|  |  |  |  |
| --- | --- | --- | --- |
| МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  ВЫСШЕГО ОБРАЗОВАНИЯ | | | |
| «НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ» | | | |
| Кафедра Вычислительной техники | | | |
| **ОТЧЕТ**  **ПО ЛАБОРАТОРНОЙ РАБОТЕ №1**  **по дисциплине «Системы компьютерного зрения»** | | | |
| Факультет: АВТФ  Группа: АММ2-19 |  | | Преподаватель:  Гаврилов А.В. |
| Студент: Беймо Ю. Д., Нишляев Е. В. | |  | |
| Новосибирск  2020 | | | |

# **Цель работы.**

Необходимо разработать программу для фильтрации изображения с помощью матрицы 3 X 3. Подобная программа содержится в архиве Filter.zip

# **Требования.**

1. Пользовательский интерфейс должен обеспечивать загрузку произвольного файла формата .jpg 640х480 и сохранение результирующего файла, задание коэффициентов (целых чисел) в матрице-ядре свертки, вывод исходного файла и результата применения фильтра.
2. Перед обработкой фильтром изображение должно быть преобразовано в полутоновое (с оттенками серого).

# **Теория матричных фильтров**

Матрица свёртки – это матрица коэффициентов, которая «умножается» на значение пикселей изображения для получения требуемого результата.(Рис1.)

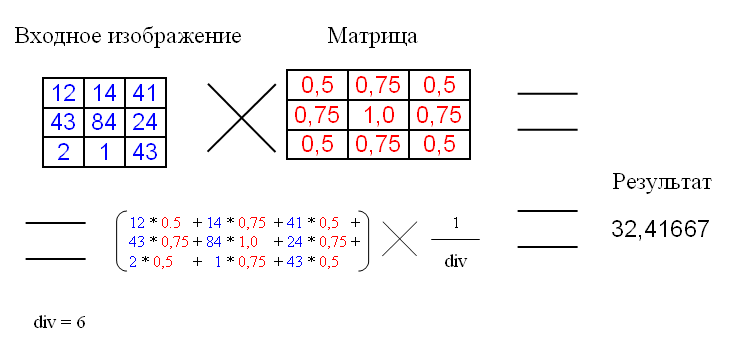


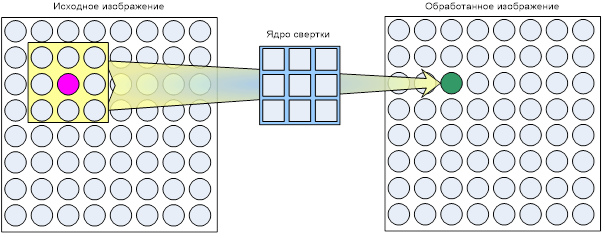
Рисунок1. Пример матрицы свертки.

div – это коэффициент нормирования, для того чтобы средняя интенсивность оставалась не изменой.

## **Фильтр размытияи резкозти**

Наиболее часто используемым фильтром, основанным на матрице свёртки, является фильтр размытия.Обычно матрица заполняется по нормальному (гауссовому закону). Имеет сложность O(hi \* wi \* n \*n), где hi, wi – размеры изображения, n – размер матрицы (ядра фильтра). Квадратное ядро (матрицу) можно заменить двумя одномерными: горизонтальным и вертикальным. Коэффициенты уже являются нормированными, так что div для этой матрицы равен одному.

Эти фильтры реализуются на основе ядра свертки. Элемент изображения получает новое значение на основе группы элементов, примыкающих к данному. Область примыкания есть квадратная матрица, размерность которой совпадает с размером выбранного ядра свертки, и центром в обрабатываемом элементе.

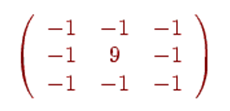


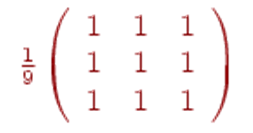
Ядро свертки представляет собой матрицу размером 3х3, 5х5, 7х7 и т.д., на которой определена некоторая функция. Ядро свертки называется окном, а заданная на нем функция – весовой или функцией окна. Каждому элементу окна соответствует число, называемое весовым множителем. Совокупность всех весовых множителей и составляет весовую функцию. Нечетные размеры окна необходимы для однозначного определения центрального элемента.

По сути, ядро свертки является фильтром, который позволяет усилить или ослабить компоненты изображения. Фильтрация осуществляется перемещением окна фильтра по изображению. Весовая функция в процессе перемещения остается неизменной. В каждом положении окна происходит операция свертки. При каждом положении окна весовая функция поэлементно умножается на значение соответствующих пикселей исходного изображения и произведения суммируются. Полученная сумма называется откликом фильтра и присваивается тому пикселю нового изображения, который соответствует положению центра окна.

Результат обработки пикселя записывается в соответствующую ячейку временной матрицы такого же размера, как и исходное изображение. Запись в отдельную временную матрицу необходима для того, чтобы исключить влияние уже обработанных пикселей на еще не обработанные.

При использовании алгоритма увеличения резкости подчеркиваются различия между цветами смежных пикселей и выделяются незаметные детали. В ядре резкости центральный коэффициент больше 1, а окружен он отрицательными числами, сумма которых на единицу меньше центрального коэффициента. Таким образом, увеличивается любой существующий контраст между цветом пикселя и цветами его соседей. При обработке каждого пикселя в изображении используется ядро резкости размерами 3×3:



При использовании алгоритма размытия в изображении перераспределяются цвета и смягчаются резкие границы. Ядро сглаживания состоит из совокупности коэффициентов, каждый из которых меньше 1, а их сумма составляет 1. Это означает, что после фильтрации каждый пиксель поглотит что-то из цветов соседей, но полная яркость изображения останется неизменной. При обработке изображения используется следующее ядро свертки:

## **3.2 Граничные условия**

Стоит упомянуть о граничных условиях (эта проблема актуальна для всех матричных фильтров). У верхнего левого пикселя не существует «соседа» с права от него, следовательно, нам не на что умножать коэффициент матрицы. (Рис. 3)

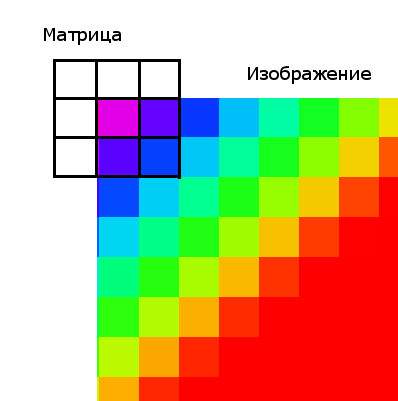


Рисунок 3. Проблема граничных условий.

Для решения этой проблемы требуется создание промежуточного изображения. Идея в том, чтобы создавать временное изображение с размерами (width + 2 • gap / 2, height + 2 • gap / 2, где width и height – ширина и высота фильтруемого изображения, а gap – размерность матрицы свертки). В центр изображения копируется входная картинка, а края заполняются крайними пикселями изображения. Размытие применяется к промежуточному буферу, а потом из него извлекается результат. Рис.4

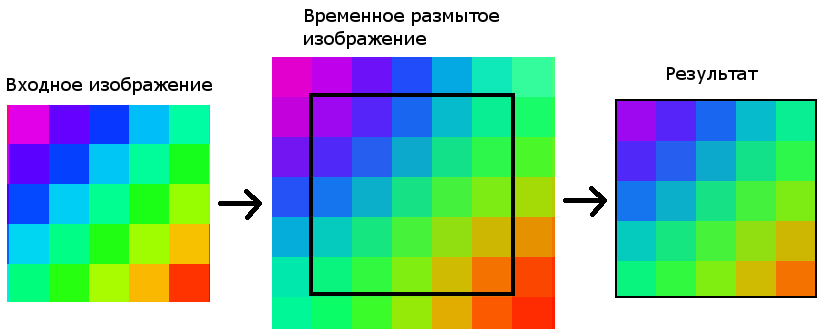


Рисунок 4. Промежуточный буфер.

Данный метод не имеет недостатков в качестве, но необходимо производить лишние вычисления.

## **3.3 Выделения контуров изображений**

* Фильтр Робертса

Пусть область 3х3, показанная на рисунке ниже (см. рис. 2), представляет собой значения яркости в окрестности некоторого элемента изображения.

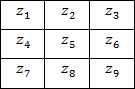


Рисунок 5. Окрестность 3х3 внутри изображения

Один из простейших способов нахождения первых частных производных в точке image состоит в применении следующего перекрестного градиентного оператора Робертса:

image

**image**

Эти производные могут быть реализованы путем обработки всего изображения с помощью оператора, описываемого масками на рисунке 6.

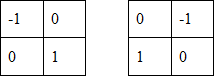


Рисунок 6. Маски оператора Робертса

Реализация масок размерами 2х2 не очень удобна, т.к. у них нет четко выраженного центрального элемента, что существенно отражается на результате выполнения фильтрации. Но этот «минус» порождает очень полезное свойство данного алгоритма – высокую скорость обработки изображения.

* Фильтр Превитта

Оператор Превитта, так же как и оператор Робертса, оперирует с областью изображения 3х3, представленной на рисунке 2, только использование такой маски задается другими выражениями:

image

image

В этих формулах разность между суммами по верхней и нижней строкам окрестности 3х3 является приближенным значением производной по оси x, а разность между суммами по первому и последнему столбцам этой окрестности – производной по оси y. Для реализации этих формул используется оператор, описываемый масками на рисунке 7, который называется оператором Превитта.

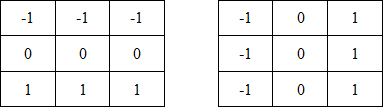


Рисунок 7. Маски оператора Превитта

* Фильтр Собеля

Оператор Собеля тоже использует область изображения 3х3, отображенную на рисунке 2. Он довольно похож на операторПревитта, а видоизменение заключается в использовании весового коэффициента 2 для средних элементов:

image

image

Это увеличенное значение используется для уменьшения эффекта сглаживания за счет придания большего веса средним точкам.

Маски, используемые оператором Собеля, отображены на рисунке 8.

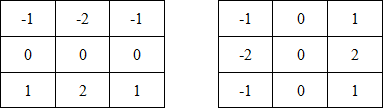


Рисунок 8. Маски оператора Собеля

Рассмотренные выше маски применяются для получения составляющих градиента Gxи Gy. Для вычисления величины градиента эти составляющие необходимо использовать совместно:

image

image ИЛИ

Т.к. значения могут быть отрицательными.

# **Программный комплекс**

Для создания приложения было принято решение использовать язык C#, а в качестве среды разработки использоваться MicrosoftVisualStudio (версия Community 2019), так как она является бесплатной и предоставляет весь необходимый функционал для разработки программ на языке программирования C#.

Для построения пользовательского интерфейса применяется графическая подсистема

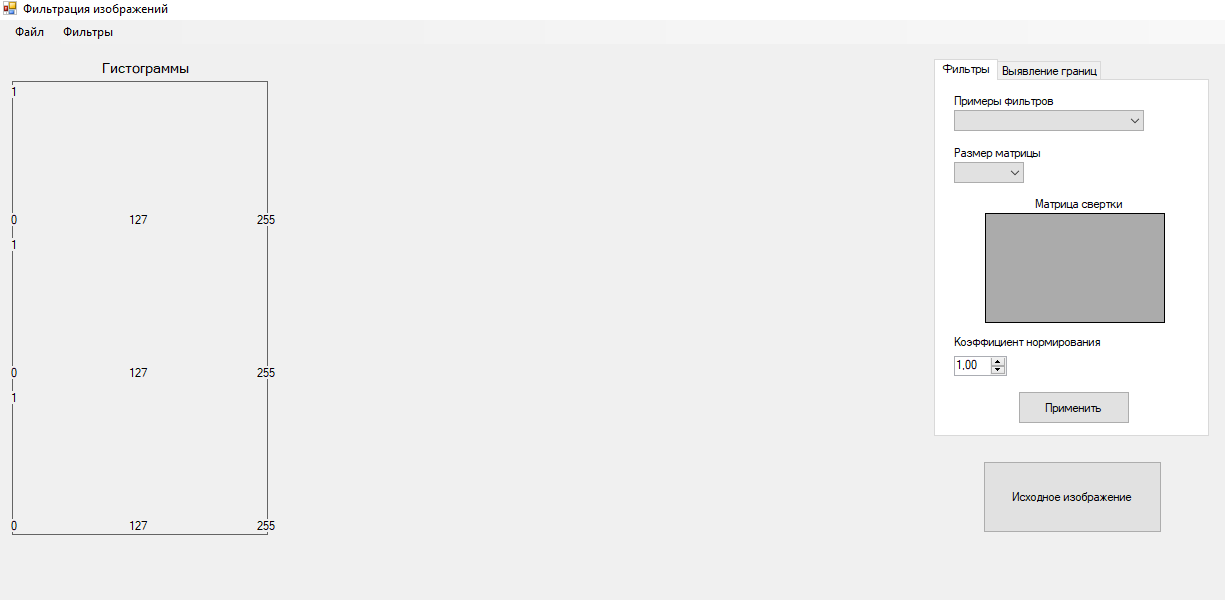


Рисунок 9.

На рисунках ниже представлено главное меню программы. Во вкладке «Файл» находятся кнопки для открытия и сохранения изображения.

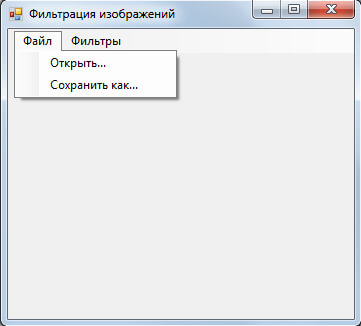


Рисунок 10

В правой части интерфейса представлены инструменты для фильтрации загруженного изображения.

Во вкладке «Фильтры» находятся: Фильтр размытия, повышения резкости. Можно увидеть матрицу свертки, коэффициент нормирования и размер матрицы. Так же если нужно можно подкорректировать эти параметры.Кнопка «применить» запускает процесс фильтрации изображения.

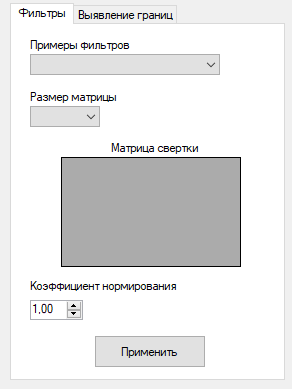


Рисунок 11

Во вкладке «выявления границ» находятся фильтры выявления контуров изображения, описанные выше. Можно увидеть маски, используемые операторами (Gxи Gy), предел и размер ядра. Так же если нужно можно подкорректировать эти параметры.Кнопка «применить» запускает процесс фильтрации изображения.

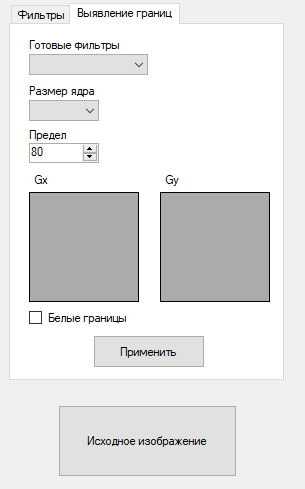


Рисунок 12

Кнопка «Исходное изображение» возвращает картинку к первоначальному виду.

В левой части интерфейса представлены Гистограммы RGBизображения.

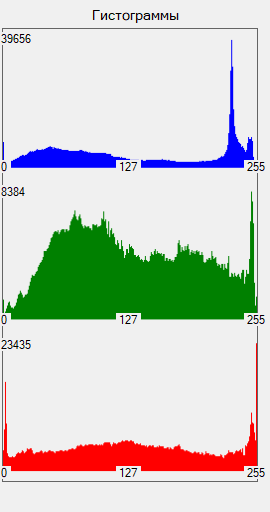


Рисунок 13

# **Пример работы программы**

**5.1 Фильтр размытия и резкости.**

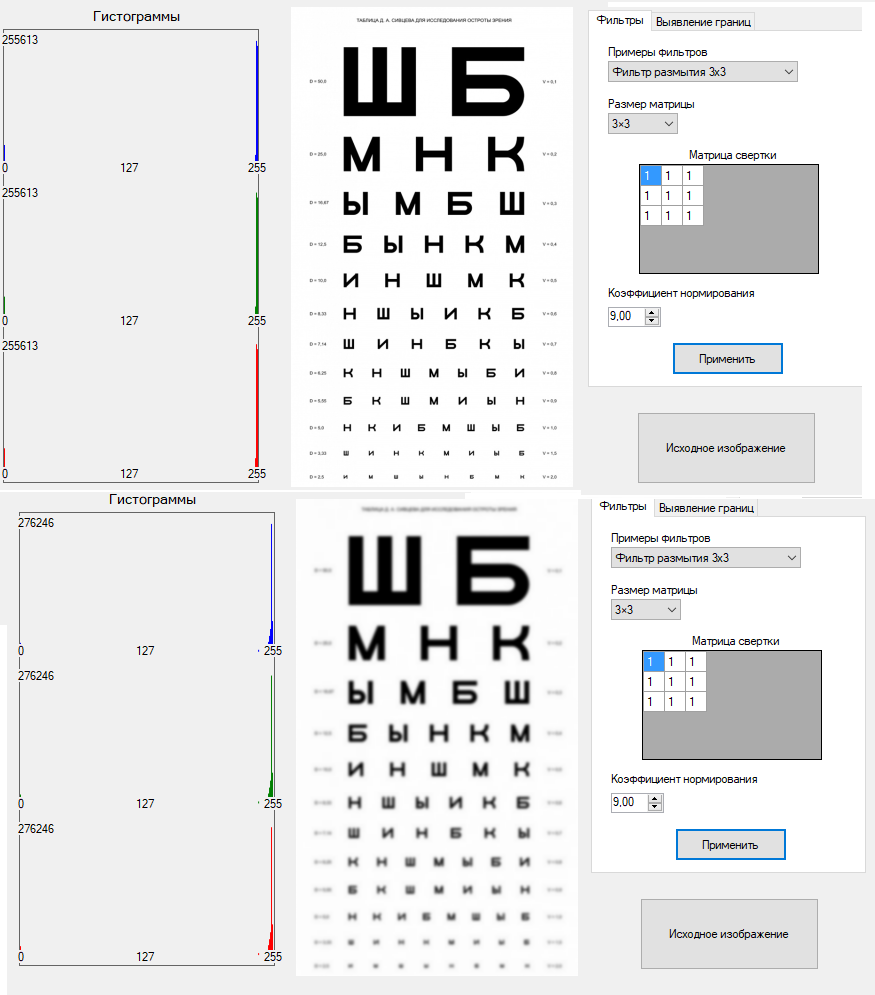


Рисунок 14. Применяем фильтр размытия.

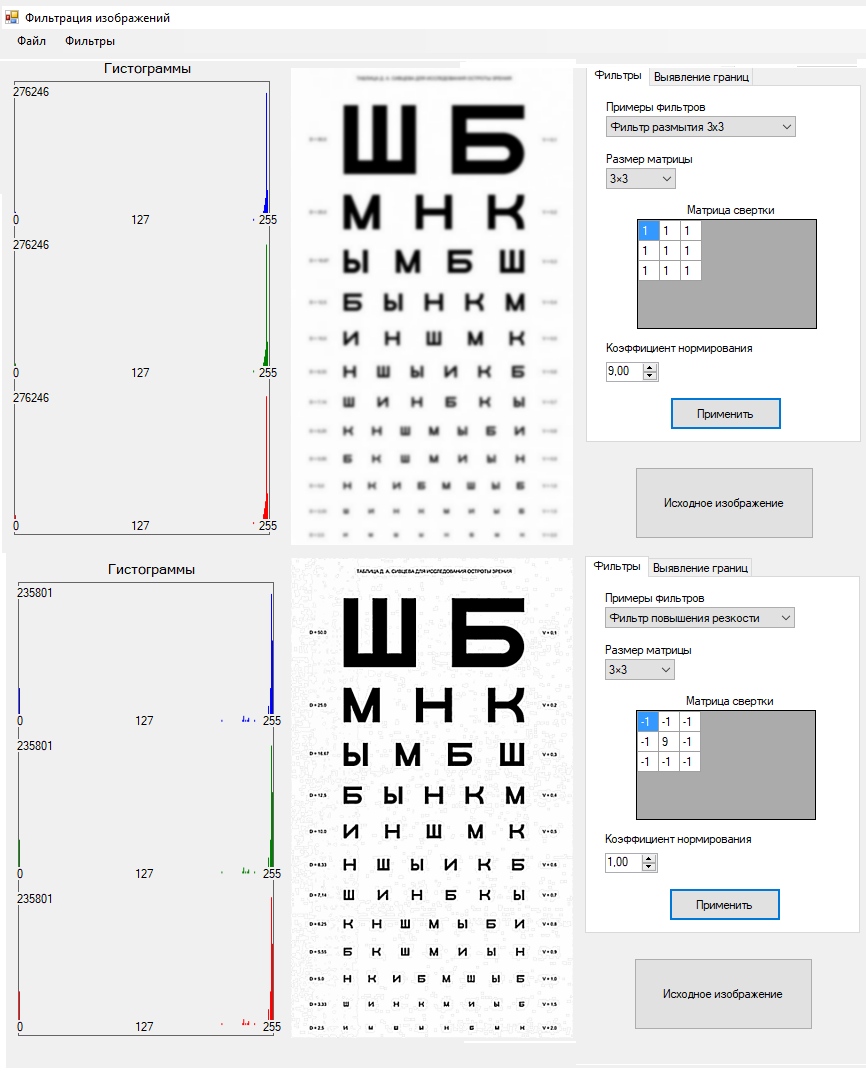


Рисунок 15. Применяем фильтр резкости.



Рисунок 16. Применяем фильтр резкости для распознания номера авто.

Благодаря фильтру можно уже различить буквы в номере ТС.

**5.2 Фильтры выявления границ**

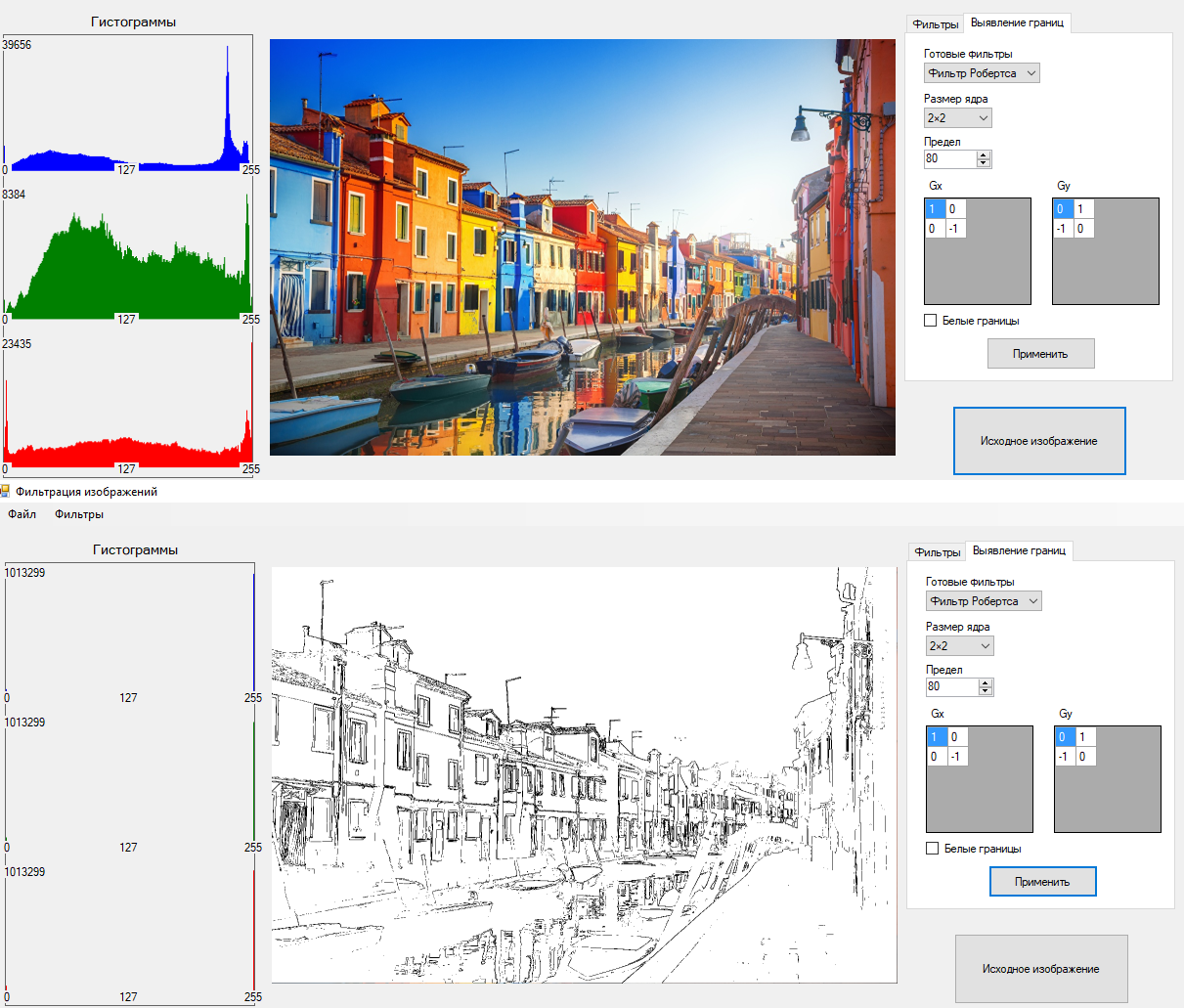


Рисунок 17. Фильтр Робертса.



Рисунок 21. Фильтр Робертся (предел= 30)

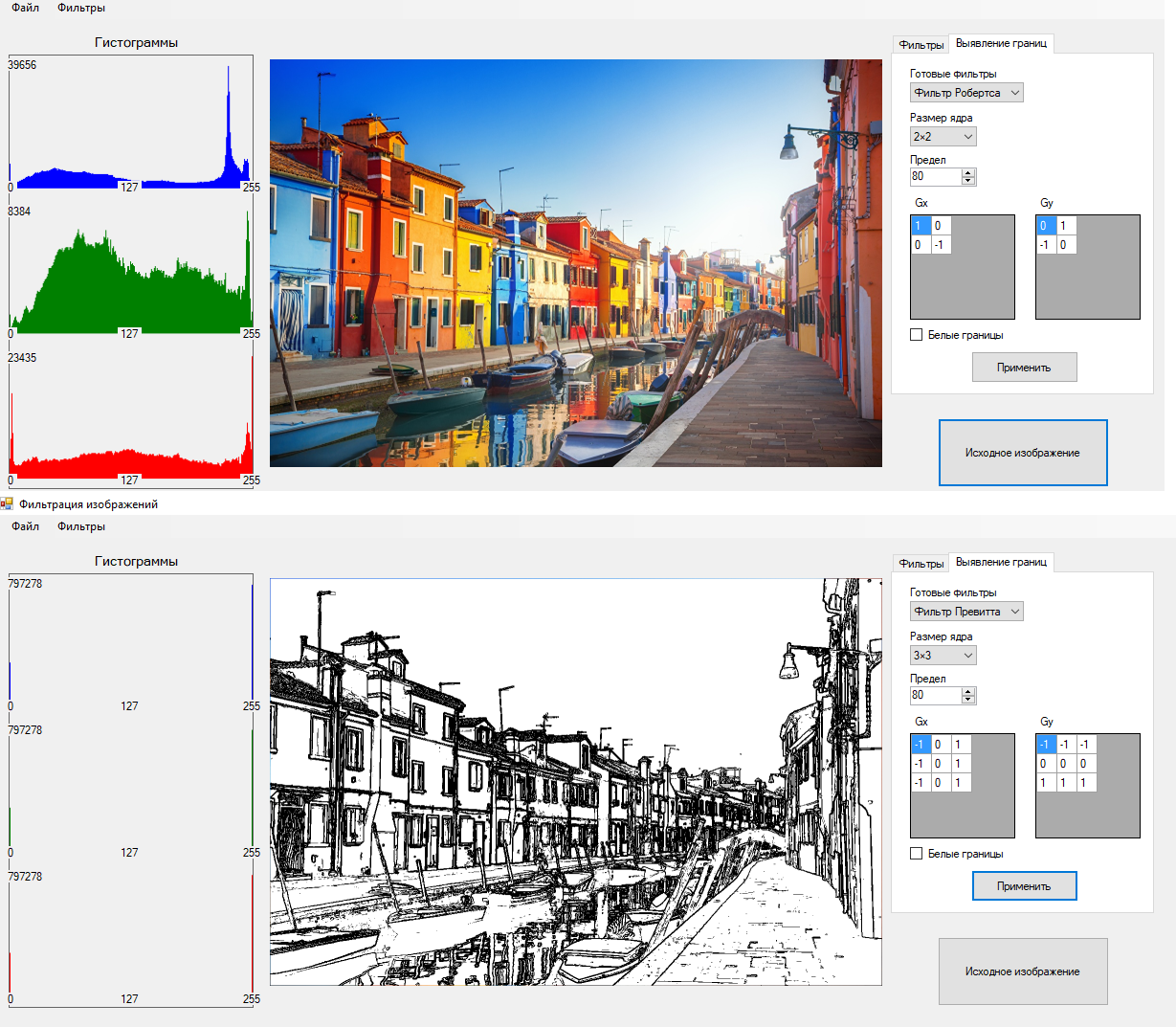


Рисунок 18. Фильтр Превитта.

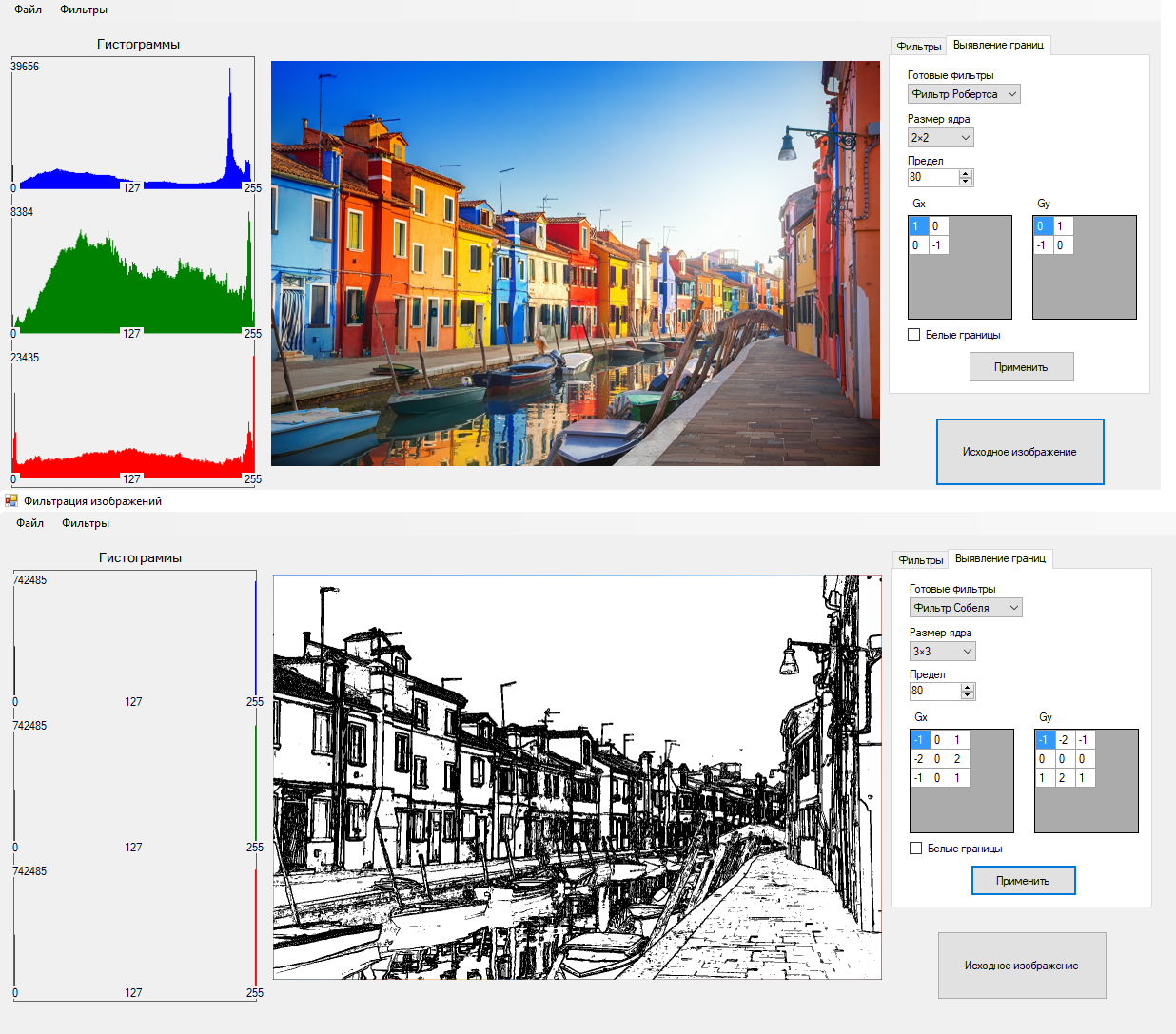


Рисунок 19. Фильтр Собеля.

# **Заключение**

В данной лабораторной работе мы познакомились с наиболее распространёнными принципами обработки изображений и алгоритмами их работы. На этой основе был разработан программный комплекс с пользовательским интерфейсом для применения на практике изученных фильтров обработки изображений. Были реализованы фильтры размытия и резкости на основе ядра свертки и фильтры границ на основе принципа скользящей маски (а именно фильтры Превитта, Собеля и Робертса).

# **Приложение А**

Архив с проектом приложения:

SKZ\_LR1\_Beimo\_Nishlyaev\_AMM2-19.rar