реализация применения алгоритма MOG к видеоряду представлена на рисунке 2.4.

```
1 import numpy as np
2 import cv2
3
4 cap = cv2.VideoCapture('vtest.avi')
5
6 fgbg = cv2.createBackgroundSubtractorMOG()
7
8 while(1):
9     ret, frame = cap.read()
10
11     fgmask = fgbg.apply(frame)
12
13     cv2.imshow('frame',fgmask)
14     k = cv2.waitKey(30) & 0xff
15     if k == 27:
16         break
17
18 cap.release()
19 cv2.destroyAllWindows()
```

Рисунок 2.4 – Программный код на языке Python для применения субтрактора фона MOG для тестового видео vtest.avi

Алгоритм MOG2 тоже основан на использовании смеси Гауссовских распределений. Алгоритм был предложен Z.Zivkovic в 2004 году. Важной особенностью этого алгоритма является то, что он выбирает соответствующее количество гауссовского распределения для каждого пикселя. Он обеспечивает лучшую адаптацию к изменениям связанным, например переменной освещения объектов.

Другой особенностью алгоритм MOG2 является добавление в матрицу маски третьего типа значений (помимо «0» - пиксель фона, «1» - пиксель

объекта) – «2» - пиксель тени объекта. Тени объектов при визуализации маски принято отмечать серым цветом.

Программная реализация применения алгоритма MOG к видеоряду представлена на рисунке 2.5.

```
1 import numpy as np
2 import cv2
3
4 cap = cv2.VideoCapture('vtest.avi')
5
6 fgbg = cv2.createBackgroundSubtractorMOG2()
7
8 while(1):
9     ret, frame = cap.read()
10
11     fgmask = fgbg.apply(frame)
12
13     cv2.imshow('frame',fgmask)
14     k = cv2.waitKey(30) & 0xff
15     if k == 27:
16         break
17
18 cap.release()
19 cv2.destroyAllWindows()
```

Рисунок 2.5 – Программный код на языке Python для применения субтрактора фона MOG2 для тестового видео vtest.avi

Алгоритм GMG основан на объединении статистической оценки фонового изображения и попиксельной байесовской сегментации. Данный алгоритм предложен Andrew B. Godbehere, Akihiro Matsukawa, Ken Goldberg в 2012 году.

Он использует последовательность из нескольких кадров (по умолчанию 120) для моделирования фона изображения. В нем используется вероятностный алгоритм сегментации переднего плана, который идентифицирует возможные объекты переднего плана с использованием

байесовского вывода. Вероятностные оценки являются адаптивными: новые изменения в кадре имеют более сильный вес, чем, чем старые. Для устранения нежелательного шума выполняются операций морфологического преобразования, такие как закрытие и открытие.

Программная реализация применения алгоритма MOG к видеоряду представлена на рисунке 2.6.

```
1 import numpy as np
2 import cv2
3
4 cap = cv2.VideoCapture('vtest.avi')
5
6 kernel = cv2.getStructuringElement(cv2.MORPH_ELLIPSE, (3,3))
7 fgbg = cv2.createBackgroundSubtractorGMG()
8
9 while(1):
10     ret, frame = cap.read()
11
12     fgmask = fgbg.apply(frame)
13     fgmask = cv2.morphologyEx(fgmask, cv2.MORPH_OPEN, kernel)
14
15     cv2.imshow('frame', fgmask)
16     k = cv2.waitKey(30) & 0xff
17     if k == 27:
18         break
19
20 cap.release()
21 cv2.destroyAllWindows()
```

Рисунок 2.6 – Программный код на языке Python для применения субтрактора фона GMG для тестового видео vtest.avi

Таблица 1 – Сравнение результатов использования субтракторов фона изображения

Один из кадров тестового видео test.avi



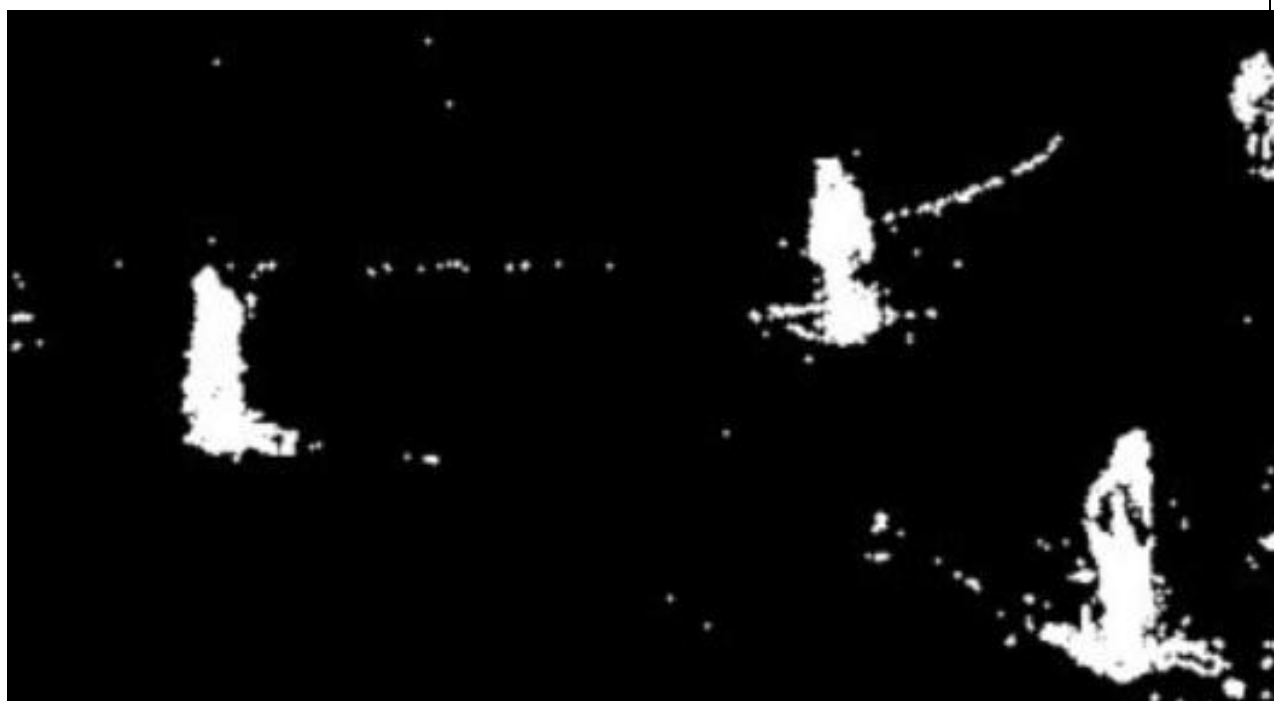
Применение субтрактора MOG:



Применение субтрактора MOG2:



Применение субтрактора GMG:



Тестирование всех типов субтрактора фона применительно к нашей задаче показало, что точность распознавания жестов не зависит от выбора используемого алгоритма. Поэтому остановим свой выбор на субтракторе MOG2, как наиболее распространенном.

Результат применения субтрактора фона MOG2 при решении задачи распознавания жестов кисти руки представлен на рисунке 2.7

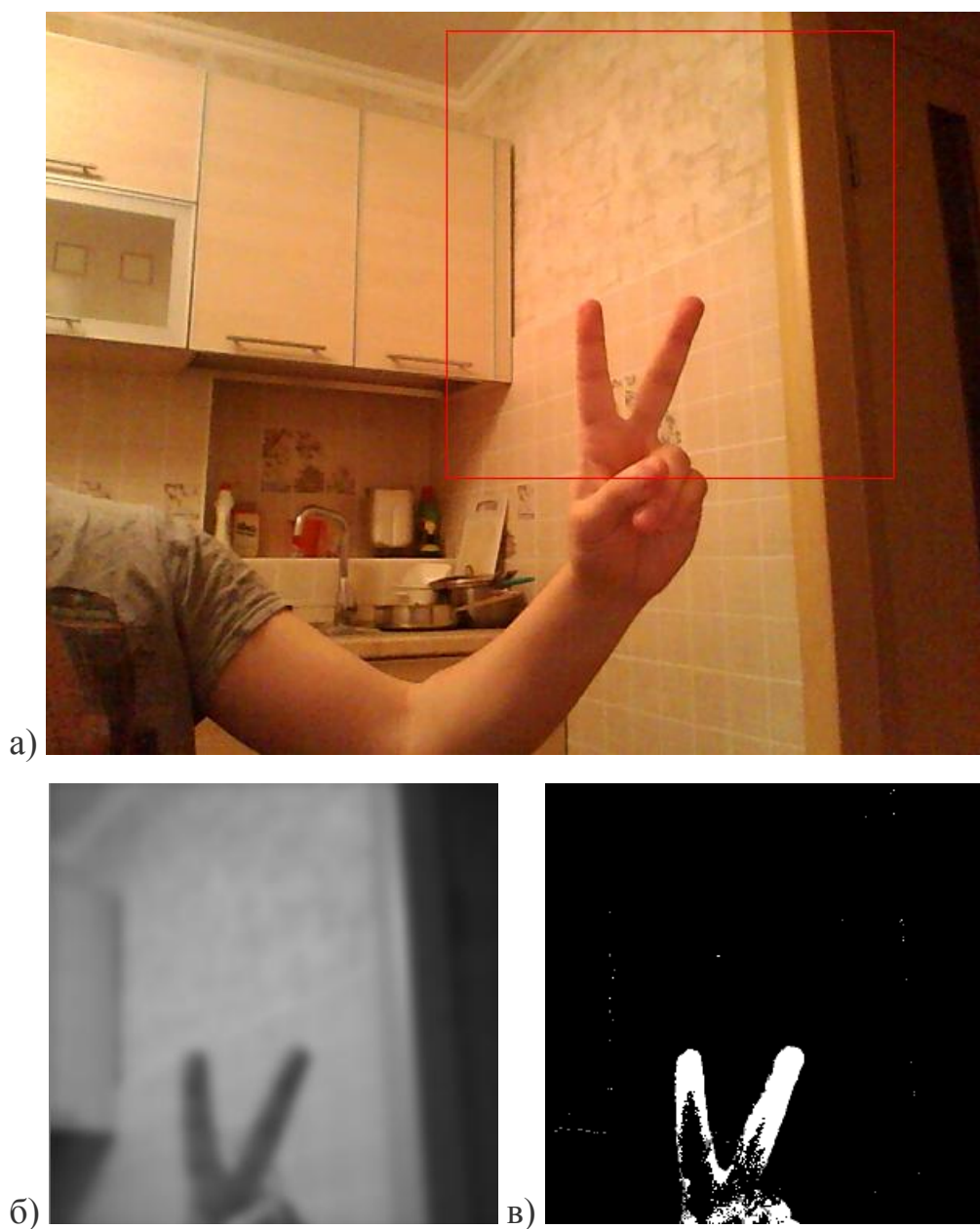


Рисунок 2.7 – Применение субтрактора фона MOG2: а – исходное изображение, б – изображение к которому применяется субтрактор фона, в – сгенерированная субтрактором маска $mask(x,y)$ переднего плана

2.3 Применение детектора границ Canny

Работа метода заключается в поиске кривых на изображениях вдоль которых происходит резкое изменение яркости пикселей.

Детектор границ Канни определяет места резкого изменения яркости пикселей с помощью расчета градиента. Для экономии вычислительных мощностей при программной реализации расчета градиентов дискретный дифференциальный оператор Собеля. Данный оператор позволяет приближено вычислять значения градиента.

Использование оператора Собеля основано на свертке изображения фильтрами представленными ниже (2.7) и (2.8).

Свертка изображения с горизонтальном направлении осуществляется фильтром FGx , а в вертикальном направлении – фильтром FGy . Оба фильтра являются целочисленными.

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

Алгоритмически реализовать расчет градиента (его значения и угла) для каждого пикселя текущего кадра можно с помощью вложенных циклов. Первый цикл будет основан на использовании индекса i , связанного с попиксельным перемещением по горизонтальной оси изображения. Тогда с учетом размера фильтра FGx индекс будет последовательно принимать следующие значения: $i \in (2, 3, \dots, \max(x) - 2)$. Второй цикл, вложенный в первый цикл, будет основан на использовании индекса j , связанного с попиксельным перемещением по вертикальной оси изображения. Тогда с учетом размера фильтра FGx индекс будет последовательно принимать следующие значения: $j \in (2, 3, \dots, \max(y) - 2)$. $\max(x)$ – ширина анализируемого изображения, $\max(y)$ – высота анализируемого изображения.

В этом случае внутри данного цикла будут выполняться следующие действия:

1. Выделение из изображения $mask(x,y)$ фрагмента $part_mask(x,y)$ размерностью 3 на 3 пикселя. При этом центральный пиксель фрагмента находится по координатам i, j изображения $mask(x,y)$.

2. Расчет значения градиента GX в горизонтальном направлении, по формуле, представленной на (2.8):

$$GX = \sum_{y=1}^3 \sum_{x=1}^3 (Part_frame_gauss(x, y) \cdot FGx(x, y)) \quad (2.8)$$

3. Расчет значения градиента GY в вертикальном направлении, по формуле, представленной на (2.9)

$$GY = \sum_{y=1}^3 \sum_{x=1}^3 (Part_frame_gauss(x, y) \cdot FGY(x, y)) \quad (2.9)$$

4. Расчет значения градиента G , как суперпозиция вертикальной и горизонтальной составляющей градиента, по формуле (2.10):

$$G = \sqrt{Gx^2 + Gy^2} \quad (2.10)$$

5. Определения угла θ_temp направления градиента, по формуле представленной на (2.11):

$$\theta_temp = \arctan 2(Gx, Gy) \quad (2.11)$$

6. Округление угла θ_temp до ближайшего значения из ряда из ряда $\frac{\pi}{4} \cdot n$, где $n = 0, 1, \dots, 7$. Округление углов будет необходимо на следующем шаге, для более точного выделения границ объектов. Округление выполняется по формуле (2.12):

$$\theta = \text{round} \left(\frac{\theta_temp}{\frac{\pi}{4}} \right) \cdot \frac{\pi}{4} - \frac{\pi}{2} \quad (2.11)$$

После выполнения всех итераций двух вложенных циклов на выходе будет получено две матрицы: $frame_edge(x,y)$, содержащих в себе значение градиентов и $frame_angle(x,y)$, содержащей в себе округленные углы наклона градиентов (2.12):

$$\begin{aligned} frame_edge(x, y) &= G \\ frame_angle(x, y) &= \theta \end{aligned} \quad (2.12)$$

Для более точного выделения границ объектов необходимо выполнить этап подавления не-максимумов градиентов. Для этого анализируются полученные на предыдущем этапе матрицы $frame_edge(x, y)$ и $frame_angle(x, y)$.

С начала с использованием $frame_angle(x, y)$ определяется наиболее часто встречающееся направление градиента. В примере, представленном на рисунке 2.5, таким направлением является угол 0° .

После того, как определено доминирующее направление градиента максимума определяются те пиксели, у которых значения градиента максимально. При доминирующем направлении градиента «угол 0° » сравниваются пиксели, расположенные по вертикали. На рисунке 2.5 штриховкой отмечены пиксели признанные границами (максимумы).

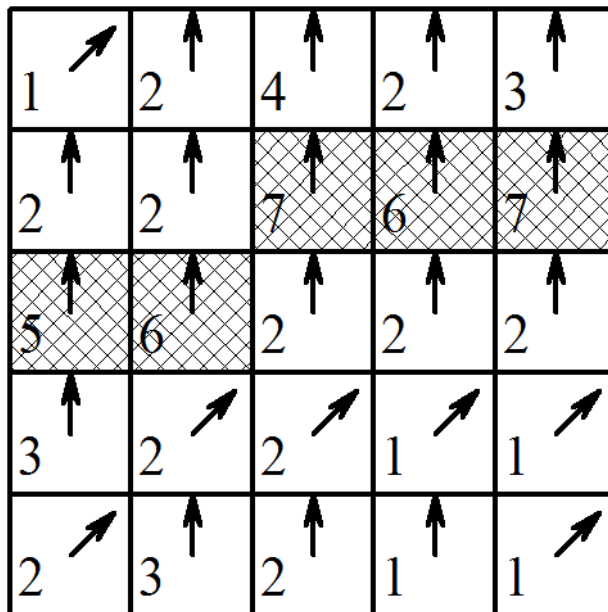


Рисунок 2.5 – Подавление не-максимумов (стрелками показано направление векторов), цифры – значение градиента.

Последним шагом выделения границ является двойная пороговая фильтрация. Верхний порог предназначен для маркировки пикселей как «границы», у которых значение градиента выше значения порога. Нижний

порог предназначен для маркировки пикселей как «не границы», у которых значение градиента ниже значения данного порога. В нашем случае, применение детектора границ Канны к маске объекта, полученного с помощью субтрактора MOG2 (рисунок 2.6 б). В этом случае удастся получить несколько замкнутых контуров: контур – кисти руки, и множество небольших контуров, связанных с присутствием шумов на изображении.

С точки зрения распознавания жеста нас интересует только контур кисти руки. Это контур легко отделить от контуров связанных с шумами – контур кисти обладает максимальной длиной (периметр). Результат использования такого эвристического правила для выделения контура кисти руки представлен на рисунке 2.6, г.

Результат использования детектора границ Канны представлен на рисунке 2.6

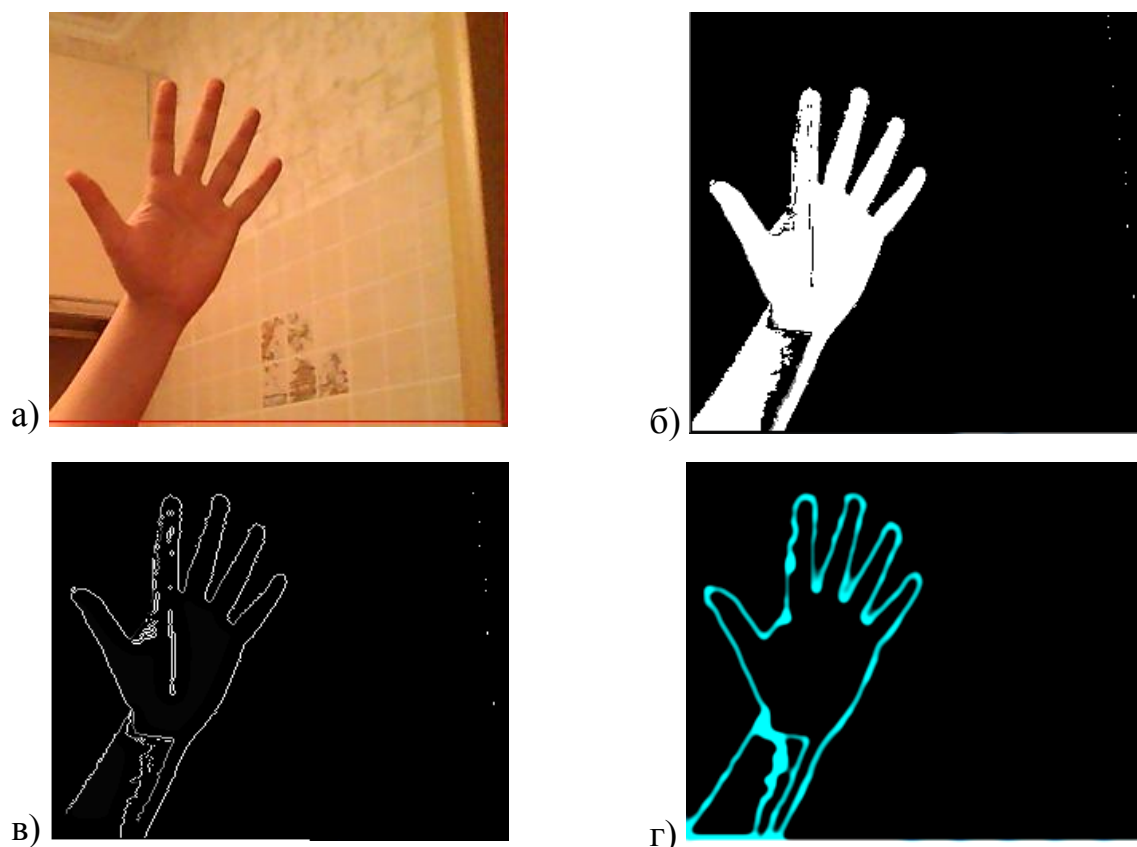


Рисунок 2.6 – Выделение контура руки человека: а – исходное изображение, б – маска переднего плана, выделенная субтрактором MOG2, в – результат применения к маске детектора границ

2.4 Алгоритм анализа контура кисти руки

Теперь, когда на предыдущем шаге мы получили контур кисти руки необходимо его проанализировать, для того, чтобы распознать демонстрируемый жест.

Для распознавания демонстрируемого жеста будем производить расчет количества отогнутых пальцев. Алгоритмически это реализовано следующим образом. Сначала путем анализа опорных точек контура определяются максимальная и минимальная координата точек сначала для оси X, затем для оси Y. По этим координатам строится прямоугольник описывающий контур кисти руки и находится его геометрический центр.

Затем в прямоугольник вписывается эллипс с тем же центром, что и у прямоугольника, а границы эллипса должны отступать от краев прямоугольника так, чтобы сохранялась пропорция показанная на рисунке 2.7 (шаг 3). Данная пропорция была определена опытным путем.

Те части контура руки, которые выступают за эллипс формируют отдельные сегменты контура - контуры кончиков пальцев. Строя отрезки из центра эллипса до всех опорных точек, легко определить внутри каждого сегмента самую удаленную точку от центра. Зная координаты этих сегментных точек можно следить за жестами руки.

В нашем случае подсчет количества сегментных точек позволяет определить количество отогнутых пальцев. При подсчете их количества не учитываются точки, расположенные на границе изображения, так как они связаны не с пальцами, а с запястьем кисти руки.

Визуализация работы алгоритма для анализа контура кисти руки представлена на рисунке 2.7.

Таким образом, предложенный алгоритм позволяет распознавать жесты кисти руки, путем расчета количества отогнутых пальцев.

Результаты тестирования алгоритма распознавания жестов кисти руки подробно разобрано в третьей главе.

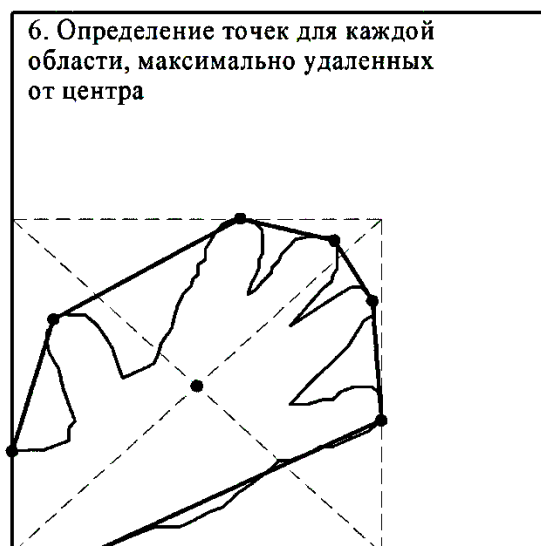
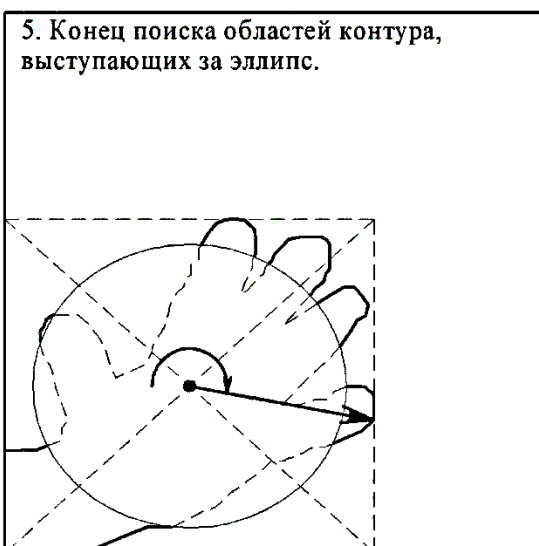
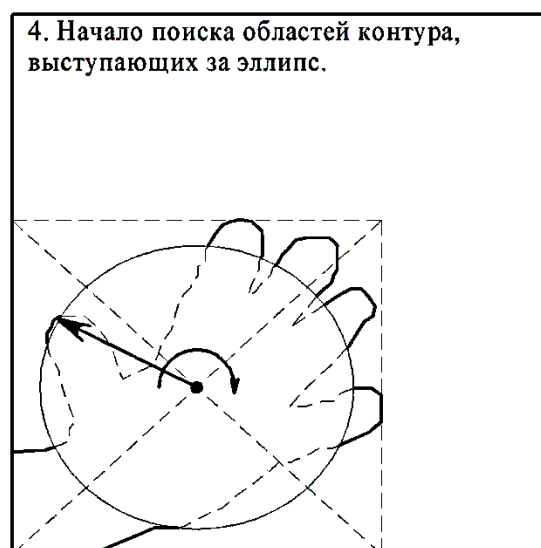
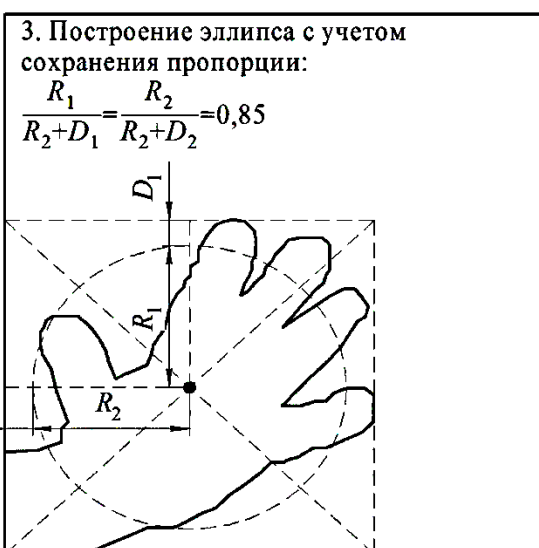
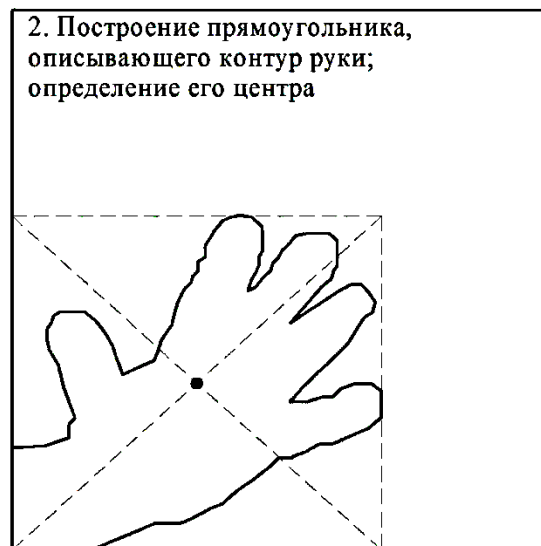
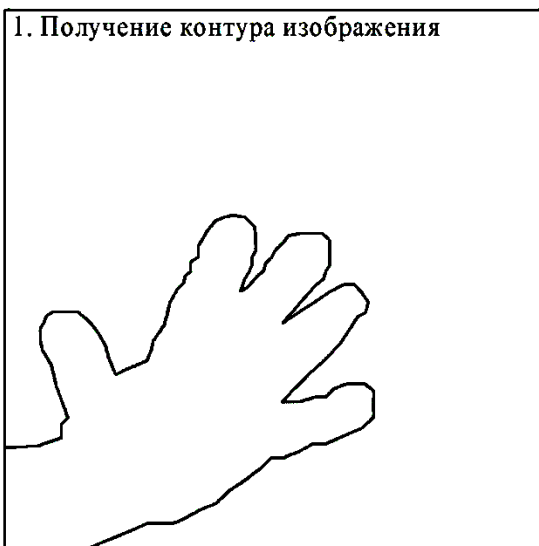


Рисунок 2.7 – Визуализация алгоритма расчета количества отогнутых пальцев

3 ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ ПРЕДЛОЖЕННЫХ РЕШЕНИЙ

3.1 Описание разработанного программного обеспечения

В ходе выполнения ВКР было спроектировано, разработано и протестировано программное обеспечение, реализующее описанный алгоритм распознавания жестов кисти руки.

Возможности и особенности разработанного программного обеспечения:

- автоматическая инициализация и запуск веб-камеры;
- работа на основе библиотеки OpenCV;
- анализ изображений на основе потокового видео, получаемого с веб-камеры;
- демонстрация результатов работы каждого из этапов предложенного алгоритма в виде отдельного изображения;
- возможность распознавания 5 жестов (количество демонстрируемых пальцев).

При запуске приложение на первом этапе происходит инициализация веб-камеры. При этом открывается окно 1 (поз. 1, рисунок 3.1), где выводится информация о поддерживаемом разрешении, частоте кадров. При проблемах инициализации веб-камеры в данном окне будет выведена информация о возникшей ошибке.

Как только веб-камера будет запущена, откроется окно 2 (поз. 2, рисунок 3.1), которое будет демонстрировать получаемое изображение. В данном окне красной рамкой выделена область, для которой запущен алгоритм распознавание жестов. Изображение из данной области подвергает обработке посредством смены цветовой модели из RGB в Gray и применения фильтра Гаусса. Результат такой обработки помещается в окно 4 (поз. 4, рисунок 3.1).

3.2 Тестирование программного обеспечения

Было проведено тестирование эффективности работы предложенного алгоритма.

Тестирование проводилось следующим образом. Перед веб-камерой ноутбука с помощью руки демонстрировались различные жесты. От программы требовалось правильно распознать количество демонстрируемый жест и вывести результат распознавания на экран.

Эффективность работы предложенного алгоритма оценивалась путем подсчета отношения количества правильно распознанных жестов к общему числу жестов. И как показал эксперимент, точность распознавания составляет 81% (в 81 из 100 случаев жесты были распознаны правильно).

При этом стоит отметить, что программа тестировалась в жестких условиях, включающих в себя неоднородный фон, содержащий в себе большое количество объектов различных цветов (что ухудшает работу субтрактора фона) и низкий уровень освещенности, добавляющий на изображение большое количество цифровых шумов.

Программа умеет распознавать жесты, даже если их демонстрировать с использованием различных пальцев. Примеры распознавание жеста «2 пальца» в разных вариациях приведены ниже:

- с помощью указательного и среднего пальцев представлен на рисунке 3.2;
- с помощью указательного пальца и мизинца, как это представлено на рисунке 3.3;
- с помощью указательного и большого пальца, как это представлено на рисунке 3.4.

Как видно из результатов (рисунки 3.2-3.4) тестирования, алгоритм успешно справляется распознаванием различных вариаций жестов, демонстрируемых с использованием различных пальцев.

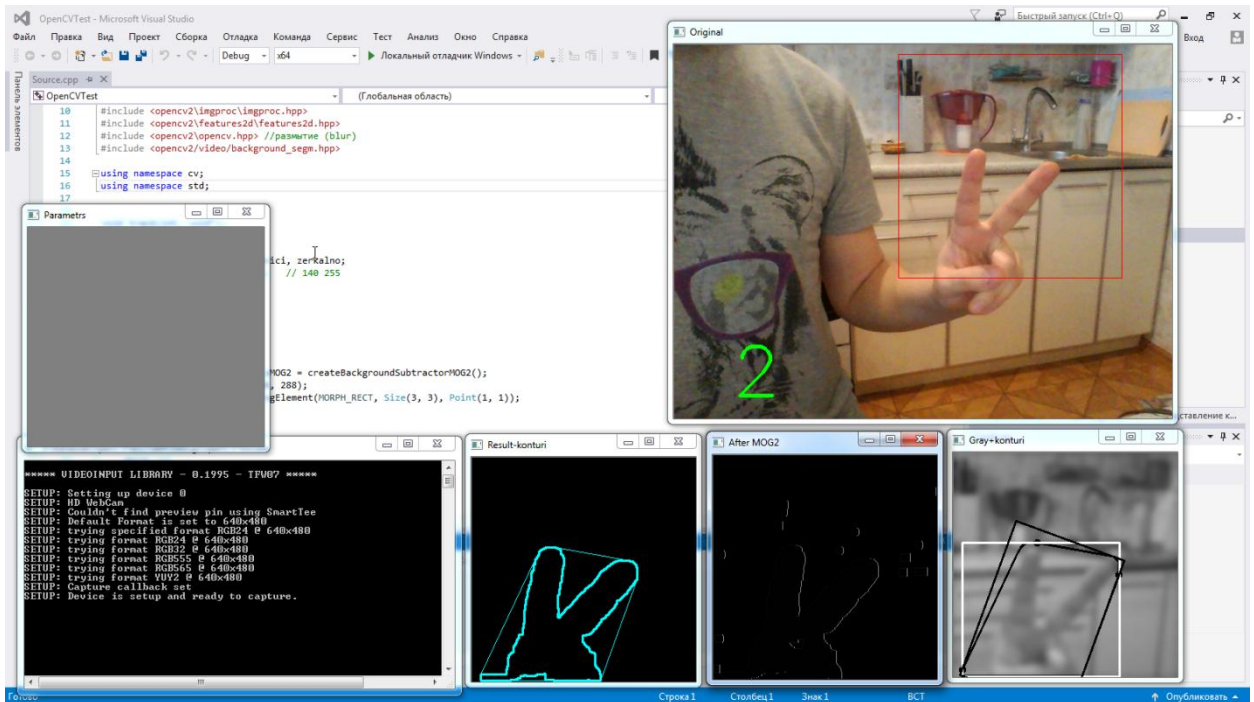


Рисунок 3.2 – Пример распознавания жеста, состоящего из указательного и среднего пальца

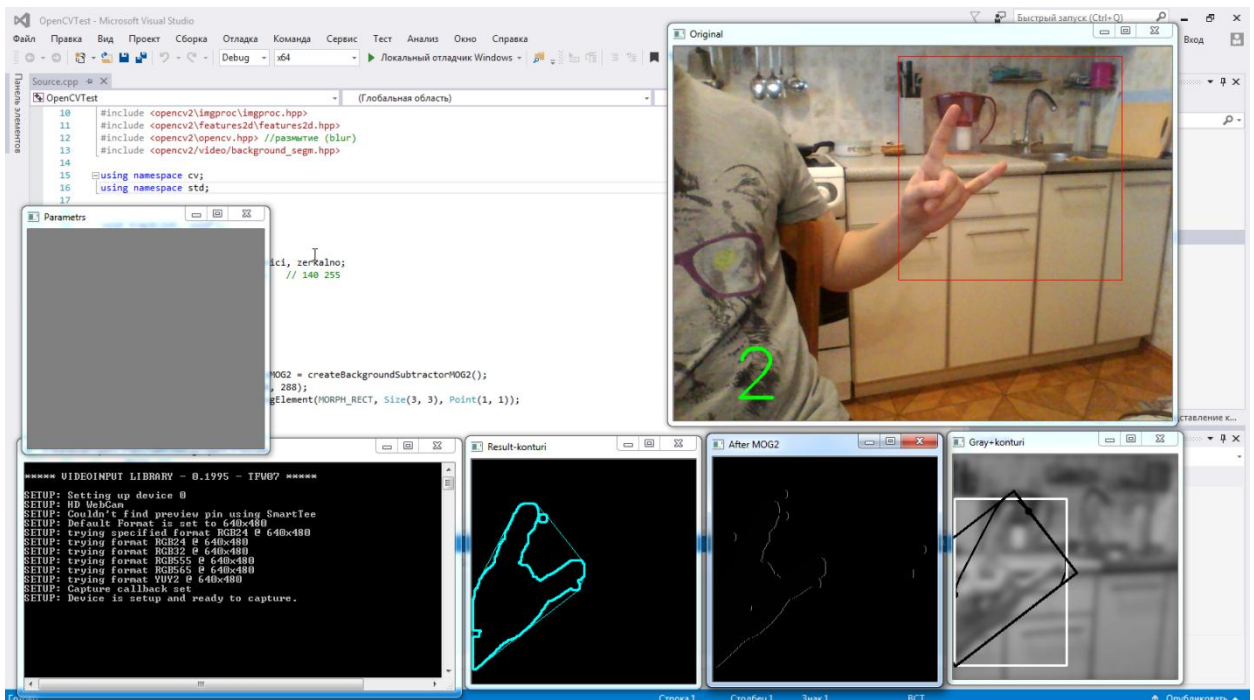


Рисунок 3.3 – Пример распознавания жеста, состоящего из указательного пальца и мизинца

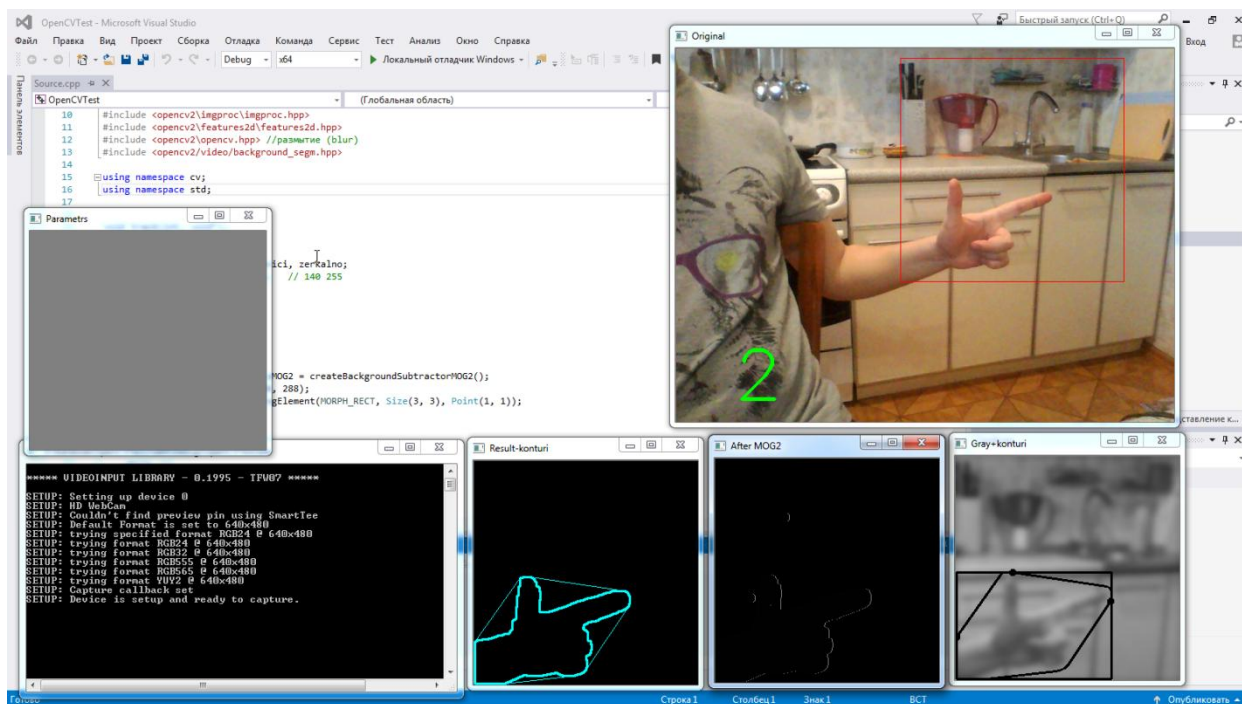


Рисунок 3.4 – Пример распознавания жеста, состоящего из указательного и большого пальца

Также проводилось тестирование алгоритма на возможность распознавания жестов с разным количеством пальцев. Пример распознавания жеста «3 пальца» представлен на рисунке 3.5, распознавание жеста «4 пальца» - рисунок 3.6, распознавание жеста «5 пальцев» - рисунок 3.7.

В процессе тестирования проверялась возможность алгоритма успешно справляться с задачей распознавания жестов в различных пространственных положениях кисти руки. Пример распознавания жестов при перевернутом положении кисти руки представлен на рисунке 3.6.

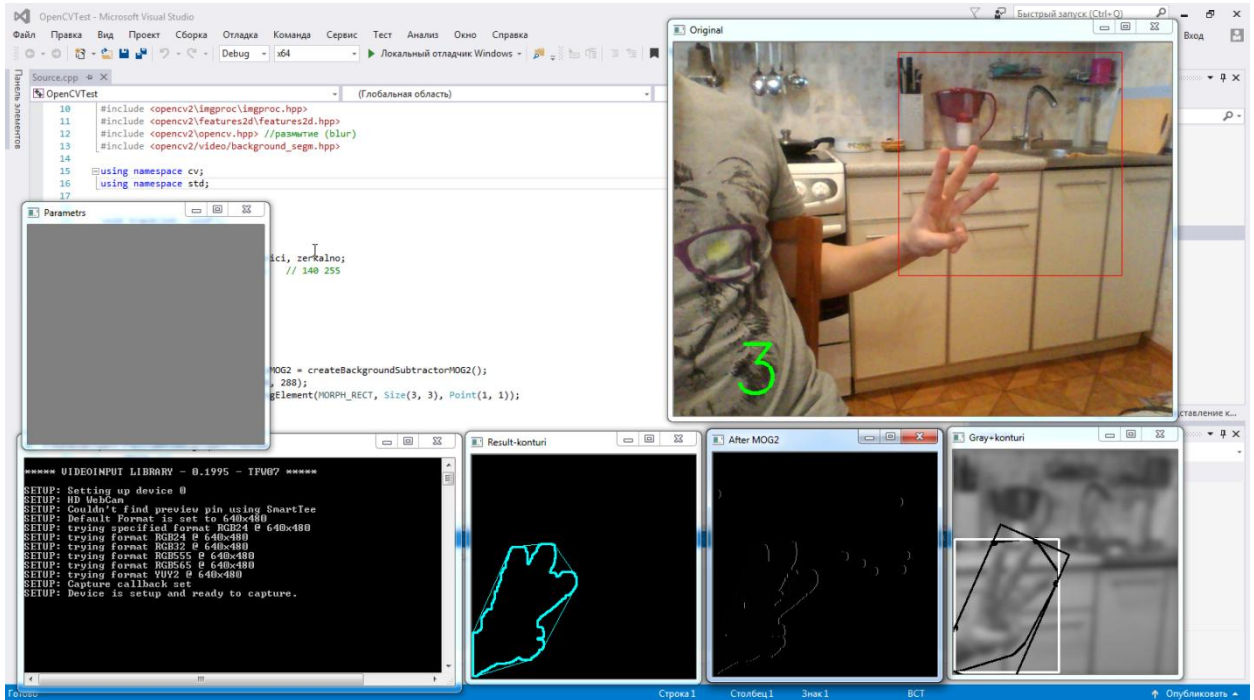


Рисунок 3.5 – Пример распознавания жеста «3 пальца»

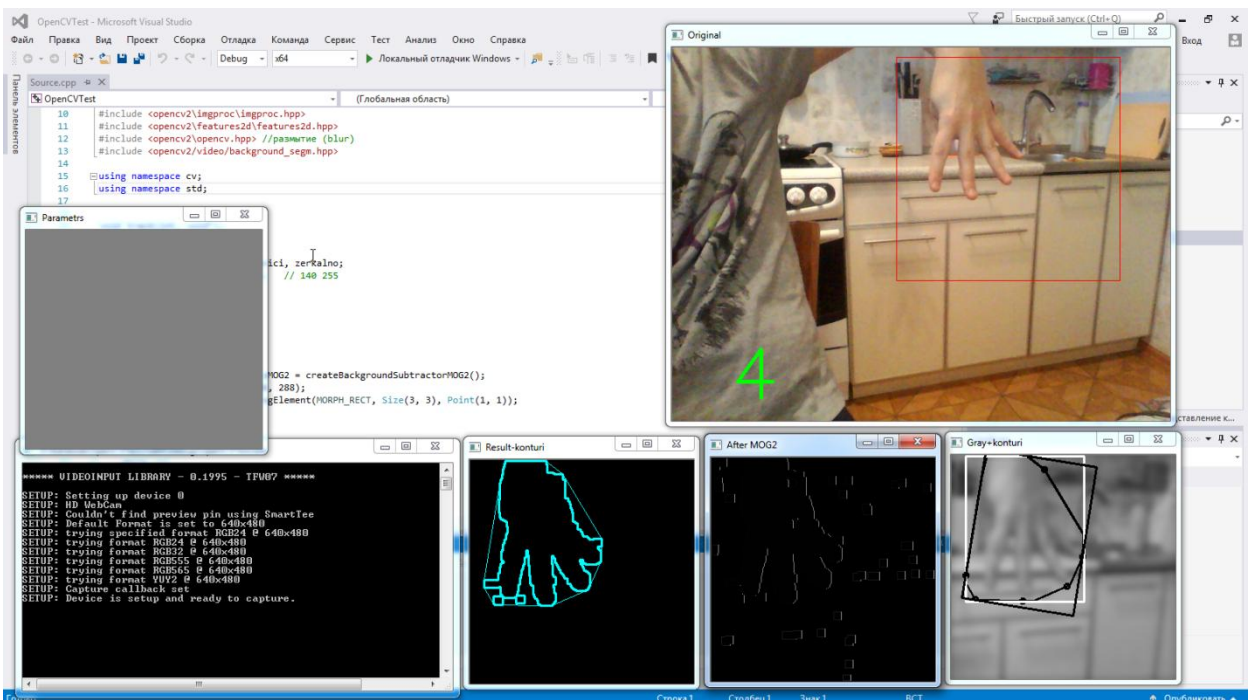


Рисунок 3.6 – Пример распознавания жеста «4 пальца» при перевернутом расположении кисти

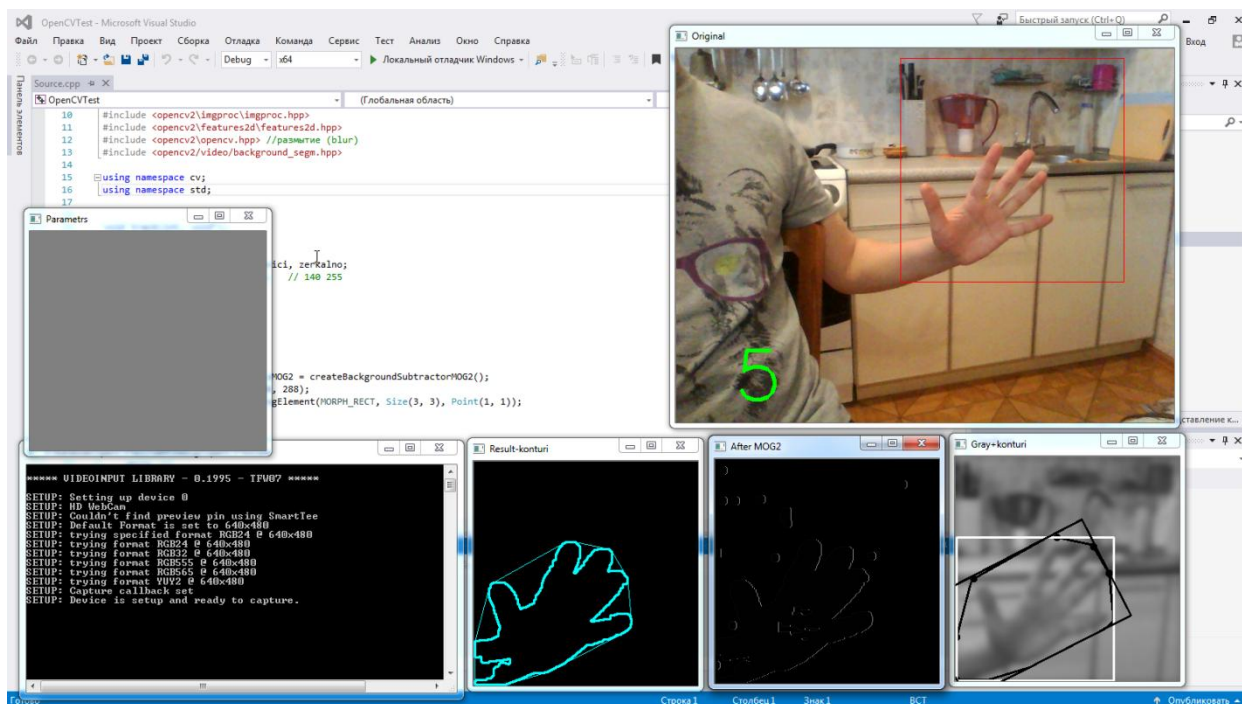


Рисунок 3.7 – Пример распознавания жеста «5 пальцев»

Как видно из результатов тестирования, предложенный алгоритм позволяет успешно решать задачу распознавания жестов кисти руки, расположенной в различных пространственных положениях. При этом, как видно, алгоритм анализа изображения успешно справляется с поставленной задачей даже в условиях недостаточной освещенности сильно неоднородным внешним фоном.

ЗАКЛЮЧЕНИЕ

При выполнении исследований в рамках данной бакалаврской работы были получены следующие результаты:

1. Обзор научных литературных источников показал, что технологии распознавания образов могут применяться при решении практических задач связанных с автоматизированным определением объектов находящихся на изображении и их свойств.

2. Анализ литературных источников показал, что перспективными исследованиями является развитие технологий направленных на создание человеко-машинных интерфейсов, отвечающих за взаимодействие человека с управляемыми машинами. Поэтому выбранная тема бакалаврской работы является актуальной.

3. В ходе выполнения бакалаврской работы разработан алгоритм распознавания жестов кисти руки. Данный алгоритм основан на преобразовании изображения в цветовую модель Gray, применении фильтра Гаусса для очистки изображения от цифровых шумов (посредством размытия), применении субтрактора фона MOG2 (для удаления фона), применения детектора границ Canny (для выделения конура кисти руки), применении алгоритма для анализа контура кисти руки (для определения демонстрируемого жеста).

5. Опытным путем оценена точность распознавания жестов предложенным алгоритмом. Она составляет около 81%. Результаты тестирования показывают, что предложенный алгоритм позволяет успешно решать задачу распознавания жестов кисти руки, расположенной в различных пространственных положениях. При этом, алгоритм анализа успешно справляется с поставленной задачей в условиях недостаточной освещенности сильно неоднородным внешним фоном.

6. В ходе выполнения бакалаврской работы было разработано и протестировано программное обеспечение, реализующее предложенный алгоритм распознавания жестов кисти руки. Особенностью данного

программного обеспечения является возможность просмотра результата на каждом этапе работы алгоритма.

В ходе выполнения бакалаврской работы была опубликована статья «Сегментация изображения с использованием алгоритма k-means» на тему практического применения алгоритмов анализа изображений.

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. Li, G. Image Recognition of Grape Downy Mildew and Grape Powdery Mildew Based on Support Vector Machine [Text] / Guanlin Li, Zhanhong Ma, Haiguang Wang // International Conference on Computer and Computing Technologies in Agriculture – 5th IFIP TC 5/SIG 5.1 Conference, CCTA 2011, Beijing, China, October 29-31, 2011, Proceedings, Part II: Computer and Computing Technologies in Agriculture V. – IFIP International Federation for Information Processing, 2012. – pp. 151-162.

2. Yonekawa, M. An Evaluation of the Image Recognition Method Using Pulse Coupled Neural Network [Text] / Masato Yonekawa, Hiroaki Kurokawa // International Conference on Artificial Neural Networks – 21st International Conference on Artificial Neural Networks, Espoo, Finland, June 14-17, 2011, Proceedings, Part I: Artificial Neural Networks and Machine Learning (ICANN 2011). – Springer-Verlag Berlin Heidelberg, 2011. – pp. 217-224

3. Son, B. Multifocus Image Sequences for Iris Recognition [Text] / Byungjun Son, Sung-Hyuk Cha, Yillbyung Lee // Pacific-Rim Symposium on Image and Video Technology – First Pacific Rim Symposium, PSIVT 2006, Hsinchu, Taiwan, December 10-13, 2006. Proceedings: Advances in Image and Video Technology. – Springer-Verlag Berlin Heidelberg, 2006. – pp. 411-420

4. Li, C. Face Recognition Based on Pose-Variant Image Synthesis and Multi-level Multi-feature Fusion [Text] / Congcong Li, Guangda Su, Yan Shang, Yingchun Li, Yan Xiang // International Workshop on Analysis and Modeling of Faces and Gestures – Third International Workshop, AMFG 2007 Rio de Janeiro, Brazil, October 20, 2007. Proceedings: Analysis and Modeling of Faces and Gestures. – Springer-Verlag Berlin Heidelberg, 2007. – pp. 261-275

5. Mottaleb, M. Assessment of Blurring and Facial Expression Effects on Facial Image Recognition [Text] / Mohamed Abdel-Mottaleb, Mohammad H. Mahoor // International Conference on Biometrics – International

Conference, ICB 2006, Hong Kong, China, January 5-7, 2006. Proceedings: Advances in Biometrics. – Springer-Verlag Berlin Heidelberg, 2005. – pp. 12-18

6. Yan, C. Face Image Gender Recognition Based on Gabor Transform and SVM [Text] / ChunJuan Yan // International Conference on Electronic Commerce, Web Application, and Communication – International Conference, ECWAC 2011, Guangzhou, China, April 16-17, 2011. Proceedings, Part II: Advanced Research on Electronic Commerce, Web Application, and Communication. – Springer-Verlag Berlin Heidelberg, 2011. – pp. 420-425

7. Wang, J. Using Original Face Image and Its Virtual Image for Face Recognition [Text] / Jianguo Wang, Shucai Fu // International Conference on Intelligent Science and Big Data Engineering – 7th International Conference, IScIDE 2017, Dalian, China, September 22-23, 2017, Proceedings: Intelligence Science and Big Data Engineering. – Springer International Publishing AG, 2017. – pp. 231-238

8. Haddad, Z. Image Processing and Pattern Recognition Tools for the Automatic Image Transcription [Text] / Zehira Haddad, Yong Chen, Jaime Lopez Krahe // International Conference on Computers Helping People with Special Needs - 15th International Conference, ICCHP 2016, Linz, Austria, July 13-15, 2016, Proceedings, Part I: Computers Helping People with Special Needs. – Springer International Publishing Switzerland, 2016. – pp. 197-203

9. Chen, W. Palm Image Recognition Using Image Processing Techniques [Text] / Wen-Yuan Chen, Yu-Ming Kuo, Chin-Ho Chung // International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems – 26th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2013, Amsterdam, The Netherlands, June 17-21, 2013. Proceedings: Recent Trends in Applied Artificial Intelligence. – Springer-Verlag Berlin Heidelberg, 2013. – pp. 572-580

10. Tomašev, N. Image Hub Explorer: Evaluating Representations and Metrics for Content-Based Image Retrieval and Object Recognition [Text] / Nenad Tomašev, Dunja Mladenić // Joint European Conference on Machine Learning and Knowledge Discovery in Databases – European Conference, ECML PKDD 2013, Prague, Czech Republic, September 23-27, 2013, Proceedings, Part III: Machine Learning and Knowledge Discovery in Databases. – Springer-Verlag Berlin Heidelberg, 2013. – pp. 637-640

11. Cho, M. Face Recognition Performance Comparison Between Real Faces and Pose Variant Face Images from Image Display Device [Text] / Mi-Young Cho, Young-Sook Jeong // Information and Communication Technology - EurAsia Conference: Third IFIP TC 5/8 International Conference, ICT-EurAsia 2015, and 9th IFIP WG 8.9 Working Conference, CONFENIS 2015, Held as Part of WCC 2015, Daejeon, Korea, October 4-7, 2015, Proceedings: Information and Communication Technology. – IFIP International Federation for Information Processing, 2015. – pp. 229-234

12. Qin, F. Development of a Recognition System for Alfalfa Leaf Diseases Based on Image Processing Technology [Text] / Feng Qin, Haiguang Wang // International Conference on Computer and Computing Technologies in Agriculture – 10th IFIP WG 5.14 International Conference, CCTA 2016, Dongying, China, October 19–21, 2016, Proceedings: Computer and Computing Technologies in Agriculture X. – IFIP International Federation for Information Processing, 2019. – pp. 218-235

13. Li, B. Visual Recognition of Traffic Accident Scene's Image Research [Text] / Bi-Xiang Li, Lan Fang // Proceedings of International Conference on Mechatronics and Intelligent Robotics (ICMIR2018): Recent Developments in Mechatronics and Intelligent Robotics. – Springer Nature Switzerland AG 2019. – pp. 1061-1068

14. Glette, K. An Online EHW Pattern Recognition System Applied to Face Image Recognition [Text] / Kyrre Glette, Jim Torresen, Moritoshi Yasunaga // Workshops on Applications of Evolutionary Computation – EvoWorkshops

2007: EvoCoMnet, EvoFIN, EvoIASP, EvoINTERACTION, EvoMUSART, EvoSTOC and EvoTransLog. Proceedings : Applications of Evolutionary Computing. – Springer-Verlag Berlin Heidelberg, 2007. – pp. 271-280

15. Cho, M. Co-recognition of Image Pairs by Data-Driven Monte Carlo Image Exploration [Text] / Minsu Cho, Young Min Shin, Kyoung Mu Lee // European Conference on Computer Vision - 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part IV: Computer Vision – ECCV 2008. – Springer-Verlag Berlin Heidelberg 2008. – pp. 144-157

16. Valev, V. A model-based image quantization technique for supervised image recognition [Text] / Ventzeslav Valev // International Conference on Computer Analysis of Images and Patterns 5th International Conference, CAIP'93 Budapest, Hungary, September 13–15, 1993 Proceedings: Computer Analysis of Images and Patterns. – Springer-Verlag Berlin Heidelberg 1993. – pp. 128-132

17. Somasundaram, K. Personal ID Image Normalization Using ISO/IEC 19794-5 Standards for Facial Recognition Improvement [Text] / K. Somasundaram, N. Palaniappan // International Conference on Mathematical Modelling and Scientific Computation International Conference, ICMMS 2012, Gandhigram, Tamil Nadu, India, March 16-18, 2012. Proceeding: Mathematical Modelling and Scientific Computation. – Springer-Verlag Berlin Heidelberg 2012. – pp. 429-438

18. Du, Q. Recognition the Target Object Based on More Image Information [Text] / Qinjun Du, Xueyi Zhang, Leping Li // International Conference on Computer Science and Information Engineering – International Conference, CSIE 2011, Zhengzhou, China, May 21-22, 2011, Proceedings, Part II: Advanced Research on Computer Science and Information Engineering. – Springer-Verlag Berlin Heidelberg, 2011. – pp. 497-502

19. Doroshenko, J. Principle and Method of Image Recognition under Diffusive Distortions of Image [Text] / Jaser Doroshenko, Lev Dulkan, Viktor Salakhutdinov, Yury Smetanin // International Conference on Pattern

Recognition and Machine Intelligence – 4th International Conference, PReMI 2011, Moscow, Russia, June 27 - July 1, 2011. Proceedings: Pattern Recognition and Machine Intelligence. – Springer-Verlag Berlin Heidelberg, 2011. – pp. 130-135

20. Mitsukura, Y. The Image Recognition System by Using the FA and SNN [Text] / Seiji Ito, Yasue Mitsukura, Minoru Fukumi, Norio Akamatsu, Sigeru Omatu // International Conference on Knowledge-Based and Intelligent Information and Engineering Systems – 7th International Conference, KES 2003, Oxford, UK, September 2003. Proceedings, Part I: Knowledge-Based Intelligent Information and Engineering Systems. – Springer-Verlag Berlin Heidelberg, 2003. – pp. 578-584