Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра вычислительных технологий**

**ЛАБОРАТОРНАЯ РАБОТА №3**

**Дисциплина: Алгоритмы цифровой обработки мультимедиа**

Работу выполнил: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Д.А. Спиридонов

Направление подготовки: 02.03.02 Фундаментальная информатика и информационные технологии

Преподаватель: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_А.А. Крамаренко

Лабораторная работа №3

Методы фильтрации и размытия изображения.

Размытие – один из базовых алгоритмов обработки изображения. Размытие строится из предположения, что каждый пиксель картинки по своему значению связан с предыдущим и не должен заметно от него отличатся. Если же так происходит, например, вокруг пикселя яркость 240, а в самом пикселе 12, то мы предполагаем, что данный пиксель – это шумы. И размытие картинки – один из главных способов нивелировать эти шумы. Базовый подход к фильтрации строится на том, что значение пикселя должно изменится под влиянием окружающих пикселей.

Для начала необходимо перевести изображение в черно-белый формат, в этом случае каждый пиксель представляет собой только одно целое неотрицательное значение длиной 8 бит – яркость.

Алгоритм кратко будет выглядеть следующим образом:

1. установить размер матрицы свертки и среднее квадратичное отклонение;
2. заполнить матрицу свертки значениями функции Гаусса с мат.

ожиданием, равным координатам центра матрицы;

1. нормировать матрицу так, чтобы сумма элементов равнялась 1;
2. создать копию изображения;
3. для каждого внутреннего пикселя копии рассчитать новое значение
4. насыщенности пикселя по формуле (1) и ЗАПИСАТЬ это значение в пиксель нового изображения

**Задание 1**. Выполнить пункты 1 и 2 алгоритма, то есть построить

матрицу Гаусса. Просмотреть итоговую матрицу для размерностей 3, 5, 7.

Для начала мы задаем значение размера ядра свертки. Обычно он задается как нечетное число (например, 3, 5, 7), чтобы матрица имела центр. Чем больше размер ядра, тем более широкий радиус учета пикселей, и, следовательно, более сильное размытие.

Далее матрица может быть инициализирована любыми значениями, в нашем случае - единицами. После чего мы заполняем ее значениями, которые вычисляются по формуле Гаусса. (смотреть рисунок 1)

Рисунок 1 – формула Гаусса

В данной формуле 𝑎, 𝑏 – координаты центрального элемента матрицы (математическое ожидание двумерной случайной величины). 𝑥, 𝑦 – индексы всех элементов матрицы, а 𝜎 – параметром размытия по Гауссу. Этот параметр определяет "ширину" распределения Гаусса и, следовательно, степень размытия. Чем больше значение стандартного отклонения, тем менее интенсивное размытие.

**Задание 2.** Нормировать полученную матрицу Гаусса. Протестировать

результаты на матрицах из предыдущего пункта.

Нормирование матрицы состоит из двух шагов:

1) Подсчет суммы всех элементов матрицы;

2) Деление всех элементов на полученную сумму.

Далее на рисунках 2-4 представлены матрицы/нормированные матрицы.

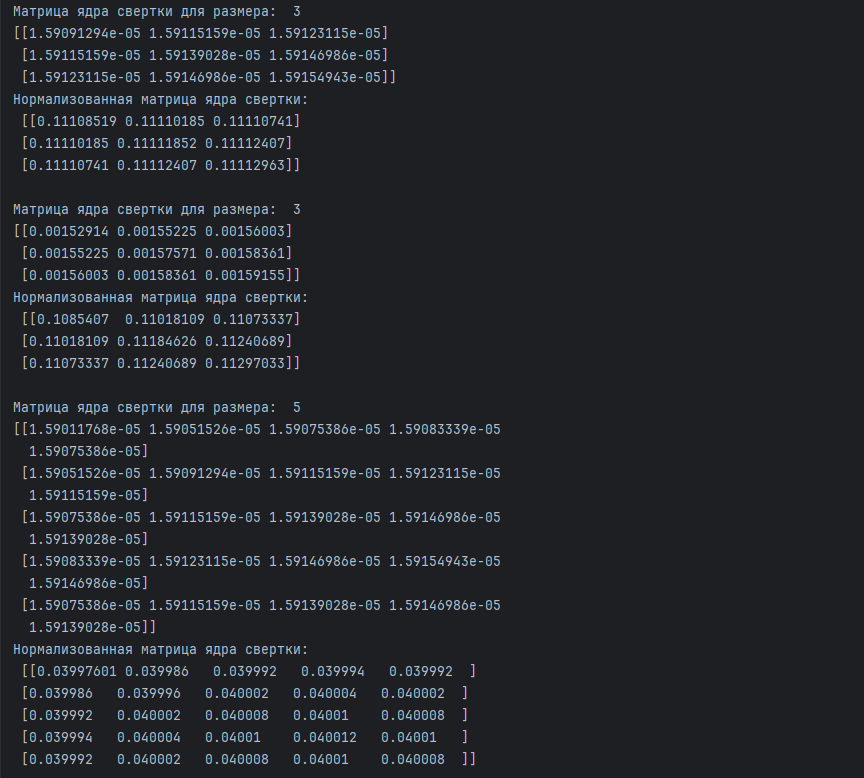


Рисунок 2 - полученные значения матриц

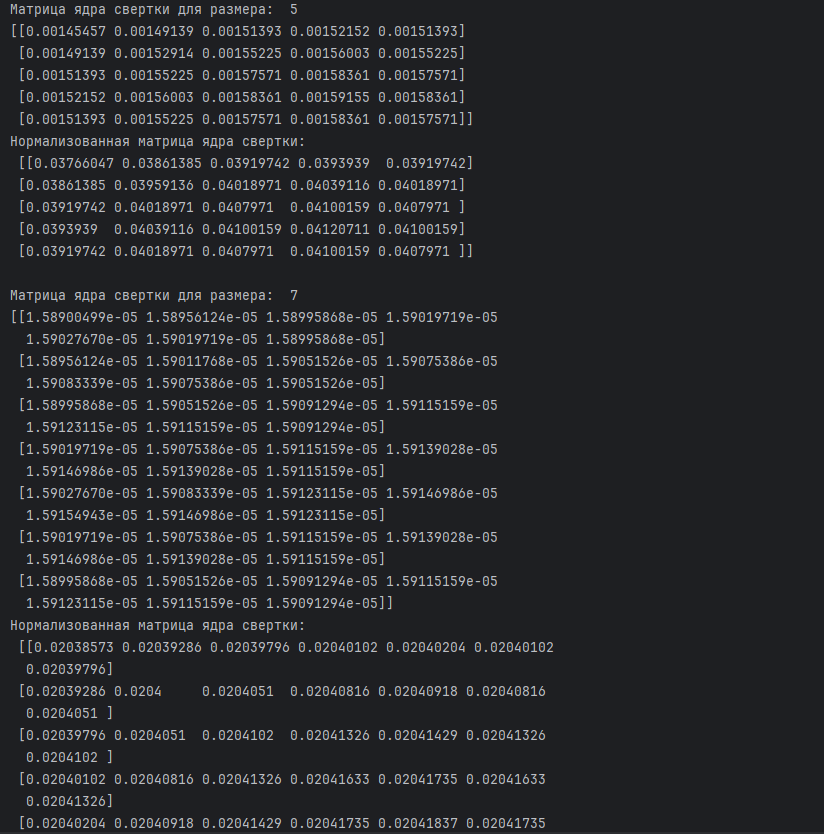


Рисунок 3 - полученные значения матриц

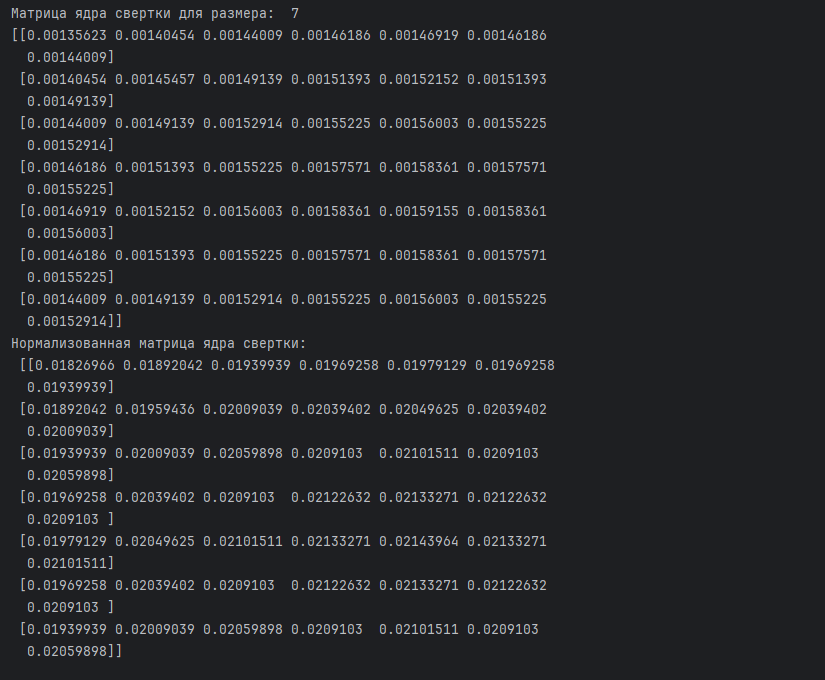


Рисунок 4 - полученные значения матриц

**Задание 3**. Реализовать фильтр Гаусса средствами языка Python.

Теперь можно приступить к операции свертки. Операция свертки заключается в преобразовании исходной матрицы размерности 𝑛 × 𝑛 в числовое значение с помощью специальной матрицы ker размерности 𝑛 × 𝑛, называемой ядром свертки.

Основные шаги операции свертки:

1. Ядро свертки перемещается по изображению;
2. Для каждого пикселя в изображении ядро позиционируется так, чтобы его центр совпадал с текущим пикселем;
3. Выполняется элементное умножение (поэлементное перемножение) значений пикселей изображения и значений соответствующих элементов ядра свертки.
4. Результаты элементных умножений суммируются, и эта сумма становится новым значением центрального пикселя в результирующем изображении.
5. Процесс повторяется для каждого пикселя в исходном изображении, и на выходе получается новое изображение. (смотреть рисунок 5)



Рисунок 5 – применение фильтра Гаусса.

**Задание 4.** Применить данный фильтр для двух разных значений

среднего квадратичного отклонения и двух разных размерностей матрицы свертки, сравнить результаты для ОДНОГО изображения.

На рисунках ниже можно увидеть, что при большем значении ядра матрицы свертки эффект размытия увеличивается, а при меньшем уменьшается. Значение стандартного отклонения работает, наоборот, при увеличении отклонения эффект размытия будет уменьшаться, при уменьшении-увеличиваться. (смотреть рисунок 6)





Рисунок 6 - применение разных значений параметров.

**Задание 5.** Реализовать размытие Гаусса встроенным методом библиотеки OpenCV, сравнить результаты с Вашей реализацией.

Для данного задания была использована функция «cv2.GaussianBlur». Результаты получились почти одинаковые. (смотреть рисунок 7)



Рисунок 7 - применение встроенного размытия Гаусса с помощью библиотеки OpenCV.