

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Государственное образовательное учреждение высшего профессионального образования

**«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ»**

НИИ Информационно-управляющих систем

Соловьев Н.В., Сергеев А.М.

**Улучшение качества
растровых изображений**

Учебное пособие

Санкт-Петербург

2010

УДК
ББК

Рецензенты:

кафедра биомедицинской электроники и охраны среды Санкт-Петербургского
государственного электротехнического университета «ЛЭТИ»;
доктор технических наук, профессор Красильников Н.Н.

Публикуется по решению НТС НИИ Информационно-управляющих систем
СПбГУ Информационных технологий, механики и оптики

Н.В.Соловьев, А.М.Сергеев

С40 Улучшение качества растровых изображений: Учеб. пособие. – СПб.:
СПбГУ ИТМО, 2010. – 158 с.

ISBN

Рассматриваются некоторые методы компьютерной обработки растровых изображений в пространственной области, применяемые для улучшения качества изображений и повышения различимости отдельных фрагментов. Приведены различные методы подавления яркостных помех, повышения контрастности и резкости изображений, а также фильтры для выделения контурных линий. Показаны особенности программной реализации методов обработки изображений в среде визуального объектно-ориентированного программирования Delphi.

Пособие предназначено для студентов, обучающихся по направлению «Информатика и вычислительная техника» и может использоваться при изучении дисциплин «Компьютерная графика» и «Цифровая обработка и распознавание изображений».

УДК

ББК

ISBN

© НИИ Информационно-управляющих систем

© Н.В.Соловьев, А.М.Сергеев

Содержание

Введение	4
1. Подавление яркостных помех	6
2. Выделение контурных линий	22
3. Повышение различимости фрагментов	30
4. Программная реализация методов обработки изображений	46
5. Задание к практическим занятиям	54
Литература	57

Введение

Многие отрасли техники, имеющие отношение к получению, обработке, хранению и передаче информации, в настоящее время ориентируются на развитие систем, в которых информация представляется в виде изображений. Изображение, которое можно рассматривать как двумерный сигнал, является значительно более емким носителем информации, чем обычный одномерный, чаще всего временной, сигнал. Вместе с тем, решение научных и инженерных задач при работе с визуальными данными требует особых подходов, опирающихся на знание специфики изображений, поскольку традиционные способы обработки и анализа одномерных сигналов мало пригодны в этих случаях.

Одной из основных проблем, возникающих при обработке изображений, является улучшение качества изображения, в частности, повышение различимости отдельных фрагментов. К причинам, понижающим качество изображений, можно отнести:

- технические яркостные помехи шумового характера;
- недостаточная или излишняя освещенность объектов съемки;
- отсутствие резкости при получении изображения;
- слишком мелкие размеры деталей, которые необходимо различить.

Основной целью компьютерной обработки изображений является нахождение методов, результат работы которых оказался бы более подходящим с точки зрения конкретного применения. Когда изображение обрабатывается для визуальной интерпретации, наблюдатель является окончательным судьей того, насколько хорошо действует конкретный метод. Визуальное оценивание качества изображения является крайне субъективным процессом, делающим понятие «хорошего изображения» некоторым абстрактным эталоном, с помощью которого необходимо сравнивать эффективность алгоритмов. Если целью является обработка изображения для восприятия другими компьютерными программами, то проблема оценивания несколько упрощается. Например, в задаче распознавания символов наилучшим будет тот метод обработки изображений, который дает бо-

лее точные результаты компьютерного распознавания. Тем не менее, даже в ситуации, когда проблема позволяет установить четкие критерии качества, обычно требуется некоторое количество попыток тестирования, пока не будет выбран наилучший метод и подобраны соответствующие параметры.

Множество методов улучшения качества изображений распадается на две большие категории: методы обработки в пространственной области и методы обработки в частотной области. Термин пространственная область относится к плоскости растрового изображения, состоящего из отдельных пикселей, и объединяет методы, основанные на прямом манипулировании пикселями изображения. Методы обработки в частотной области основываются на модификации сигнала, формируемого путем применения к изображению преобразования Фурье или других аналогичных преобразований. Некоторые методы удобно формулируются в пространственной области, а для формулировки других больше подходит частотная область. В дальнейшем рассматриваются только пространственные методы повышения качества изображений, что в основном связано с необходимостью значительного объема вычислений при выполнении преобразований Фурье для растровых изображений больших размеров (800×600 пикселей и более).

Яркостные помехи могут существенно снижать различимость отдельных фрагментов изображения, но следует отметить, что для современных систем получения изображения в цифровом виде характерно практически полное отсутствие сколько-нибудь существенных яркостных помех, однако полученные с их помощью изображения все же могут иметь недостаточную различимость отдельных фрагментов, что в основном обуславливается:

- недостаточной контрастностью изображения, т.к. известно, что глаз человека не в состоянии различить границу между фрагментами, яркость которых отличается на одну градацию [1];
- недостаточной резкостью изображения, что приводит к размыванию границ между фрагментами.

1. Подавление яркостных помех

1.1 Причины возникновения яркостных помех

Основными источниками яркостных помех или шума на цифровом изображении являются: процесс его получения, оцифровка, а также процесс передачи. Работа сенсоров, реагирующих на электромагнитное излучение в видимом спектральном диапазоне, зависит, как от внешних условий в процессе получения изображения, так и от качества самих сенсоров. Например, в процессе получения изображения с помощью фотокамеры с ПЗС матрицей, основными факторами, влияющими на величину шума, являются уровень освещенности и температура сенсоров [2].

В процессе передачи изображения могут искажаться помехами, возникающими в каналах связи. Например, при передаче изображения с использованием беспроводной связи, оно может быть искажено в результате разряда молнии или других возмущений в атмосфере.

Как показано на рисунке 1.1, модель процесса искажения предполагает действие некоторого искажающего оператора H на исходное изображение $f(x,y)$ и добавление аддитивного шума $\eta(x,y)$, что дает искаженное изображение $z(x,y)$. Здесь и далее под $f(x,y)$ и $z(x,y)$ понимается значение яркости изображения в точке или пикселе (в случае растрового изображения) с координатами x,y . Задача восстановления состоит в построении некоторого приближения $z'(x,y)$ исходного изображения $f(x,y)$ по искаженному изображению $z(x,y)$, при наличии некоторой информации как относительно искажающего оператора, так и шума [3]. Несомненно – чем больше мы знаем об операторе H и функции $\eta(x,y)$, тем ближе функция $z'(x,y)$ будет к $f(x,y)$.

Если H – линейный оператор, то искаженное изображение может быть представлено в пространственной области в виде:

$$z(x,y) = h(x,y)*f(x,y)+\eta(x,y), \quad (1.1)$$

где $h(x,y)$ — функция, представляющая искажающий оператор в пространственной области (ядро фильтра), а символ «*» используется для обозначения свертки.

В дальнейшем предполагается, что шум не зависит от пространственных координат, и между значениями элементов изображения и значениями шумовой составляющей нет корреляции. В таком случае поведение шума основывается на статистических свойствах значений яркости компоненты шума в модели, показанной на рисунке 1.1. Эти значения могут рассматриваться как случайные величины, характеризующиеся функцией плотности распределения вероятностей.

Если искажение изображения обусловлено исключительно наличием шума, то равенство (1.1) приобретает вид $z(x,y) = f(x,y) + \eta(x,y)$.

Так как слагаемое, описывающее шум, неизвестно, то просто вычесть его из функции $z(x,y)$ невозможно. В тех ситуациях, когда на изображении присутствует только аддитивный шум, пространственная фильтрация является лучшим из возможных методов его подавления.

Основные виды пространственных фильтров можно разделить на две группы: усредняющие и упорядочивающие, причем усредняющие фильтры наряду с разностными в основном относятся к линейным фильтрам, а упорядочивающие и адаптивные – к нелинейным.

1.2 Модели шума

Существует большое количество различных математических моделей случайного шума, основанных на статистических свойствах. Математическая простота, характерная для работы с моделями гауссова или нормального шума обусловила широкое распространение этих моделей на практике. Причем эта простота оказывается столь привлекательной, что зачастую гауссовы модели используются даже в тех ситуациях, когда их применение оправдано, в лучшем случае, лишь частично.

Функция плотности распределения вероятностей гауссовой случайной величины z задается выражением [3]
$$p(z) = \frac{1}{\sqrt{2\pi D}} e^{-\frac{(z-\mu)^2}{2D}}, \quad (1.2)$$

где z представляет собой значение случайной величины (в случае изображения – яркость), μ — среднее значение случайной величины, D — ее дисперсия. Если

плотность распределения случайной величины z описывается функцией (1.2), то приблизительно 70% ее значений попадают в диапазон $\{(\mu - \sigma), (\mu + \sigma)\}$, и примерно 95% — в диапазон $\{(\mu - 2\sigma), (\mu + 2\sigma)\}$, где σ – среднеквадратическое отклонение ($\sigma = \sqrt{D}$).

Функция плотности распределения вероятностей экспоненциального шума

$$\text{задается выражением [3] } p(z) = \begin{cases} ae^{-az}, & \text{при } z \geq 0 \\ 0, & \text{при } z < 0 \end{cases}, \quad (1.3)$$

где $a > 0$. Среднее значение и дисперсия для этого распределения имеют вид

$$\mu = \frac{1}{a}, \quad D = \frac{1}{a^2}. \quad (1.4)$$

Функция плотности распределения вероятностей равномерного шума зада-

$$\text{ется выражением [3] } p(z) = \begin{cases} \frac{1}{b-a}, & \text{при } a \leq z \leq b \\ 0, & \text{в остальных случаях} \end{cases} \quad (1.5)$$

Среднее значение и дисперсия для этого распределения имеют вид

$$\mu = \frac{a+b}{2}, \quad D = \frac{(b-a)^2}{12} \quad (1.6)$$

Данный шум, спектр которого является постоянной величиной, называется обычно «белым шумом», т.к. содержит все частоты появления случайной величины в равных пропорциях.

Функция плотности распределения вероятностей биполярного импульсного

$$\text{шума} \text{ задается выражением [3] } p(z) = \begin{cases} P_a, & \text{при } z = a \\ P_b, & \text{при } z = b \\ 0, & \text{в остальных случаях} \end{cases}, \quad (1.7)$$

где P_a, P_b – вероятности появления шума соответствующей яркости. Если $b > a$, то пиксель с яркостью b выглядит как светлая точка на изображении, а пиксель с яркостью a выглядит, наоборот, как темная точка. Если одно из значений вероятности равно нулю, то импульсный шум называется униполярным. Если ни одна из вероятностей не равна нулю и они приблизительно равны по величине, то импульсный шум называют шумом «соль и перец», т.к. визуально такой шум походит на крупинцы соли и перца, случайно рассыпанные по изображению.

Функции плотности распределения вероятностей (1.2) – (1.7) представляют собой набор средств, которые позволяют моделировать искажения, связанные с широким диапазоном встречающихся на практике шумов. Например, гауссов шум возникает на изображении в результате воздействия таких факторов, как шум в электронных цепях и шум сенсоров. Экспоненциальное распределение отвечает шуму на изображениях, получаемых с использованием лазеров. Импульсный шум возникает, если при получении изображения имеют место быстрые переходные процессы, например, неправильная коммутация. Равномерное распределение, пожалуй, в наименьшей степени подходит для описания встречающихся на практике явлений. Однако это распределение весьма полезно как основа для создания различных моделей зашумленных изображений с заданными параметрами [3].

Экспериментальные исследования показывают, что за исключением небольшого различия в общем уровне яркости, изображения, искаженные гауссовым, экспоненциальным и равномерным шумом визуально мало различимы, хотя их гистограммы существенно отличаются друг от друга [3]. Только внешний вид изображения, искаженного импульсным шумом, явно указывает на тип шума, который привел к данному искажению.

На рисунках 1.2 – 1.5 приведены исходное тестовое изображение и результаты его искажения равномерным и импульсным шумом, а также разрезы функции яркости по строке, более наглядно демонстрирующие искажения. Изображение на рисунке 1.2 выбрано в качестве тестового, т.к. на нем одновременно присутствуют плавные переходы яркости (открытые части тела), резкие переходы яркости (предметы мебели) и регулярные структуры (одежда).

1.3 Построение оценок для параметров шума

Параметры функции плотности распределения вероятностей шума могут быть частично известны из технических характеристик оптических сенсоров, однако часто необходимо оценить эти параметры в целом для конкретной системы, используемой при получении изображения. Если система находится в распоряжении пользователя, то один из простых способов изучения характеристик ее шума

заключается в том, чтобы получить набор изображений тестового объекта, например, большой, равномерно освещенной, сплошной серой поверхности. Результаты статистической обработки полученных изображений достаточно хорошо описывают шум системы.

В тех случаях, когда доступны только изображения, ранее сформированные системой, а сама система недоступна, рассмотрение небольших участков изображения примерно постоянной яркости дает возможность оценить параметры функции плотности распределения вероятностей шума. Приближение для среднего значения и дисперсии шума может быть получено на основе простых статистических формул $\mu = \sum_{z \in S} z_i p(z_i)$, $D = \sum_{z \in S} (z_i - \mu)^2 p(z_i)$ (1.8)

где z_i – значения яркости элементов участка изображения S , $p(z_i)$ – соответствующие нормализованные значения гистограммы яркости пикселей из S (см. п. 3.1).

Вид гистограммы определяет, какая из функций плотности распределения вероятностей является наиболее подходящей. Если форма гистограммы приблизительно гауссова, то для ее описания достаточно определить среднее и дисперсию согласно (1.8). Для распределений других типов, следует рассматривать выражения для среднего и дисперсии как уравнения для параметров a , b из (1.4) и (1.6), решив которые, можно найти параметры распределения для (1.3) и (1.5) [3].

Обработка импульсного шума осуществляется по-другому, т.к. в этом случае требуется оценить фактическую вероятность появления черных и белых точек на изображении. Для получения такой оценки необходимо, чтобы были видны как черные, так и белые точки. Таким образом, для вычисления гистограммы пригодна только такая область изображения, в которой значения яркости лежат в средней части диапазона и относительно постоянны. Высоты пиков, соответствующих черным и белым точкам, дают оценку вероятностей P_a и P_b из (1.7).

1.4 Подавление шума усредняющими фильтрами

Действие некоторых рассмотренных ниже усредняющих и разностных фильтров можно представить в виде свертки обрабатываемого изображения с

фильтром, ядро которого представлено виде матрицы или маски, коэффициенты которой имеют разное значение (вес), т.е. $z'(x, y) = \sum_{(s,t) \in S_{xy}} h(s, t)z(s, t)$, (1.9)

где $z(s, t)$, $z'(x, y)$ – яркости пикселей с координатами (s, t) и (x, y) соответственно искаженного и восстановленного изображений, $h(s, t)$ – весовые коэффициенты матрицы ядра фильтра, S_{xy} – область задания ядра фильтра или его апертура (центр апертуры S_{xy} совпадает с пикселем $z(x, y)$ искаженного изображения). В большинстве случаев апертура симметрична и имеет квадратную или круговую форму. На практике $h(s, t)$ имеют целочисленные значения, а результат свертки при необходимости умножается на нормирующий множитель, что позволяет сохранять без изменений области с постоянной яркостью.

Среднеарифметический фильтр является простейшим среди усредняющих фильтров [4]. Пусть S_{xy} обозначает прямоугольную окрестность, т.е. некоторую область изображения размером $m \times n$ с центром в пикселе (x, y) . Процедура фильтрации предполагает вычисление среднего арифметического значения яркости пикселей искаженного изображения $z(x, y)$ по окрестности S_{xy} . Значение яркости пикселя с координатами (x, y) восстановленного изображения представляет собой это значение, т.е.

$$z'(x, y) = \frac{1}{mn} \sum_{(s,t) \in S_{xy}} z(s, t). \quad (1.10)$$

Из сравнения (1.9) и (1.10) видно, что весовые коэффициенты матрицы ядра

среднеарифметического фильтра постоянны и равны $\frac{1}{mn}$, а его апертура и есть окрестность S_{xy} . Уменьшение шума происходит в результате сглаживания локальных вариаций яркости, но при значительной апертуре фильтра качество изображения ухудшается из-за заметного размывания перепадов яркости, как показано на рисунке 1.6. Если на изображении, обработанном среднеарифметическим фильтром с апертурой 3×3 (а), регулярные структуры (линии на брюках и платке) еще различимы, то после обработки фильтром с апертурой 9×9 (б) они незаметны, не смотря на значительные перепады яркости на них. Видны только линии на

платке, т.к. расстояние между линиями на нем превышает размер апертуры фильтра.

Результат обработки изображения среднегеометрическим фильтром, описывается выражением [3]

$$z'(x, y) = \left(\prod_{(s,t) \in S_{xy}} z(s, t) \right)^{\frac{1}{mn}}. \quad (1.11)$$

Здесь значение восстановленного изображения в каждом пикселе является корнем степени mn из произведения значений яркости в пикселях окрестности S_{xy} . Применение среднегеометрического фильтра приводит к сглаживанию, сравнимому с тем, которое достигается при использовании среднеарифметического фильтра, но при этом теряется меньше деталей изображения, как показано на рисунке 1.7.

Обработка сравнительно редко встречающимся в литературе среднегармоническим фильтром описывается выражением [3]

$$z'(x, y) = \frac{mn}{\sum_{(s,t) \in S_{xy}} \frac{1}{z(s, t)}}. \quad (1.12)$$

Среднегармонический фильтр хорошо работает в случае униполярного «белого» импульсного шума, т.е. когда значение шума соответствует появлению белых точек на изображении, но не работает в случае униполярного «черного» импульсного шума, а так же хорошо выполняет подавление гауссова шума, как показано на рисунке 1.8.

Обработка изображения контргармоническим фильтром описывается выражением [3]

$$z'(x, y) = \frac{\sum_{(s,t) \in S_{xy}} z(s, t)^{Q+1}}{\sum_{(s,t) \in S_{xy}} z(s, t)^Q}, \quad (1.13)$$

где Q называется порядком фильтра. Этот фильтр хорошо приспособлен для уменьшения или почти полного устранения импульсного шума, причем при положительных значениях Q фильтр устраняет «черную» часть импульсного шума, а при отрицательных значениях Q фильтр устраняет «белую» часть импульсного шума, но обе части шума не могут быть устранены одновременно. Наличие «бе-

лого» шума на изображении, обработанном контргармоническим фильтром, приводит к появлению светлых (Q положительное) или темных (Q отрицательное) пятен, размер которых зависит от значения Q , как показано на рисунке 1.9. Следует отметить, что контргармонический фильтр при $Q = 0$ превращается в среднеарифметический фильтр, а при $Q = -1$ – в среднегармонический фильтр.

Судя по изображениям, приведенным на рисунках 1.6 (b), 1.7 (b) и 1.8 (a), результаты обработки зашумленных «белым» шумом изображений среднеарифметическим, среднегеометрическим и среднегармоническим фильтрами визуаль-но практически не отличаются друг от друга.

Все рассмотренные выше усредняющие фильтры имеют одно общее свойство – влияние на результат, оказываемое каждым попавшим в апертуру пикселем, не зависит от его расположения в апертуре. В результате реакция сглаживающего фильтра на импульсную функцию представляет собой совпадающую по форме с апертурой область с постоянной яркостью. Желательно же иметь результат близкий к результату работы расфокусированной видеокамеры – маленькая яркая точка на темном фоне должна превратиться в симметричную круглую размытую область, которая ярче всего в центре, а при приближении к краю постепенно теряет яркость. Очевидно, что в дающем похожую реакцию усредняющем фильтре маска должна иметь разные весовые коэффициенты, показывающие степень влияния соответствующего пикселя на результат свертки, а сумма этих коэффициентов должна быть равна единице.

Примером такого фильтра является гауссиан [4], у которого в центре весовые коэффициенты пикселей существенно больше, чем на его границах. Коэффициенты матрицы дискретного ядра или маски гауссиана определяются по формуле

$$h_{ij} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{((i-k-1)^2 + (j-k-1)^2)}{2\sigma^2}\right) \quad (1.14)$$

где h_{ij} – значение элемента матрицы с координатами i, j (верхний левый элемент матрицы – h_{11}), σ – среднеквадратичное отклонение гауссиана (в пикселях), k – радиус апертуры, т.е. размер квадратной апертуры соответственно $2k+1 \times 2k+1$.

При вычислении весовых коэффициентов матрицы ядра гауссиана по (1.14) необходимо правильно подобрать соотношение между σ и k , а также произвести масштабирование полученных коэффициентов [5]. Если σ меньше чем один пиксель, то эффект сглаживания будет незначительным, т.к. весовые коэффициенты всех пикселей кроме центральных будут очень малыми. Если же σ велико, то k также должно быть большим, иначе не будет уменьшения размывания на краю апертуры. Масштабирование ядра выполняется так, что бы все коэффициенты можно было округлить до целых чисел без существенных потерь в точности. Нормирующий множитель свертки равен обратному значению к сумме коэффициентов, что позволяет сохранить без изменений области с постоянной яркостью. Ниже приведены примеры масок гауссианов размером 3×3 ($\sigma=1$) и нижние правые четверти масок 7×7 ($\sigma=2$) и 11×11 ($\sigma=2$):

$$H_G = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}; \begin{bmatrix} 90 & 70 & 33 & 9 \\ 70 & 55 & 26 & 7 \\ 33 & 26 & 12 & 3 \\ 9 & 7 & 3 & 1 \end{bmatrix}; \begin{bmatrix} 119 & 105 & 72 & 39 & 16 & 7 \\ 105 & 93 & 64 & 34 & 14 & 5 \\ 72 & 64 & 44 & 24 & 10 & 3 \\ 39 & 34 & 24 & 13 & 5 & 2 \\ 16 & 14 & 10 & 5 & 2 & 1 \\ 7 & 5 & 3 & 2 & 1 & 0 \end{bmatrix} \quad (1.15)$$

На рисунке 1.10 показаны результаты обработки зашумленных изображений гауссианом с маской 9×9 .

Преимущество применения гауссиана для сглаживания яркостных помех в первую очередь состоит в том, что результат двух последовательных сверток изображения гауссианами с σ_1 и σ_2 соответственно эквивалентен свертке исходного изображения гауссианом с $\sigma = \sqrt{\sigma_1^2 + \sigma_2^2}$ [5]. Таким образом, вместо достаточно затратной по времени свертки с $\sigma=10$ пикселей, для которой может понадобиться матрица весовых коэффициентов размером 51×51 , можно применить последовательную свертку с ядром гораздо меньших размеров.

Из литературы известно [5], что многократная свертка с самим собой обычного усредняющего фильтра с постоянной весовой матрицей дает результат близ-

кий к гауссиану. Учитывая свойство ассоциативности свертки, это означает, что если многократно применять к изображению сглаживающий усредняющий фильтр, то результат будет практически совпадать с результатом сглаживания этого изображения гауссианом.

Важным преимуществом гауссиана, позволяющим значительно сократить время обработки изображения, является возможность представить его как произведение двух одномерных гауссианов [5], т.е. гауссиан имеет разделяемое ядро, что позволяет заменить свертку изображения двумерным гауссианом на его последовательную свертку одномерными гауссианами по строкам и столбцам.

1.5 Подавление шума упорядочивающими фильтрами

Упорядочивающие фильтры, основанные на порядковых статистиках, представляют собой пространственные фильтры, вычисление отклика которых требует предварительного ранжирования, т.е. упорядочивания значений пикселей, заключенных внутри обрабатываемой фильтром области S_{xy} изображения. Отклик фильтра определяется по результатам упорядочивания и не может быть представлен в виде свертки (1.9).

Наиболее известным из фильтров, основанных на порядковых статистиках, является медианный фильтр. Действие этого фильтра, как следует из его названия, состоит в присвоении пикселю восстановленного изображения с координатами (x,y) значения медианы упорядоченного множества значений яркости из окрестности S_{xy} обрабатываемого изображения [4], т.е. $z'(x,y) = \text{med}_{(s,t) \in S_{xy}} \{z(s,t)\}$. (1.16)

Широкая популярность медианных фильтров обусловлена тем, что они прекрасно приспособлены для подавления некоторых видов случайных шумов, и при этом приводят к меньшему размыванию контуров по сравнению с линейными сглаживающими фильтрами того же размера. Медианные фильтры эффективны при наличии как биполярного, так и униполярного импульсного шума, но мало эффективны при устранении гауссова шума, как показано на рисунке 1.11. Существенным недостатком медианных фильтров является подавление и искажение мелких объектов, например, регулярных структур на рисунке 1.11.

Хотя медианные фильтры принадлежат к числу наиболее часто используемых в обработке изображений фильтров, основанных на порядковых статистиках, это отнюдь не единственный пример таких фильтров. Медиана (1.16) представляет собой середину упорядоченного набора чисел, однако использование иных характеристик этого набора предоставляет много других возможностей. Например, использование крайних значений ранжированного набора чисел приводит к фильтрам максимума и минимума, основанным на выборе максимального или минимального значения яркости из окрестности S_{xy} , соответственно [3].

Фильтр максимума полезен при обнаружении наиболее ярких точек на изображении. Кроме того, поскольку униполярный «черный» импульсный шум принимает минимальные значения, применение этого фильтра приводит к уменьшению такого шума, так как в процессе фильтрации из окрестности S_{xy} выбирается максимальное значение.

Соответственно, фильтр минимума полезен при обнаружении наиболее темных точек на изображении и его применение приводит к уменьшению униполярного «белого» импульсного шума.

Последовательное применение максимального и минимального фильтров с одинаковым размером апертуры позволяет эффективно удалять дефекты (пятна, разрывы линии контура) на бинарных изображениях без изменения формы и размеров фрагментов, несущих полезную информацию, как показано на рисунке 1.12. Максимальный размер удаляемых дефектов равен размеру окрестности S_{xy} .

Фильтр срединной точки или Чебышевский заключается в вычислении среднего между максимальным и минимальным значениями в окрестности S_{xy} [3]:

$$z'(x, y) = \frac{1}{2} \left(\max_{(s,t) \in S_{xy}} \{z(s, t)\} + \min_{(s,t) \in S_{xy}} \{z(s, t)\} \right) \quad (1.17)$$

Следует отметить, что этот фильтр, объединяя в себе методы порядковых статистик и усреднения, лучше всего работает при наличии случайно распределенных шумов со значительной дисперсией, как показано на рисунке 1.13.

Фильтр усеченного среднего усредняет значения яркости после удаления $d/2$ наименьших и $d/2$ наибольших значений из множества всех значений функции

$$z(s,t) \text{ в окрестности } S_{xy} \text{ [3], т.е. } z'(x,y) = \frac{1}{mn-d} \sum_{(s,t) \in S_{xy}} z_r(s,t), \quad (1.18)$$

где $z_r(s,t)$ – оставшиеся после удаления значения яркости, число которых равно $mn - d$. Значение d может изменяться в диапазоне от 0 до $mn - 1$. Если $d = 0$, то фильтр усеченного среднего сводится к среднеарифметическому фильтру. В случае $d = (mn - 1)$, фильтр превращается в медианный фильтр. Использование фильтра усеченного среднего с другими значениями d полезно в тех случаях, когда на изображении одновременно имеются несколько видов шума, например, комбинация импульсного и гауссова шума, как показано на рисунке 1.14.

Как правило, форма окрестности S_{xy} выбирается квадратной с нечетным числом пикселей, но возможно использование и других форм, например, прямоугольной, крестообразной, кругообразной и т.п. Выбор формы определяется требуемыми свойствами фильтра, например, крестообразная окрестность для медианного фильтра позволяет сохранять без изменения яркость угловых пикселей прямоугольных фрагментов [6].

Еще одним интересным примером нелинейного фильтра, сочетающего методы усреднения и упорядочивания яркости пикселей окрестности S_{xy} , является фильтр Кувахары [7]. Последовательность действий этого фильтра для получения яркости обработанного пикселя $z'(x,y)$ по яркости пикселей из окрестности пикселя исходного изображения $z(x,y)$ следующая. Квадратная окрестность пикселя $z(x,y)$ размером $n \times n$ (n – нечетное число, $n > 3$) разделяется на четыре непересекающиеся квадратные области размером $0,5n - 1$, симметричные относительно центра окрестности. Для каждой области вычисляется средняя яркость и дисперсия, а $z'(x,y)$ присваивается значение средней яркости той области, дисперсия в которой минимальна по сравнению с дисперсиями остальных трех областей.

При правильном подборе размера окрестности S_{xy} фильтр Кувахары позволяет подавлять гауссов и импульсный шум с меньшим размытием границ фрагментов, чем усредняющие фильтры, а также удалять мелкие детали на изображе-

нии, как показано на рисунке 1.15. Основная область применения этого фильтра – подготовка аэрофотоснимков для автоматической сегментации и распознавания.

Существенный интерес представляет сравнение результатов воздействия различных фильтров на одно и то же тестовое изображение. В таблицах 1 – 10 приведены значения одномерных функций яркости, соответствующих типичным фрагментам изображения (импульсное, резкое, плавное и периодическое изменение яркости), а также результаты обработки этих изображений усредняющими (таблицы 1 – 5) и упорядочивающими (таблицы 6 – 10) одномерными фильтрами. На рисунках 1.16 – 1.25 приведены графики функций яркости исходных и обработанных фрагментов изображений в соответствии с таблицами 1 – 10.

1.6 Подавление шума адаптивными фильтрами

Рассмотренные до сих пор фильтры применяются к изображению без учета того, как свойства изображения меняются от пикселя к пикселю. Результат работы адаптивных фильтров зависит от статистических свойств изображения внутри области действия фильтра, которая определяется прямоугольной $m \times n$ окрестностью S_{xy} с центром в пикселе с координатами (x,y) . Из литературы [3] известно, что возможности адаптивных фильтров превосходят возможности фильтров, рассмотренных ранее. Платой за усовершенствование методов фильтрации является увеличение сложности фильтров, времени обработки изображения и числа задаваемых параметров.

Простейшими характеристиками случайной величины являются ее среднее значение и дисперсия. Среднее значение определяет меру средней яркости той области, по которой оно вычисляется, а дисперсия позволяет оценить меру отклонения яркости в этой области. Эти параметры и берутся за основу при создании адаптивного усредняющего фильтра, поскольку их величины тесно связаны с внешним видом изображения.

Отклик адаптивного усредняющего фильтра обусловлен:

– значением яркости изображения с шумом в пикселе (x,y) ;

- дисперсией D_η шума, превращающего яркость пикселя исходного изображения $f(x,y)$ в яркость пикселя искаженного изображения $z(x,y)$;
- локальным средним μ_z по значениям в окрестности S_{xy} ;
- локальной дисперсией D_z по значениям в окрестности S_{xy} .

Результат работы фильтра определяется следующими условиями:

- если дисперсия шума D_η равна нулю, то отклик фильтра должен быть равен значению $z(x,y)$, что соответствует отсутствию шума;
- если $D_z \gg D_\eta$, то значение отклика фильтра должно быть примерно равно $z(x,y)$, т.к. большое значение локальной дисперсии обычно связано с наличием контуров, которые должны быть сохранены;
- если обе дисперсии принимают значения одного порядка, то отклик фильтра должен быть равен μ_z , что соответствует отклику обычного среднеарифметического фильтра, т.к. можно предположить, что в данном случае отклонения яркости в окрестности S_{xy} обусловлены только шумом.

Отклик фильтра, удовлетворяющего перечисленным условиям, задает вы-

$$\text{ражение [3]: } z'(x,y) = z(x,y) - \frac{D_\eta}{D_z}(z(x,y) - \mu_z). \quad (1.19)$$

Единственной величиной, которая должна быть заранее известна или оценена по методике, изложенной выше в разделе 1.3, является полная дисперсия шума D_η , причем в качестве ее грубой оценки можно использовать дисперсию яркости всего изображения. Остальные входящие в формулу (1.19) величины вычисляются для каждого пикселя по значениям элементов изображения в окрестности S_{xy} с центром в этом пикселе. В формуле (1.19) неявно предполагается, что $D_\eta < D_z$. Поскольку рассматриваются модели только аддитивного шума, такое предположение является оправданным, но при реализации формулы (1.19) следует предусмотреть дополнительную проверку условия $D_\eta < D_z$ и при его нарушении использовать в формуле (1.19) значение 1 вместо величины соответствующего отношения. Это позволяет предотвратить появление бессмысленных результатов, т.е. отрицательных значений яркости при некоторых значениях μ_z . В результате метод

адаптивной фильтрации становится нелинейным. Другой подход может состоять в том, чтобы допустить появление отрицательных значений, но затем изменить шкалу яркости, что приведет к уменьшению диапазона яркости всего изображения. Результаты обработки изображения с «белым» шумом среднеарифметическим и адаптивным усредняющим фильтрами приведены на рисунке 1.26.

Медианные фильтры хорошо работают до тех пор, пока пространственная плотность импульсного шума невелика (P_a и P_b не превышают 0,2). Адаптивный медианный фильтр [3] позволяет справиться с импульсным шумом, вероятность которого превышает указанные значения, и в большей степени, чем обычный медианный фильтр, сохранить мелкие детали в областях, не искаженных импульсным шумом.

Подобно всем рассмотренным до сих пор фильтрам, адаптивный медианный фильтр осуществляет обработку в прямоугольной окрестности S_{xy} . Однако, в отличие от этих фильтров, адаптивный медианный фильтр при необходимости увеличивает размеры окрестности S_{xy} .

Отклик фильтра обусловлен следующими величинами:

- минимальным z_{min} и максимальным z_{max} значениями яркости в S_{xy} ;
- медианой z_{med} значений яркости в S_{xy} ;
- значением яркости z_{xy} в пикселе с координатами (x,y) ;
- максимально допустимым S_{max} размером окрестности S_{xy} .

Адаптивный медианный фильтр работает следующим образом. В начале для окрестности S_{xy} , обрабатываемого пикселя с координатами (x,y) , вычисляются z_{min} , z_{med} , z_{max} и проверяется выполнение условия $z_{min} < z_{med} < z_{max}$. (1.20)

Если условие (1.20) выполняется, то проверяется выполнение еще одного условия $z_{min} < z_{xy} < z_{max}$. Если и оно выполняется, то отклик фильтра равен z_{xy} , если нет, то отклик фильтра равен z_{med} .

Если условие (1.20) не выполняется, то размер окрестности S_{xy} увеличивает, вычисления повторяются и условие (1.20) проверяется для новых значений z_{min} , z_{med} , z_{max} . При достижении размера окрестности S_{xy} значения S_{max} отклик фильтра принимается равным z_{xy} .

Фильтр преследует три основные цели: удалить биполярный импульсный шум, обеспечить сглаживание шумов других типов и свести к минимуму такие искажения, как чрезмерное утончение или утолщение мелких деталей, как показано на рисунке 1.27.

Значения z_{min} и z_{max} воспринимаются фильтром статистически как значения импульсных составляющих шума, даже если они не равны наименьшему и наибольшему возможным значениям яркости на изображении, т.е. проверкой условия (1.20) фильтр пытается определить, является ли медиана z_{med} импульсом. Если условие (1.20) выполнено, то z_{med} не является импульсом и далее проверяется, является ли импульсом значение z_{xy} в центре окрестности. Если z_{xy} не является импульсом, то отклик фильтра равен z_{xy} . Сохранение значений яркости в таких пикселях минимизирует искажения, вносимые обработкой изображения. Если $z_{xy} = z_{min}$ либо $z_{xy} = z_{max}$, то отклик фильтра соответствует отклику обычного медианного фильтра и равен z_{med} , которое не является значением импульсного шума, как следует из выполнения условия (1.20).

Если условие (1.20) не выполняется, т.е. в окрестности S_{xy} текущего размера отсутствуют пиксели хотя бы трех разных яркостей, размер окрестности увеличивается и процесс повторяется до тех пор, пока либо не будет найдена медиана, отличная от импульса, либо размеры окрестности не превысят максимально допустимый размер. В последнем случае отклик фильтра равен z_{xy} , причем нет гарантий, что это значение не является импульсным. Чем больше вероятности шума P_a и/или P_b , или чем больше максимальный допустимый размер окрестности S_{max} , тем меньше вероятность такого отклика.

После получения значения обрабатываемого элемента изображения, центр окрестности смещается в позицию следующего элемента, и фильтр начинает обработку находящейся в новом положении окрестности S_{xy} исходного размера.

Следует отметить, что кроме подавляющих помехи фильтров, рассмотренных выше, существует и большое количество других фильтров, выполняющих аналогичные функции [8], но реже применяемых для обработки изображений. Например, взвешенные, разностные, комбинированные фильтры и ряд других.

2. Выделение контурных линий

Из литературы [3] известно, что информация, позволяющая отличать объекты друг от друга по их изображениям, в значительной степени содержится в контурных линиях. Основным принципом большинства методов выделения контурных линий, отделяющих соседние фрагменты, является вычисление частных производных от функции яркости по координатам. Предполагается, что яркости фрагментов постоянны и существенно отличаются друг от друга. Знак производной функции яркости зависит от направления перехода яркости, т.е. производные на участках повышения яркости между фрагментами положительны, а на участках понижения яркости – отрицательны. Первую производную функции яркости можно использовать для обнаружения наличия контурной линии, а вторую производную – для определения ширины этой линии.

Так как яркость изображения является функцией двух переменных, градиент функции яркости в каждой точке определяется как двумерный вектор

$$\mathbf{G}[z(x,y)] = \begin{bmatrix} G_x \\ G_y \end{bmatrix}, \text{ где } G_x = \frac{\partial z(x,y)}{\partial x}, G_y = \frac{\partial z(x,y)}{\partial y} - \text{частные производные.}$$

Известно [2], что вектор \mathbf{G} указывает направление максимального изменения функции $z(x,y)$ в точке (x,y) , которое можно определить как угол между осью X и направлением вектора \mathbf{G} , причем $\text{tg } \alpha = \frac{G_y}{G_x}$. При выделении контурной линии используется длина этого вектора $z'(x,y) = |\mathbf{G}| = \sqrt{G_x^2 + G_y^2}$ или для упрощения вычислений – $|\mathbf{G}| = |G_x| + |G_y|$, где $|\cdot|$ – взятие абсолютного значения, т.к. яркость результирующего изображения в любой точке не может иметь отрицательных значений.

На рисунке 2.1 приведен разрез функции яркости изображения объект-фон, идеального с точки зрения отсутствия шума, и соответствующие графики первой и второй производных функции яркости с учетом знака.

Для дискретных изображений вычисление частных производных сводится к вычислению разности яркостей соседних пикселей различными способами, т.е.

фактически к пространственной фильтрации путем свертки с различными по размеру и значению коэффициентов масками разностных фильтров. Тогда,

$$G_x = \sum_{(s,t) \in S_{xy}} h_x(s,t)z(s,t), G_y = \sum_{(s,t) \in S_{xy}} h_y(s,t)z(s,t), \quad (2.1)$$

где h_x, h_y – коэффициенты соответствующих масок H_x, H_y ; S_{xy} – окрестность обрабатываемого пикселя.

Далее в виде квадратных матриц приведены коэффициенты масок наиболее известных фильтров.

Маски фильтра Робертса [4], использующего для определения градиента яркости минимальное число пикселей, имеют вид $H_x = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, H_y = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$, где текущий пиксель (x,y) соответствует верхнему левому элементу маски.

Маски фильтра Собеля [4] для квадратной окрестности из 9-ти пикселей

$$\text{имеют вид, } H_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}, H_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}. \quad (2.2)$$

Маски фильтра Превитт [3] получаются из масок фильтра Собеля путем замены значения коэффициентов 2 на 1 с сохранением знака, что позволяет несколько упростить алгоритм обработки и повысить помехоустойчивость, т.к. маски фильтра Превитт являются результатом вычисления первой производной от функции второго порядка, аппроксимирующей значения яркости в окрестности обрабатываемого пикселя [9]. Результаты обработки фильтрами Робертса, Собеля и Превитт качественного и зашумленного изображений приведены на рисунках 2.2, 2.3 и 2.4.

Для вычисления второй производной функции яркости применяется оператор Лапласа, маска которого для дискретного изображения может иметь вид [6]:

$$H_L = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{pmatrix} \text{ или } \begin{pmatrix} 2 & -1 & 2 \\ -1 & -4 & -1 \\ 2 & -1 & 2 \end{pmatrix}. \quad (2.3)$$

Данные маски обладают круговой симметрией, что позволяет не вычислять частные производные по отдельности, т.е. $z'(x, y) = \left| \sum_{(s,t) \in S_{xy}} h_L(s, t) z(s, t) \right|$ (2.4).

Необходимо отметить, что фильтр, реализующий оператор Лапласа, чувствителен к малым перепадам яркости, как показано на рисунке 2.6. Его целесообразно применять к изображениям с максимально подавленным шумом. Однако, последняя маска – маска согласованного лапласиана, обеспечивает некоторую помехоустойчивость, т.к. является результатом вычисления лапласиана от функции второго порядка, аппроксимирующей значения яркости в окрестности обрабатываемого пикселя [9], что видно при сравнении рисунков 2.6 (а) и (б).

Анализ применения фильтров, выделяющих контурные линии, показывает, что импульсная помеха этими фильтрами размывается, область перепада яркости выделяется фильтрами, берущими первую производную функции яркости, как широкая линия, а фильтром Лапласа – как две параллельные линии, что хорошо видно на рисунках 2.2 и 2.5.

Иногда полезны фильтры, выделяющие контурные линии только в заданном направлении, в результате работы которых яркость контурной линии на обработанном изображении зависит от перепада яркости и направления линии. Например, маска фильтра, выделяющего линии перепада яркости, расположенные под углом в 45° , может иметь вид [12]:

$$H_{45} = \begin{pmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{pmatrix} \text{ или } \begin{pmatrix} 1 & -1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & 1 \end{pmatrix}. \quad (2.5)$$

Интересным вариантом нелинейного пространственного фильтра для выделения контура является фильтр Кирша [4]: $z'(x, y) = \max(r_1, \dots, r_8)$, где r_n – абсолютные значения перепадов яркости по восьми направлениям в окрестности обрабатываемого пикселя. Значения r_i определяются путем линейной пространственной фильтрации. Матрица фильтра для направления «с севера на юг» (NS) имеет вид:

$$H_{NS} = \begin{pmatrix} 5 & 5 & 5 \\ -3 & 0 & -3 \\ -3 & -3 & -3 \end{pmatrix}. \quad (2.6)$$

Для остальных семи направлений матрицы фильтров получаются вращением данной матрицы вокруг центра. Фильтр Кирша позволяет выделять контуры, со слабым перепадом яркости, значительно увеличивая как уровень яркости пикселей, относящихся к контурным линиям, так и шум, что видно на рисунке 2.7. На рисунке 2.7 (а) хорошо видна клетчатая структура, которую можно заметить как на исходном изображении (рисунок 1.2), так и на изображениях, обработанных другими фильтрами, выделяющими перепады яркости (рисунки 2.2 – 2.6). Появление такой структуры на исходном изображении объясняется особенностями сжатия изображений методом JPEG [13].

В отличие от усредняющих фильтров, сумма коэффициентов масок которых равна единице, сумма коэффициентов фильтров (2.2), (2.3), (2.5) и (2.6), выделяющих перепады яркости, равна нулю, что позволяет областям с постоянной яркостью на исходном изображении иметь нулевое значение яркости пикселей на обработанном изображении. Из анализа масок разностных фильтров видно, что все они, за исключением фильтра Робертса, усиливают перепад яркости, причем коэффициент усиления можно оценить как сумму коэффициентов маски одного знака. Обычно в качестве значения яркости пикселя изображения, полученного сверткой исходного изображения и фильтра с какой-либо маской (2.2), (2.3), (2.5) или (2.6), берется результат свертки по абсолютной величине, что позволяет усилить незначительные перепады яркости, как показано на рисунке 2.2. Если желательно получить истинные значения перепадов яркости, то результат свертки уменьшается на коэффициент усиления фильтра. В последнем случае, как видно из рисунка 2.3, результаты обработки фильтрами Собеля и Превитт мало отличаются друг от друга.

Результат обработки изображения фильтром Уоллеса, также применяемого для выделения перепадов яркости, нельзя представить в виде свертки изображения с некоторой маской, так как он (результат) определяется формулой [4]:

$$z'(x, y) = C_{norm} \left| \ln \left(\frac{z(x, y)}{z(x-1, y)} * \frac{z(x, y)}{z(x, y-1)} * \frac{z(x, y)}{z(x+1, y)} * \frac{z(x, y)}{z(x, y+1)} \right) \right|, \quad (2.7)$$

где C_{norm} – нормирующий коэффициент, $| \cdot |$ – взятие абсолютного значения.

Так как стоящая в знаменателе (2.7) яркость у некоторых пикселей может быть равна нулю, то следует перед обработкой увеличить яркости всех пикселей на единицу.

Из (2.7) видно, что значение логарифма может меняться в пределах от нуля при равенстве всех яркостей до $\pm 22,2$ при максимально возможной разности яркостей центрального пикселя и четырех его соседей. Коэффициент C_{norm} желательно подобрать так, чтобы получаемые по формуле (2.7) значения яркости всех пикселей обработанного изображения занимали весь допустимый диапазон яркости (0...255). Такого результата можно добиться или подбирая значения C_{norm} экспериментально до обработки, или произведя обработку изображения в два прохода. На первом проходе при $C_{norm} = 1$ вычисляются по (2.7) значения $z'(x, y)$ и запоминаются как действительные числа, после чего определяются максимальное и минимальное значения $z'(x, y)$ на всем изображении, по которым и вычисляется

$C_{norm} = \frac{255}{z'_{max} - z'_{min}}$. На втором проходе яркость пикселя (x, y) обработанного изображения определяется как $z''(x, y) = [C_{norm}(z'(x, y) - z'_{min})]$, где $[..]$ – результат округления до целого числа.

За счет нелинейности логарифмической функции фильтр Уоллеса в большей степени усиливает незначительные перепады яркости. Результаты обработки фильтром Уоллеса качественного и зашумленного изображений приведены на рисунке 2.8.

Фильтры, выделяющие контурные линии, относятся к высокочастотным фильтрам, т.е. фильтрам, подавляющим низкочастотную составляющую изображения. Теоретически на результирующем изображении яркость должна быть отлична от нуля только у пикселей, входящих в контурные линии. На практике из-за наличия на исходном изображении помех подавляющее большинство пикселей результирующего изображения после обработки высокочастотным фильтром,

имеют отличную от нуля яркость, причем высокочастотный фильтр усиливает шум, как видно на рисунках 2.4, 2.6, 2.7 (b) и 2.8 (b). Для выделения контурных линий применяется пороговое отсечение с экспериментально подбираемым уровнем порога.

Для ослабления влияния шума на результат выделения контурных линий оператором Лапласа рекомендуется произвести сглаживание шума гауссианом перед обработкой изображения лапласианом, который усиливает шум значительно больше, чем фильтры, определяющие первую производную. Последовательную обработку изображения этими двумя фильтрами можно заменить обработкой комбинированным фильтром, который представляет собой результат применения оператора Лапласа к гауссиану и называется LoG-фильтром [2]. Маски таких фильтров симметричны относительно центра и результат обработки изображения LoG-фильтром определяется по формуле (2.4) аналогично оператору Лапласа.

В зависимости от характеристик гауссиана маски LoG-фильтров могут иметь разные размеры. Маска LoG-фильтра размером 3×3 ($\sigma = 1$) и нижние правые четверти масок размером 7×7 ($\sigma = 1$) и 11×11 ($\sigma = 2$) приведены ниже:

$$H_{LoG} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}; \begin{bmatrix} 104 & 0 & -6 & -2 \\ 0 & -8 & -3 & -1 \\ -6 & -3 & -2 & 0 \\ -2 & -1 & 0 & 0 \end{bmatrix}; \begin{bmatrix} 180 & 103 & -1 & -23 & -9 & -2 \\ 103 & 52 & -14 & -22 & -8 & -1 \\ -1 & -14 & -24 & -15 & -4 & -1 \\ -23 & -22 & -15 & -7 & -2 & 0 \\ -9 & -8 & -4 & -2 & 0 & 0 \\ -2 & -1 & -1 & 0 & 0 & 0 \end{bmatrix}. \quad (2.8)$$

LoG-фильтры с масками минимально возможного размера выделяют границы между малыми областями, но излишне чувствительны к фрагментам границ с большой кривизной и дают значительные отклики на шум, т.е. их результат практически совпадает с результатом обработки изображения оператором Лапласа. Маски большого размера дают заметный отклик только на гладкие границы между большими областями за счет существенного сглаживания гауссианом не только шума, но и мелких деталей на изображении, как можно увидеть на рисунке 2.9. Эксперименты показывают, что зрительная система человека использует аппарат,

результаты работы которого похожи на результаты последовательного применения нескольких LoG-фильтров разного размера [2].

Коэффициент усиления высокочастотной составляющей изображения при его обработке LoG-фильтром с одной из масок, приведенных в (2.8), определяется суммой коэффициентов h_{LoG} одного знака и может быть достаточно велик в масках большого размера. Как следствие, полученные в результате обработки контрастных изображений значения, могут значительно превышать максимально допустимую яркость. Простое отсечение яркости больше максимально допустимой приводит к излишнему подчеркиванию шума, а изображение становится практически двух градационным. Во избежание данного явления следует или перейти к маске с действительными значениями коэффициентов, или при обработке вводить нормирующий коэффициент, подобранный аналогично C_{norm} в фильтре Уоллеса так, чтобы значения яркости всех пикселей обработанного изображения не превышали максимально допустимого значения.

Из рисунка 2.1 видно, что первая производная, т.е. градиент функции яркости, имеет ширину несколько пикселей, а вторая производная дает две узкие параллельные линии. Для получения контурной линии шириной в один пиксель наиболее часто применяются два метода.

Первый метод [2] заключается в отнесении к контурной линии пикселей, в которых переходит через ноль вторая производная яркости, полученная обработкой оператором Лапласа с одной из масок (2.3) или LoG-фильтром с какой-либо из масок (2.8). Соответственно, вычислять отклик фильтра по (2.4) следует с учетом знака. Основным недостатком данного метода – возможное появление двойных линий контура при обработке размытых изображений, т.к. вторая производная выделяет границы зоны изменения яркости.

Второй метод, называемый методом немаксимального подавления [5], заключается в отнесении к контурной линии пикселей, в которых величина градиента яркости максимальна вдоль направления градиента. На первом этапе вычисляются значение и направление градиента яркости для всех пикселей изображения с помощью фильтра Робертса, Собеля (2.2) или Превитт. Далее производится

поиск цепочек пикселей, соответствующих контурным линиям. В каждую цепочку входят соседние пиксели с максимальным локальным градиентом, которые находятся интерполяцией градиентов ближайших точек как показано на рисунке 2.10.

Рассмотрим второй этап более подробно. Пусть известны величина и направление градиента яркости в пикселе A , т.е. в точке изображения, соответствующей центру пикселя. Линейной интерполяцией по значениям градиента яркости в ближайших пикселях найдем величину градиента в точках B и C , лежащих на линии направления градиента и являющихся кандидатами на локальный максимум градиента. Если градиент в пикселе A больше, чем в точках B и C , то пиксель A – локальный максимум градиента. В противном случае, кандидатом на локальный максимум следует выбрать пиксель, ближайший к точке с максимальным градиентом, и продолжить поиск максимума градиента.

Если локальный максимум градиента яркости найден, например в пикселе A , то этот пиксель включается в контурную линию, а направлением поиска следующего пикселя данного контура является перпендикуляр к линии градиента, причем следует снова провести интерполяцию, чтобы найти значение градиента в точке D между пикселями. В качестве следующего пикселя – кандидата на локальный максимум яркости выбирается ближайший к точке D пиксель.

В результате получается бинарное изображение, т.к. по окончании обработки каждый пиксель будет или отнесен к какой-либо контурной линии, или нет. Для уменьшения числа ложных контуров желательно перед вычислением градиентов яркости провести подавление помех, а при поиске контурных линий исключать из рассмотрения пиксели с незначительным уровнем градиента. На рисунке 2.11 показаны результаты обработки изображения, приведенного на рисунке 1.2, без предварительного размывания (а) и с размыванием фильтром Гаусса и установкой порога отсечения градиента яркости (б).

3. Повышение различимости фрагментов

3.1 Характеристики качества изображения

Одной из главных характеристик качества изображения, позволяющих дать его статистическую оценку, является гистограмма яркости, представляющая собой графическое отображение функции $P(z)$, где значение $P(z_i)$ – число пикселей яркости z_i на всем изображении [5]. Нормализованные гистограммы $p(z)$, где

$$p(z_i) = \frac{P(z_i)}{N} \quad (N - \text{число пикселей изображения}),$$

позволяют сравнивать гистограммы изображений разных размеров. Причем значение $p(z_i)$ является оценкой вероятности того, что яркость случайно выбранного на изображении пикселя равна z_i .

Форма гистограммы и ее диапазон часто позволяют дать качественную оценку изображения и возможность отнести его к одному из известных классов:

- бимодальная гистограмма характерна для изображений с хорошо различимыми контрастными фрагментами, например, печатные буквы на странице текста;
- одномодальная гистограмма часто соответствует изображениям с размытыми фрагментами, например, крупные объекты на фоне с мелкими деталями;
- равномерная гистограмма может наблюдаться у изображений с большим количеством мелких фрагментов различной яркости.

При оценке качества изображения используются такие понятия как, контрастность и резкость. Контрастность всего изображения или глобальная контрастность [6] $C = z_{\max} - z_{\min}$, (3.1)

где z_{\max} , z_{\min} – соответственно максимальная и минимальная яркость пикселей на изображении. Под локальной контрастностью понимается перепад яркости между соседними фрагментами (объект-фон), т.е. $C_l = |z_{\max}^{(l)} - z_{\min}^{(l)}|$, где $z_{\max}^{(l)}$, $z_{\min}^{(l)}$ – максимальная и минимальная яркость пикселей, совпадающих с границей зоны перехода (I) между фрагментами, как показано на рисунке 3.1. Относительная контрастность

$C_{om} = \frac{C}{z_m}$, где z_m – максимально возможная яркость (в большинстве случаев

$z_m=255$), позволяет определить возможность и целесообразность линейного увеличения контрастности.

$$\text{Резкость } R \text{ определяется как отношение [6] } R = \frac{C_l}{d}, \quad (3.2)$$

где d – ширина зоны перехода (l) между фрагментами, как показано на рисунке 3.1. Можно сказать, что резкость определяется производной функции яркости, т.е. чем более размытой является граница перехода между фрагментами, тем меньше резкость. Из (3.2) видно, что в отличие от контрастности резкость изображения может быть только локальной. Не только зона перехода между фрагментами одного изображения, как правило, имеет разную ширину, но и разные соседние фрагменты могут иметь разную яркость. Возможное в принципе вычисление максимальной и средней резкости всего изображения, как правило, дает мало полезной информации для оценки качества изображения.

Из (3.1) и (3.2) видно, что увеличение локальной контрастности путем изменения яркости всех пикселей соседних фрагментов приводит и к увеличению резкости. Увеличение только резкости не изменяет яркость пикселей фрагментов, за исключением тех из них, которые находятся на границе фрагментов.

Визуальная оценка качества изображения с точки зрения контрастности и резкости, как правило, ограничивается такими качественными характеристиками как высокое, среднее и низкое. Анализируя результаты работы различных алгоритмов улучшения качества изображения и повышения различимости его фрагментов, следует использовать изображения с различными комбинациями этих характеристик: изображения с высокими резкостью и контрастностью, с низкими резкостью и контрастностью, с высокой резкостью и низкой контрастностью, с низкой резкостью и высокой контрастностью. Кроме этого, для каждой комбинации желательно рассмотреть изображения с мелкими и крупными фрагментами, а также изображения, искаженные шумами, различными по виду и параметрам.

Существующие количественные методы оценки качества изображения, например, пиковое отношение сигнал-шум [9] или интегральная оценка [15], выходят за рамки данного пособия.

3.2 Увеличение контрастности

В большинстве методов увеличения контрастности растрового изображения яркость пикселя преобразованного изображения $z'(x,y)$ зависит от яркости только соответствующего пикселя исходного изображения $z(x,y)$, что позволяет перед обработкой сформировать массив соответствия яркостей в соответствии с заданным алгоритмом $z'(x,y) = f(z)$. Такой массив значительно сокращает время обработки, т.к. после его формирования собственно обработка текущего пикселя исходного изображения сводится к присвоению соответствующему пикселю преобразованного изображения яркости, равной значению элемента массива с номером, равным яркости текущего пикселя исходного изображения, как показано на рисунке 3.2.

При функциональном преобразовании яркости пикселей преобразованного изображения не зависят от гистограммы яркости исходного изображения и формирование значений z' массива соответствия яркостей производится последовательным подставлением целочисленных значений яркости z в функцию преобразования шкалы яркости $f(z)$. При гистограммном преобразовании вид функции $f(z)$ зависит от значений гистограммы яркости исходного изображения и перед формированием массива соответствия яркостей необходимо построить нормализованную гистограмму яркости исходного изображения.

Изображения, полученные в неблагоприятных условиях освещенности, часто не используют весь диапазон возможных градаций яркости, что предопределяет их низкую информативность. Контрастность изображения, яркости элементов которого расположены в узком промежутке возможных значений, низкая. В результате соседние на изображении фрагменты имеют слишком близкую яркость, что и обуславливает их плохую различимость. Один из методов повышения качества таких изображений состоит в увеличении на максимально возможную величину разности значений яркости соседних фрагментов, например, линейным растяжением шкалы яркости [10]. Преобразование шкалы имеет вид:

$$z' = \frac{z - z_{\min}}{C_{om}}, \quad (3.3)$$

где z , z' – яркость пикселя до и после преобразования, z_{\min} – минимальная яркость исходного изображения, C_{om} – относительная контрастность изображения. Из (3.3) видно, что существенное увеличение различимости фрагментов достигается только при низкой контрастности исходного изображения.

По сравнению с плотной гистограммой исходного изображения, гистограмма преобразованного изображения оказывается растянутой и в ней появляются пропуски, но сами значения гистограммы не изменяются, а изменяются только яркости, к которым они относятся, как показано на рисунке 3.3. Именно наличие в какой-либо части гистограммы пропусков (разреженность), т.е. отсутствие на изображении пикселей некоторых идущих подряд яркостей, является косвенным признаком контрастности фрагментов, относящихся по яркости к этой части гистограммы.

При C_{om} близкой к единице увеличение контрастности линейным преобразованием всей шкалы яркости неэффективно. Но если известно, что плохо различимые фрагменты расположены в узком диапазоне яркости, то можно провести усечение шкалы яркости до требуемого диапазона с последующим кусочно-линейным преобразованием полученной шкалы [11]. В результате пиксели, яркость которых находится выше или ниже выбранного диапазона, получают соответственно максимально и минимально возможную яркость, а выделенный диапазон будет линейно растянут, как показано на рисунке 3.4. На гистограмме преобразованного изображения хорошо видно значительное увеличение числа пикселей максимальной яркости.

Нелинейное преобразование шкалы яркости или ее гамма-коррекция [3] при C_{om} близком к 1 позволяет увеличить контрастность в локальном диапазоне яркостей за счет снижения контрастности в других диапазонах. Например, если известно, что плохо различимые фрагменты имеют высокую яркость, то можно применить степенное преобразование шкалы яркости $z' = z_m z_{om}^n$, (3.4)

где $z_{om} = \frac{z}{z_m}$ – относительная яркость. Если яркости пикселей исходного изображения занимают весь возможный диапазон, то z_{om} изменяется в пределах 0...1. Значение n выбирается экспериментально, как правило, из диапазона $n = 2...10$. Как видно из графика на рисунке 3.5 в результате преобразования шкала яркости в зоне близкой к нулю оказывается сжатой, а в зоне близкой к z_m – растянутой, что приводит к повышению контрастности светлых фрагментов за счет понижения контрастности темных фрагментов, как показано на рисунке 3.6. Обратный результат дает преобразование (3.4) при диапазоне $n = 0,1...0,9$. Следует отметить, что гистограмма преобразованного согласно (3.4) изображения отличается по форме от гистограммы исходного изображения за счет частичного слияния близких по значению яркостей, как показано на рисунке 3.6. На исходном изображении (а) плохо различимы как темные области (лицо и детали арки), так и светлые (структура дорожки). Гамма-коррекция с $n > 1$ (б) увеличивает контрастность дорожки, а с $n < 1$ (в) – арки.

Аналогичные результаты можно получить, изменяя шкалу яркости логарифмическим или экспоненциальным преобразованием с соответствующими нормирующими коэффициентами [11].

Одновременно увеличить контрастность темных и светлых фрагментов изображения за счет частичного слияния фрагментов средней яркости можно, преобразовав (3.4) к виду [11] $z' = 0,5z_m(1 + (2z_{om} - 1)^n)$ для $n > 1$ нечетной степени. График функции, называемой степенным преобразованием со смещенным нулем, приведен на рисунке 3.5, а пример обработки показан на рисунке 3.7. На изображении одновременно повысилась различимость структуры дорожки и арки за счет слияния пикселей средней яркости (здание), что подтверждается появлением пика в средней части гистограммы.

Другим методом, позволяющим одновременно увеличить различимость, как темных, так и светлых фрагментов является соляризация [10], при которой преобразование шкалы яркости имеет вид $z' = 4z_{om}(z_m - z)$. (3.5)

Функция (3.5), описывающая данное преобразование, является квадратичной параболой, параметры которой подобраны так, чтобы $z' = z_m$ для $z = 0,5z_m$. Как видно из графика на рисунке 3.8 в результате преобразования пиксели, имеющие на исходном изображении яркость, симметричную относительно середины шкалы, приобретают одинаковую яркость. В результате фрагменты, имеющие близкую яркость, как в темной, так и в светлой области шкалы, разносятся по шкале на достаточное для увеличения их различимости расстояние, а фрагменты средней яркости приобретают яркость близкую к максимальной и частично сливаются, как показано на рисунке 3.9. На обработанном соляризацией изображении небо стало практически черным, дорожка резко потемнела и ее структура стала различима лучше, как и детали арки, а вот контрастность фасада здания уменьшилась. Метод называется «соляризация», т.к. создается впечатление, что светлые фрагменты «загорели» на солнце.

Увеличение различимости фрагментов изображения можно получить и путем эквализации [10] или выравнивания гистограммы, т.е. такого преобразования шкалы яркости, при котором гистограмма результирующего изображения будет приближаться по форме к равномерной. Метод эквализации гистограммы основан на предположении, что наибольшая контрастность достигается на изображении, гистограмма которого представляет равномерное распределение пикселей по яркостям на всем возможном диапазоне. Дискретное преобразование шкалы яркости

$$\text{имеет вид } z'_i = z_m \sum_{k=0}^i p(z_k), \quad (3.6)$$

где z'_i – значение элемента преобразованной шкалы яркости, соответствующей i -ой яркости исходной шкалы, $p(z_k)$ – нормализованная гистограмма яркости исходного изображения ($k = 0 \dots 255$).

В результате эквализации та часть гистограммы яркости исходного изображения, в которой соседние яркости имеют большие значения $p(z)$, растягивается за счет слияния тех областей гистограммы, в которых соседние яркости имеют низкие значения $p(z)$, как показано на рисунке 3.10 (а).

С помощью кумулятивных гистограмм можно так преобразовать изображение, чтобы его гистограмма имела требуемый вид [3]. Кумулятивная (накопительная) гистограмма строится по нормализованной гистограмме следующим образом: $p_c(z_i) = \sum_{k=0}^i p(z_k)$, что с точностью до постоянного множителя z_m совпадает с (3.6), причем $p_c(z)$ является монотонно возрастающей (неубывающей) функцией.

На первом этапе преобразования необходимо построить требуемую гистограмму и гистограмму обрабатываемого изображения, а также их кумулятивное представление $p'_c(z)$ и $p_c(z)$ соответственно. Далее для нахождения значений преобразованной шкалы яркости необходимо для каждой яркости z_i найти соответствующее ей значение кумулятивной гистограммы обрабатываемого изображения $p_c(z)$, после чего найти равное ему значение $p'_c(z')$ на требуемой кумулятивной гистограмме. Соответствующее значение яркости z' и будет искомым, т.е. $p'_c(z') = p_c(z)$. Графическая иллюстрация метода приведена на рисунке 3.11. На рисунке 3.11 (а) показаны исходная и желаемая гистограммы, а на рисунке 3.11 (б) – соответствующие им кумулятивные гистограммы и последовательность операций для получения значения преобразованной шкалы яркости.

Показанная на рисунке 3.11 (б) кумулятивная гистограмма для равномерного распределения пикселей по яркостям (эквализованная гистограмма) представляет собой прямую линию – наиболее простой случай преобразования гистограммы к требуемому виду. Пример кумулятивных гистограмм исходного изображения и преобразованного эквализацией гистограммы приведен на рисунке 3.10 (б). Хотя гистограмма обработанного изображения (а) не выглядит равномерной, кумулятивная гистограмма (б) точно соответствует графику, показанному на рисунке 3.11 для эквализации.

На рисунке 3.12 приведен пример преобразования изображения для получения гистограммы желаемого вида. На исходном изображении (а) имеется большое число мелких деталей, гистограмма яркости одномодальная и смещена в светлую область шкалы яркости, что отразилось на форме кумулятивной гистограммы. В качестве желаемой выбрана бимодальная гистограмма в виде двух треугольников

с равными высотами (b), соответствующие пикам гистограммы значения яркости – 20 и 230. Контрастность преобразованного изображения (c) увеличилась, средняя яркость понизилась, а гистограмма яркости стала бимодальной, причем в ней появились пропуски в области высокой и средней яркости.

Если необходимо увеличить контрастность темных областей, не изменяя контрастности светлых областей, или наоборот, то можно использовать метод адаптивного увеличения контрастности, основанный на локальных статистиках – средней яркости и дисперсии яркости в некоторой окрестности S преобразуемого пикселя [3]. В зависимости от соотношения этих характеристик по всему изображению и в окрестности пикселя определяется необходимость увеличения яркости пикселя. В данном случае преобразуется уже не шкала яркости, а яркость каждого пикселя изображения следующим образом:

$$z' = \begin{cases} k_0 z & \text{если } \mu_z \leq k_1 \mu; \quad k_2 \sigma \leq \sigma_z \leq k_3 \sigma \\ z & \text{в противном случае} \end{cases}, \quad (3.7)$$

где μ_z, σ_z – средняя яркость и среднеквадратичное отклонение яркости в окрестности S пикселя z ; μ, σ – средняя яркость и среднеквадратичное отклонение яркости по всему изображению; k_0, k_1, k_2, k_3 – нормирующие параметры. Размер локальной окрестности в преобразовании (3.7) следует выбирать так, чтобы с одной стороны – сохранить детали и уменьшить количество вычислений на каждом шаге, а с другой – получать достаточно достоверные μ_s, σ_s .

Значения нормирующих параметров зависят от конкретной задачи. Например, если необходимо увеличить контрастность темных областей, то коэффициент увеличения яркости k_0 устанавливается больше единицы, но в таких пределах, чтобы и после обработки сохранялся визуальный баланс яркостей на изображении (темные области не должны становиться слишком светлыми). Значение k_1 , определяющее порог срабатывания по критерию средней яркости, в данном случае выбирается меньше единицы, т.к. средняя яркость темных областей меньше средней яркости всего изображения. Подбор параметра k_2 позволяет исключить обработку областей с почти постоянной яркостью, в которых дисперсия яркости определяется только шумом. Параметр k_3 устанавливается меньше единицы, т.к. обра-

батывать следует только области с низкой контрастностью, в которых средне-квадратичное отклонение меньше, чем по всему изображению.

Результат применения к исходному изображению, показанному на рисунке 3.4 (а), адаптивного увеличения контрастности приведен на рисунке 3.13. Сравнивая его с результатом кусочно-линейного преобразования яркости, показанным на рисунке 3.4 (б), можно заметить, что в отличие от последнего на рисунке 3.13 сохранилось изображение башни с часами и лучше различимы детали фасада на здании в левой стороне снимка.

Локальное увеличение контрастности фрагмента изображения можно получить, используя гистограмму яркости некоторой окрестности преобразуемого пикселя [12]. После полученной локальной гистограммы выполняется требуемое преобразование шкалы яркости, например, эквализация, и по его результату определяется яркость центрального пикселя окрестности. Далее центр окрестности смещается на следующий пиксель и процедура повторяется. Возможно использование непересекающихся областей для вычисления локальных гистограмм, но такой более быстрый метод часто приводит к появлению на обработанном изображении псевдо контуров на границах областей.

Результат локального увеличения контрастности эквализацией гистограммы яркости приведен на рисунке 3.14. Различимость деталей по всему изображению значительно возросла, но результат потерял естественность.

Неравномерное освещение объектов сцены может привести к тому, что на изображении появятся как излишне, так и недостаточно освещенные фрагменты. Для компенсации неравномерности освещения рекомендуется следующая последовательность действий [15]. В начале изображение обрабатывается усредняющим фильтром, например, гауссианом, с большим размером апертуры. При правильном подборе размера апертуры результат обработки $z_G(x,y)$ представляет собой приближенное распределение освещенности по изображению. Далее для каждого пикселя (x,y) исходного изображения производится деление его яркости на соответствующее значение освещенности, т.е. $z'(x,y) = C_{norm} \frac{z(x,y)}{z_G(x,y)}$. Норми-

рующий коэффициент C_{norm} подбирается таким образом, чтобы полученные значения $z'(x,y)$ занимали весь диапазон возможных значений яркости (0 – 255). Результат обработки изображения с существенно неравномерным освещением приведен на рисунке 3.15. Сравнивая исходное (а) и обработанное (б) изображения, можно заметить полное исчезновение светлого пятна (солнце) и некоторое увеличение контрастности, но все изображение слишком темное. Для повышения различимости деталей к изображению (б) применено кусочно-линейное растяжение в диапазоне 0 – 90 (с). Следует отметить, что для повышения различимости деталей часто приходится применять последовательную обработку несколькими различными методами и экспериментально подбирать параметры обработки.

3.3 Увеличение резкости

Из (3.2) следует, что увеличение резкости возможно как при увеличении локальной контрастности C^l фрагмента, так и при уменьшении ширины зоны границы фрагмента d . На практике существенно увеличить контрастность фрагментов изображения, а тем более уменьшить ширину границы между ними возможно далеко не всегда.

Визуально увеличить резкость фрагментов без изменения контрастности можно путем подчеркивания границ контурной линии, отделяющей фрагменты друг от друга, т.е. изменения яркости тех пикселей, которые находятся на границе линии контура, отделяющей один фрагмент от другого [9]. При этом яркость темных граничных пикселей понижается, а яркость светлых – повышается. В результате локальная контрастность C^l для пикселей граничной линии l увеличивается при сохранении толщины граничной линии, т.е. увеличивается и резкость R согласно (3.2). Эффективное увеличение резкости таким способом возможно только при условии, что яркость пикселей соседних фрагментов исходного изображения находится в среднем диапазоне шкалы яркости.

Основная проблема, возникающая при применении данного подхода, заключается в следующем. С одной стороны, требуется увеличить резкость фраг-

ментов с малой разностью яркостей, а с другой – не изменять резкость хорошо различимых фрагментов и не увеличивать резкость шумовой составляющей.

Для определения величины увеличения резкости можно использовать вторую производную функции яркости изображения, т.е. свертку фильтром Лапласа [9]. Тогда $z' = z - k_L z_L$, (3.8)

где k_L – коэффициент усиления резкости фильтром Лапласа, выбираемый экспериментально из диапазона $k_L = 1 \dots 5$, z_L – результат свертки окрестности 3×3 пикселя яркостью z с ядром H_L , причем z_L берется с учетом знака. Дискретные маски фильтра Лапласа H_L приведены в (2.3).

Если перепадов яркости в окрестности нет, то $z_L = 0$ и $z' = z$. Если центр окрестности находится в начале границы перехода от низкой яркости к высокой, то $z_L > 0$ и согласно (3.8) $z' < z$. Если центр окрестности в конце этой границы, то $z_L < 0$ и соответственно $z' > z$. Следовательно, диапазон яркости границы увеличивается приблизительно на $2k_L z_L$.

Подчеркивание границ фрагментов можно получить, используя методы нерезкого или нечеткого маскирования [3]. Например, в результате свертки изображения с гауссианом можно получить размытое изображение. Маска гауссиана размером 3×3 приведена в (1.15). Тогда $z' = z_G + k_G(z - z_G)$, (3.9)

где k_G – коэффициент усиления резкости Гауссианом, выбираемый экспериментально из диапазона $k_G = 1 \dots 10$, z_G – яркость соответствующего пикселя изображения, полученного в результате свертки с Гауссианом. Если перепадов яркости в окрестности центрального пикселя с яркостью z нет, то $z = z_G$ и согласно (3.9) $z' = z$, т.е. изменения яркости пикселя на преобразуемом изображении не происходит. Если центр окрестности расположен в начале границы перехода от низкой яркости к высокой, то $z_G > z$ и согласно (3.9) $z' < z$ при $k_G > 1$. Если центр окрестности в конце этой границы, то $z_G < z$ и соответственно $z' > z$. Следовательно, диапазон яркости границы увеличивается приблизительно на $2(k_G - 1)|z - z_G|$.

На рисунке 3.16 приведены исходное размытое изображение (а) и результаты увеличения резкости с использованием фильтра Лапласа (б) и нерезким маскированием (с) с разрезами функции яркости по одной и той же строке. Из рисунка

3.16 видно, что оба метода дают вполне удовлетворительные результаты, но при одинаковом коэффициенте усиления ($k_L = k_G = 5$) резкость увеличивается больше с использованием фильтра Лапласа, чем при нерезком маскировании. Однако, с точки зрения визуальной естественности, результаты обработки нерезким маскированием предпочтительнее.

3.4 Адаптивное увеличение резкости

Основной недостаток методов определения яркости текущего пикселя для увеличения резкости согласно (3.8) и (3.9) заключается в том, что коэффициент усиления остается постоянным для всего изображения. В результате степень усиления резкости остается одинаковой при любом перепаде яркости в окрестности текущего пикселя. Желательно иметь возможность изменять коэффициент усиления резкости в зависимости от величины разброса яркостей в окрестности, т.е. использовать методы адаптивного вычисления коэффициента усиления резкости для каждого пикселя обрабатываемого изображения.

Например, коэффициент усиления резкости в (3.9) можно изменять в зависимости от дисперсии яркости окрестности следующим образом [15]:

$$k_G = k_n \frac{Z_c}{\sqrt{D_z}}, \quad (3.10)$$

где $k_n = 0 \dots 1$ – постоянный нормирующий множитель, Z_c – средняя яркость всего (!) изображения, D_z – дисперсия яркости окрестности обрабатываемого пикселя, размер которой подбирается экспериментально.

Согласно (3.10) при одинаковой дисперсии коэффициент усиления резкости k_G больше для тех изображений, у которых Z_c ярче. Это означает, что резкость фрагментов двух одинаковых изображений, отличающихся только средней яркостью, будет разной. Иногда более эффективно использовать отношение

$k_G = \frac{k_0}{\sqrt{D_z}}$, где k_0 – задаваемая константа.

Из (3.10) видно, что чем меньше D_z , тем больше коэффициент усиления. Если D_z близко к нулю, то следует установить $k_G = 1$, тогда яркость равномерных и слабо зашумленных участков изображения не будет изменяться. Пороговое значение D_z выбирается экспериментально. Отсутствие проверки величины D_z приводит к резкому усилению шума, что в большинстве случаев нежелательно. Однако иногда такое усиление может оказаться полезным, т.к. позволит выявить на изображении практически неразличимые фрагменты. На рисунке 3.17 приведено изображение с разным уровнем размывания областей и плохо различимым фрагментом (а), результаты адаптивного усиления резкости с пороговым ограничением дисперсии яркости окрестности (б) и без порога (с). В результате усиления яркости без порогового ограничения дисперсии окрестности в нижней части изображения проявилась структура, яркость которой на исходном изображении отличается от яркости фона только на одну градацию.

Хорошие результаты дает адаптивное увеличение резкости на основе локальной контрастности [15], который вычисляется как $C_z = \frac{|z - z_c|}{z + z_c}$, (3.11)

где z_c – средняя яркость некоторой окрестности пикселя яркостью z . Нелинейное усиление локальной контрастности C_z^* получается из $C_z^* = f(C_z)$, где $f(C_z)$ – нелинейная, монотонная функция, которая удовлетворяет условиям: $f(C_z) > C_z$, $C_z = 0 \dots 1$, $f(C_z) = 0 \dots 1$, а C_z определяется согласно (3.10). В качестве функции f часто используют степенную функцию $C_z^* = C_z^n$, (3.12)

где $n = 0, 1 \dots 1, 0$. Результирующая яркость центрального пикселя окрестности вы-

числяется согласно следующему правилу:
$$z' = \begin{cases} z_c \frac{1 - C_z^*}{1 + C_z^*} & \text{при } z < z_c \\ z_c \frac{1 + C_z^*}{1 - C_z^*} & \text{при } z > z_c \end{cases} . \quad (3.13)$$

Из уравнения (3.12) следует, что при $n = 1$ $C_z^* = C_z$ и соответственно $z' = z$. Чем меньше n , тем больше C_z^* относительно C_z , а два варианта вычисления z' в зависимости от соотношения z и z_c позволяют увеличить резкость границы между

фрагментами с близкой яркостью, причем несимметрично для темных ($z < z_c$) и светлых ($z > z_c$) фрагментов.

В таблице 11 для резкого изменения яркости приведены значения локальной контрастности в линейной окрестности размером 5 пикселей, результаты адаптивного увеличения резкости при $n = 0,5$ и значения относительного коэффициента резкости, который вычисляется как $R_z = \frac{|z - z_c|}{\Delta}$, где Δ – величина перепада яркости между соседними фрагментами. Из таблицы 11 видно, что:

- значение локальной контрастности соответствует уровню перепада яркости между соседними фрагментами;
- увеличение резкости возрастает с ростом перепада яркости;
- относительный коэффициент резкости обратно пропорционален величине перепада яркости при постоянном значении n .

Результаты моделирования адаптивного увеличения резкости одного перепада яркости при разных значениях n приведены в таблице 12 и на рисунке 3.18. Из них видно, что при малых значениях n увеличение резкости слишком велико, что может привести к снижению качества визуального восприятия изображения и различимости фрагментов.

Результаты моделирования адаптивного увеличения резкости плавного перепада яркости при $n=0,5$ для линейной окрестности в 5 пикселей приведены в таблице 13. Из нее видно, что больше всего изменяется яркость пикселей, находящихся на границе перепада яркости, а яркость пикселей в средней части области перепада не изменяется, т.е. происходит именно подчеркивание границ контурной линии.

На рисунке 3.19 показан результат обработки исходного изображения с плохо различимыми фрагментами в темной области усилением локальной контрастности. Естественность полученного изображения существенно снизилась, но различимость малоконтрастных деталей возросла.

В уравнении (3.12) значение n остается постоянным при любых значениях локальных контрастностей, что приводит к нежелательному усилению как резко-

сти фрагментов с высокой контрастностью, так и шумовой составляющей изображения. Избавиться от этого недостатка можно, выбирая для каждого пикселя значения n в зависимости от характеристик локальной окрестности, т.е. используя адаптивную степень усиления локальной контрастности [15] n_z . Например, для (3.11) можно вычислять $n = n_z$ следующим образом:

$$n_z = n_{\min} + (n_{\max} - n_{\min})k_n^s, \quad (3.14)$$

где n_{\min} , n_{\max} – минимальное и максимальное значения степени n соответственно ($n_{\min} = 0,1 \dots 0,3$, $n_{\max} = 0,7 \dots 0,9$), k_n – коэффициент адаптации ($k_n = 0 \dots 1$), s – постоянный степенной коэффициент ($s > 1$). Согласно (3.12), максимальное усиление локальной контрастности достигается при $k_n = 0$. т.е. при $n_z = n_{\min}$, а минимальное – при $k_n = 1$. т.е. при $n_z = n_{\max}$.

В качестве характеристики локальной окрестности пикселя для задания k_n из (3.14) часто используются [15]: протяженность гистограммы L_z , энтропия ε_z или среднеквадратичное отклонение яркости σ_z . Следует отметить, что размеры окрестности для вычисления этих характеристик могут существенно отличаться от размеров окрестности для вычисления локальной контрастности. Рекомендуемый в литературе [15] размер окрестности для вычисления этих характеристик составляет 15 - 29 пикселей. Данный метод часто применяется для обработки аэрофотоснимков с мелкими малококонтрастными деталями.

Протяженность гистограммы локальной окрестности обрабатываемого пикселя определяется как $L_z = \frac{z_{\max} - z_{\min}}{P_{\max}}$, (3.15)

где z_{\min} , z_{\max} – минимальное и максимальное значения яркости в окрестности соответственно, P_{\max} – максимальное значение гистограммы яркости окрестности.

Рассмотрим зависимость L_z от формы гистограммы локальной окрестности. Из (3.15) следует, что фрагменты с постоянной яркостью имеют $L_z = 0$. т.к. на них $z_{\min} = z_{\max}$. Фрагменты с равномерной гистограммой и максимально возможным диапазоном яркости имеют $L_z = 255$, при условии, что число пикселей окрестности $N > 255$, т.е. размер окрестности 16 пикселей ($16 \times 16 = 256$) и более, т.к. на них, ориентировочно, $z_{\min} = 0$, $z_{\max} = 255$, $P(z) = P_{\max} = 1$. У фрагментов с бимо-

дальной гистограммой L_z зависит от диапазона яркостей и соотношения числа пикселей в окрестности с минимальной и максимальной яркостью. При условии равенства этих чисел $L_z = \frac{\Delta}{0,5N}$, где Δ – диапазон яркости. Желательно определять k_n так, чтобы $k_n = 1$ при $L_z = 0$, k_n стремилось к нулю при L_z стремящемся к a и k_n стремилось к 1 при L_z стремящемся к единице и далее. Данному условию отвечает уравнение

$$k_n = 1 - \exp\left[-\frac{(L_z - a)^2}{0,0392}\right], \quad (3.16)$$

где a – параметр, настраиваемый экспериментально. Известно [15], что хорошие результаты повышения резкости согласно (3.15) можно получить при $a = 0,5$.

Энтропия характеризует степень неопределенности совокупности событий

и для окрестности пикселя определяется как:
$$\varepsilon_z = -\frac{\sum_{i=1, N} p_i \log_2(p_i)}{\log_2(N)}, \quad (3.17)$$

где p_i – вероятность наличия у i -го пикселя яркости z , т.е. значение нормализованной гистограммы яркости окрестности, N – число пикселей в окрестности, i – номер пикселя из окрестности пикселя z . Из (3.17) следует, что энтропия приобретает минимальное значение на однородных фрагментах, а максимальное – на фрагментах с элементами, значения яркостей которых находятся на противоположных краях диапазона. В таком случае, для получения адаптивного усиления резкости можно задать k_n следующим образом

$$k_n = \frac{\varepsilon_z - \varepsilon_{\min}}{\varepsilon_{\max} - \varepsilon_{\min}}, \quad (3.18)$$

где ε_{\max} , ε_{\min} – соответственно максимальное и минимальное значение энтропии окрестностей по всему (!) изображению для заданной окрестности.

Среднеквадратичное отклонение яркости локальной окрестности σ_z вычисляется как

$$\sigma_z = \sqrt{\frac{1}{N} \sum_{i=1, N} (z_i - z_c)^2}, \quad (3.19)$$

где z_i – яркость пикселя из заданной окрестности, z_c – средняя яркость окрестности, N – число пикселей в окрестности.

Согласно (3.19) среднеквадратичное отклонение ведет себя аналогично предыдущей характеристике локальной окрестности, следовательно, для вычисления k_n вполне можно использовать формулу (3.18), заменив ε на σ .

На рисунке 3.20 приведено исходное изображение с мелкими плохо различимыми деталями и результаты его обработки с использованием протяженности гистограммы, энтропии и среднеквадратичного отклонения яркости для нахождения значения адаптивной степени усиления локальной контрастности. На рисунке видно, что все даже минимальные перепады яркости резко усилились, и теперь остается только решить – где проявились действительно существующие мелкие детали, а где только усилился шум?

4. Программная реализация методов обработки изображений

Программная реализация рассмотренных выше методов обработки растровых изображений имеет некоторые особенности. Приведенные далее примеры соответствуют приложениям, написанным в среде визуального объектно-ориентированного программирования Delphi [14], но аналогичные процедуры можно найти и в других популярных средах программирования.

Открытие изображения удобно выполнять с помощью диалогового окна «Открыть изображение» (класс `TOpenPictureDialog`). После открытия изображение хранится и отображается в компоненте класса `TImage`. В качестве формата исходного файла рекомендуется использовать bmp-формат, т.к. в нем изображение хранится в исходном (несжатом) виде. Применение форматов, хранящих изображение в сжатом виде с потерями информации, например, в jpg-формате, может привести к появлению нежелательных эффектов в результате обработки, например, выявлению границ блочной структуры, используемой в алгоритме сжатия методом JPEG [16]. Следует отметить, что в среде Delphi имеется модуль JPEG, позволяющий открывать изображения jpg-формата.

Пример процедуры загрузки изображения bmp-формата в созданный визуальный компонент TImage с использованием диалогового компонента OpenPictureDialog приведен ниже.

```
procedure TFimg.NOpenClick(Sender: TObject);  
// реакция на нажатие кнопки ОК в диалоговом окне выбора  
// файла изображения OpenPictureDialog  
begin  
    if OpenPictureDialog.Execute then  
        Image.Picture.LoadFromFile(OpenPictureDialog.FileName);  
end;
```

Для более быстрой обработки изображения желательно иметь несколько компонентов, в которых оно хранится, т.к. при обработке невидимого пользователю изображения время на постоянную прорисовку каждого изменения не затрачивается. Кроме того, алгоритмы, реализующие различные фильтры, формируют результирующее изображение на основе исходного, не допуская изменение последнего в процессе обработки. Пример использования визуальных компонентов-контейнеров класса TImage и их невидимых составляющих типа TBitmap для копирования содержимого изображения (именно копирования данных, а не ссылки по указателю) приведен ниже.

```
var  
    Bitmap: TBitmap;  
    Image: TImage;  
begin  
    Bitmap := TBitmap.Create;  
    Bitmap.Assign(Image.Picture); //из Image - в Bitmap  
    . . . .  
    Image.Picture.Assign(Bitmap); //из Bitmap - в Image  
    Bitmap.Free;  
end;
```

Иногда бывает необходимо скопировать в созданное пустое изображение некоторую часть исходного изображения. Это можно сделать, например, таким образом.

```
var
  Bitmap: TBitmap;
  BitmapArea: TRect;
  Image: TImage;
begin
  Bitmap := TBitmap.Create;
  with BitmapArea do
    begin
      Left := 0; Top := 0;
      Right := Image.Width - 1; Bottom := Image.Height - 1;
    end;
  with Bitmap do
    begin
      Height := Image.Height; Width := Image.Width;
      Canvas.CopyMode:= CMSrcCopy;
      Canvas.CopyRect(BitmapArea, Image.Canvas, BitmapArea);
    end;
  Image.Picture.Assign(Bitmap);
  Bitmap.Free;
end;
```

Использование линеек процесса (классы TProgressBar или TGauge) может существенно замедлить работу приложения, но в долгих циклических процедурах метод Application.ProcessMessages создаёт эффект независимого приложения (выполняется его периодическое обновление, а также обработка системных событий). Аналогичный эффект создаёт изменение вида курсора установкой свойства Screen.Cursor := crHourglass (песочные часы); в начале

работы процедуры и возврат к курсору в виде стрелки после окончания обработки
- Screen.Cursor := crDefault.

Для работы с отдельными пикселями изображения можно использовать массив `Bitmap.Canvas.Pixels[X,Y]`, так как он представляет собой двумерный массив всех пикселей, в каждом элементе которого хранится цвет пикселя (класс `TColor`). Один пиксель описывается четырьмя байтами, где самый старший байт – служебный, а остальные – байты значений цветовых каналов: голубого – B, зеленого – G и красного – R (младший байт). Примеры получения доступа к отдельным каналам для выделения нужных байтов из четырехбайтового значения с использованием функций Delphi или логических операций `and` и `or` и операций сдвига `shl` и `shr` приведен ниже:

```
var
    B, G, R: Byte;
    Image: TImage;
    Pixel: TColor;
begin
    Pixel := Image.Canvas.Pixels[0,0];
    G := GetGValue(Pixel);
    R := GetRValue(Pixel);
    B := GetBValue(Pixel);
//или
    R := Pixel and $000000FF;
    G := (Pixel and $0000FF00) shr 8;
    B := (Pixel and $00FF0000) shr 16;
//
    ...
// изменение цвета пикселя
    Image.Canvas.Pixels[0,0] := rgb(R,G,B);
//или
    Pixel := B shl 8;
```

```

Pixel := (Pixel or G) shl 8;
Pixel := Pixel or R;
Image.Canvas.Pixels[0,0] := Pixel;
//
end;

```

Необходимо отметить, что обработка изображения через массив `Pixels[X,Y]` из-за индексации по массиву требует значительных ресурсов и времени. Избежать этого можно, используя метод `Bitmap.ScanLine`, который возвращает указатель на целую строку пикселей. К сожалению, элементы строки, на которую указывает `ScanLine`, это уже не 4-байтовый цвет пикселя, а реальные единицы информации, соответствующие цветности конкретного изображения (для чёрно-белого изображения – 1 бит, для 4-цветного – 2 бита и т.д., для полноцветного – 3 или 4 байта). В результате при обращении к отдельному пикселю в малоцветных (от 256 цветов и меньше) изображениях могут возникнуть трудности, вызванные всё теми же побитовыми операциями для выделения значений цветовых каналов. Для таких изображений рекомендуется с помощью какого-либо графического редактора принудительно приводить цветность к 65 тысячам цветов до загрузки изображения в программу обработки. Пример получения доступа к отдельным цветовым каналам полноцветного (трехбайтовый цвет) изображения с использованием метода `Bitmap.ScanLine` приведен ниже.

```

var
  B, G, R: Byte;
  Image: TImage;
  P: PByteArray;
begin
  Image.Picture.Bitmap.PixelFormat := PF24Bit;
  P := Image.Picture.Bitmap.ScanLine [0];
  X := 0;
  repeat
    R := P [X + 2];

```

```

G := P [X + 1];
B := P [X];
X := X + 3;
until X > 3 * Image.Width - 1;
end;

```

В монохромных изображениях формата BMP значения цветовых каналов R , G , B у каждого пикселя совпадают между собой и соответствуют значению яркости данного пикселя, т.е. для любого i -го пикселя $R = G = B = z_i$. Именно это значение и следует использовать, применяя приведенные в разделах 1 – 3 формулы для обработки изображения. Полученное новое значение яркости z_i' определяет измененные значения цветовых каналов $R' = G' = B' = z_i'$.

В полноцветных изображениях формата BMP значения цветовых каналов R , G , B у каждого пикселя в общем случае отличаются друг от друга. Применение описанных в разделах 1 – 3 методов обработки к каждому цветовому каналу по отдельности, может привести не только к изменению яркости пикселя, но и к искажению его цветового оттенка, что нежелательно. Избежать такого эффекта можно, выделив яркостную Y и цветовые I , Q составляющие полноцветного пикселя, например, по формулам [1]:

$$\begin{aligned}
Y &= 0.299 * R + 0.587 * G + 0.114 * B, \\
I &= 0.596 * R - 0.275 * G - 0.321 * B, \\
Q &= 0.212 * R - 0.523 * G + 0.311 * B.
\end{aligned}
\tag{4.1}$$

Значение Y характеризует яркость i -го пикселя z_i и изменяется в результате обработки изображения. Получив новое значение яркости $z_i' = Y'$ обработанного пикселя, необходимо по формулам (4.2) определить измененные значения цветовых каналов R' , G' , B' , используя новое значение яркостной составляющей Y' с вычисленными по (4.1) и оставшимися без изменений цветовыми составляющими пикселя I , Q .

$$\begin{aligned}
R' &= Y' + 0.956 * I + 0.621 * Q, \\
G' &= Y' - 0.272 * I - 0.647 * Q, \\
B' &= Y' - 1.107 * I + 1.704 * Q.
\end{aligned}
\tag{4.2}$$

Применение формул (4.2) для вычисления значений цветовых каналов обработанного пикселя может привести к выходу некоторых полученных значений за пределы разрешенного диапазона градаций (0 – 255). В процессе обработки изображения такие события необходимо отслеживать и заменять вышедшее из диапазона значение цветового канала на ближайшее допустимое.

Следует отметить, что наряду с (4.1) и (4.2) в литературе [2, 3] можно найти и другие формулы для выделения яркостной и цветовых составляющих пикселя.

Экспериментальные исследования показывают [3], что при подавлении яркостных помех на полноцветных изображениях хорошие результаты можно получить, применяя фильтрацию по каждому цветовому каналу отдельно.

При использовании методов, основанных на вычислении каких-либо характеристик окрестности пикселя, возникает проблема пограничной области изображения. Она состоит в том, что при вычислении характеристик окрестности пикселя, находящегося от края изображения на расстоянии меньшем радиуса окрестности, отсутствуют какие-либо данные о яркости пикселей, координаты которых находятся вне изображения.

Существует три способа решения данной проблемы [6]:

- яркость такого пикселя принимается равной яркости ближайшего к нему пикселя изображения;
- производится расширение обрабатываемого изображения до необходимых размеров путем его зеркального отображения относительно краев;
- обработанное изображение уменьшается по краю на радиус окрестности путем обрезания его пограничной области.

Если размер окрестности не превышает 3 – 7 пикселей, то можно рекомендовать первый способ, как наиболее простой в реализации. Для больших окрестностей предпочтение следует отдать второму способу, т.к. он снижает влияние крайних пикселей на результат вычисления характеристик окрестности. Третий способ можно использовать при условии, что информативные фрагменты находятся в центральной части изображения.

Приведем некоторые типичные ошибки, встречающиеся при программировании различных методов обработки изображений.

При преобразовании данных из одного типа в другой часто не учитывается возможность переполнения, например:

```
var s: Integer; z: Byte;  
.  
.  
.  
.  
.  
z:=s;
```

В приведенном фрагменте значение переменной целого типа s (диапазон $-\pm 32000$) присваивается переменной байтового типа z (диапазон $0 \dots 255$). Фактически значения младших восьми бит переменной s записываются в восемь бит, отведенные под переменную z , т.е. при $s = 256$ (в двоичном коде $\dots 10000000$) $z = 0$. Если z – яркость пикселя, то на изображении вместо пикселя максимальной яркости получается пиксель минимальной яркости. Для устранения ошибки следует перед присвоением s изменением типа проверять значение переменной s большим диапазоном, например:

```
if s > 255 then z:=255 else z:=s;
```

Отсутствие анализа используемых формул с целью минимизации вычислений приводит к неоправданному увеличению времени обработки изображения. Например, в формуле (3.6) для эквализации гистограммы яркости используется нормализованная гистограмма яркости исходного изображения $p(z_k)$, каждое значение которой есть отношение числа пикселей яркости z_k к числу пикселей всего изображения. Преобразовав формулу (3.6), можно сократить время вычислений за счет уменьшения числа операций деления и замены в цикле суммирования действительных чисел $p(z_k)$ на целые $P(z_k)$, например,

$$z'_i = z_m \sum_{k=0}^i p(z_k) = z_m \sum_{k=0}^i \frac{P(z_k)}{N} = \frac{z_m}{N} \sum_{k=0}^i P(z_k), \text{ где } N - \text{число пикселей.}$$

Для наглядности отображения результатов обработки следует, планируя интерфейс, разместить рядом исходное и обработанное изображения, их гистограммы яркости и разрезы функции яркости, причем последние желательно выполнять по одной и той же строке или столбцу.

Приведенные фрагменты исходных текстов, внимательный анализ конкретного алгоритма обработки и продуманный интерфейс позволяют создать эффективный программный продукт в форме Windows-приложения, реализующий загрузку, отображение и обработку растровых изображений, для которых требуется доступ к значениям цветовых каналов отдельных пикселей.

5. Задание к практическим занятиям

1. Изучить алгоритм работы и особенности заданного метода (методов) обработки изображения в соответствии с полученным вариантом задания, вычислить результаты обработки типичных фрагментов (импульсы, перепады яркости).

2. Разработать интерфейс и написать программу, реализующую данный метод (методы), в форме Windows-приложения. Приложение должно обеспечивать:

- ввод и отображение на дисплее монохромного (по желанию студента – полноцветного изображения) растрового исходного изображения формата bmp;
- возможность настройки параметров обработки (размеры апертуры и тип фильтра, коэффициенты усиления и т.п.);
- максимально эффективную программную реализацию обработки заданным методом (методами) исходного изображения;
- отображение результата обработки на экране дисплея и сохранение его в заданном пользователем файле;
- отображение на дисплее координат и яркости любого выбранного пользователем пикселя исходного изображения и яркости соответствующего ему пикселя обработанного изображения, а также дополнительные возможности для оценки результата работы, например, гистограммы яркости и разрезы функций яркости изображений по строкам и столбцам.

3. Подготовить с помощью графических редакторов нескольких тестовых изображений, позволяющих проверить правильность работы программы, и подобрать реальные изображения, демонстрирующие особенности, преимущества и

недостатки работы заданного метода (методов) при различных настройках параметров.

4. Провести эксперименты по обработке подготовленных тестовых и реальных изображений, сделать подробный анализ полученных результатов.

5. Оформить отчет о проделанной работе, который должен включать в себя:

- полученное задание;
- описание метода обработки, особенностей его применения и программной реализации;
- вычисленные результаты обработки типичных фрагментов и ожидаемые результаты обработки тестовых изображений;
- текст программы с подробными комментариями;
- описание интерфейса;
- исходные тестовые и реальные изображения, результаты их обработки с дополнительной информацией (гистограммы, разрезы функции яркости и т.п.) и комментариями;
- выводы.

Варианты заданий:

1. Среднегеометрический и среднегармонический фильтры.
2. Гамма-коррекция, соларизация и преобразование со смещенным нулем.
3. Адаптивное увеличение резкости с учетом локальной контрастности.
4. Выделение контура фильтрами Лапласа с разными масками и Робертса.
5. Среднеарифметический и адаптивный усредняющий фильтры.
6. Адаптивное увеличение резкости с учетом дисперсии окрестности.
7. Выделение контура фильтрами Собеля, Превитт и Уоллеса.
8. Двумерные и одномерные гауссианы с разным размером маски.
9. Эквализация и получение требуемой (бимодальной) гистограммы яркости.
10. Выделение контура LoG-фильтрами с разным размером маски.
11. Выделение линии контура фильтрами Робертса и Кирша.
12. Локальное увеличение контрастности эквализацией гистограммы яркости.

13. Фильтр срединной точки и фильтр Кувахары.
 14. Обычный и адаптивный медианные фильтры.
 15. Адаптивное увеличение контрастности с учетом локальной статистики.
 16. Фильтр Кувахары и фильтр усеченного среднего.
 17. Медианный и контргармонический фильтры.
 18. Линейное и кусочно-линейное преобразование шкалы яркости.
 19. Повышение резкости фильтром Лапласа и нечетким маскированием.
 20. Медианный, минимальный и максимальный фильтры.
 21. Изменение степени увеличения резкости с учетом среднеквадратичного отклонения яркости пикселей окрестности.*
 22. Компенсация неравномерности освещения.*
 23. Изменение степени увеличения резкости с учетом энтропии яркости пикселей окрестности.*
 24. Получение контурной линии немаксимальным подавлением.*
 25. Изменение степени увеличения резкости с учетом протяженности гистограммы яркости пикселей окрестности.*
- Усложненные варианты заданий помечены*.

Литература

1. Красильников Н.Н. Цифровая обработка изображений. – М.: Вузовская книга, 2001. – 320 с.
2. Шапиро Л., Стокман Дж. Компьютерное зрение. – М.: БИНОМ. Лаборатория знаний, 2006. – 752 с.
3. Гонсалес Р., Вудс Р. Цифровая обработка изображений. – М.: Техносфера, 2005. – 1072 с.
4. Прэтт У. Цифровая обработка изображений – М.: Мир, 1982. – 790 с.
5. Форсайт Д., Понс Ж. Компьютерное зрение. Современный подход. – М.: Вильямс, 2004. – 928 с.
6. Ерош И.Л., Сергеев М.Б., Соловьев Н.В. Обработка и распознавание изображений в системах превентивной безопасности: Учеб. пособие. – СПб.: СПбГУАП, 2005. – 154 с.
7. Монич Ю.И., Старовойтов В.В., Коноплин Е.Е. Сегментация примерно однородных по яркости областей цифровых изображений // Искусственный интеллект – 2008, № 3 – С. 332 – 338.
8. Мироновский Л.А., Слаев В.А. Алгоритмы оценивания результатов трех измерений. – СПб.: «Профессионал», 2010. – 192 с.
9. Методы компьютерной обработки изображений / под ред. В.А.Сойфера. – М.: ФИЗМАТЛИТ, 2003. – 784 с. 91
10. Гузман И.С. и др. Цифровая обработка изображений в информационных системах – Новосибирск: НГТУ, 2000. – 156 с.
11. Сергеев М.Б., Соловьев Н.В., Стадник А.И. Методы повышения контрастности растровых изображений для систем цифровой обработки видеoinформации // Информационно-управляющие системы – 2007, №1 – С. 2 – 7.
12. Фу К. и др. Робототехника: Пер.с англ. / Под ред. В.Г.Градецкого. – М.: Мир, 1989. – 624 с.
13. Ватолин Д., Ратушняк А., Смирнов М., Юкин В. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео. - М.: ДИАЛОГ-МИФИ, 2002. – 384 с.

14. Керман М.К. Программирование и отладка в Delphi. Учебный курс. – Киев: Вильямс, 2004. – 720 с.
15. Журавель И.М. Краткий курс теории обработки изображений. – информационных ресурс Интернет <http://matlab.exponenta.ru>
16. Все о сжатии. Методы сжатия медиаданных. – информационных ресурс Интернет <http://www.compression.ru>

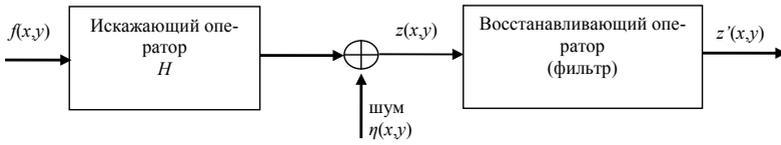


Рисунок 1.1 – Модель процесса искажения/восстановления изображения.

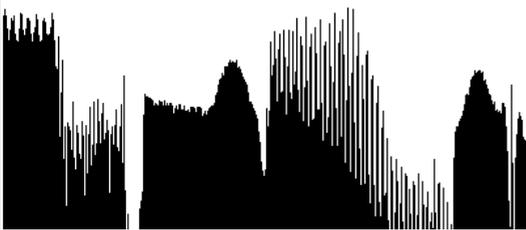


Рисунок 1.2 – Исходное тестовое изображение (Барбара) [16] и разрез яркости по строке 220.

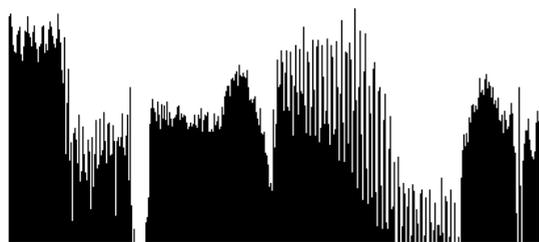
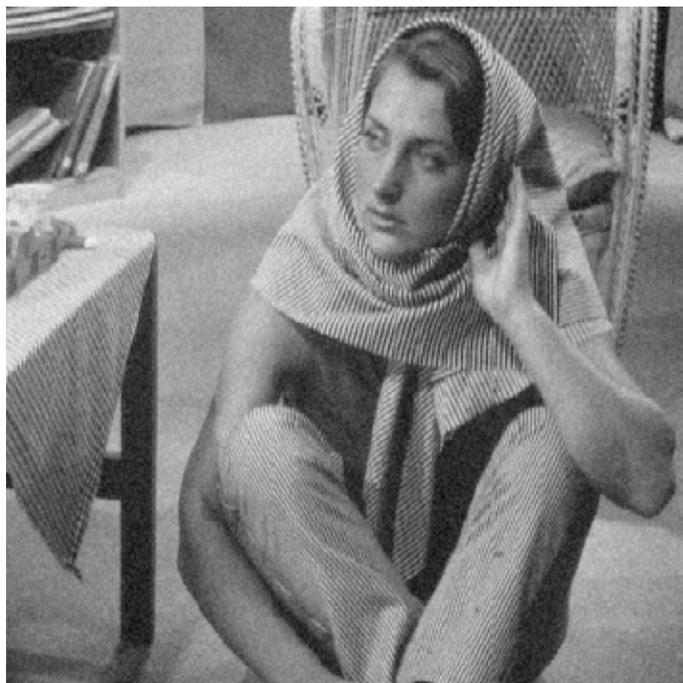


Рисунок 1.3 – Результат искажения изображения 1.2 «белым» шумом с амплитудой ± 20 .

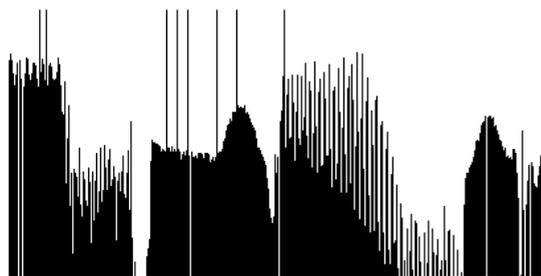
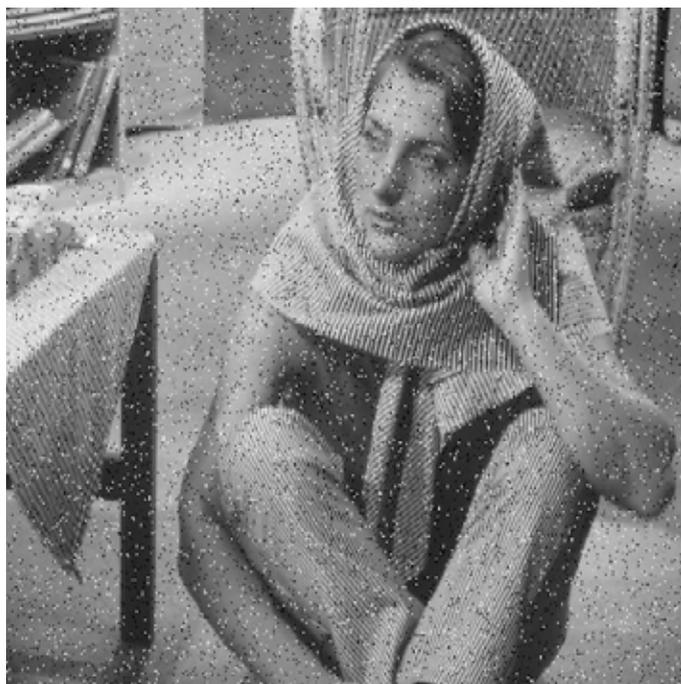


Рисунок 1.4 – Результат искажения изображения 1.2 шумом «соль и перец» с частотой 5%.

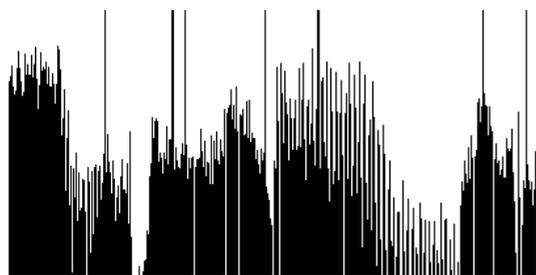
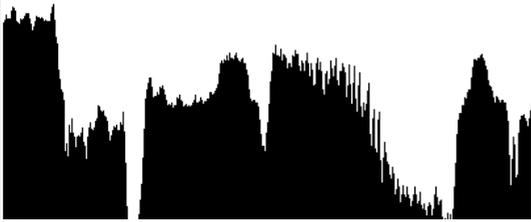


Рисунок 1.5 – Результат совместного искажения изображения 1.2 шумом «соль и перец» и «белым» шумом

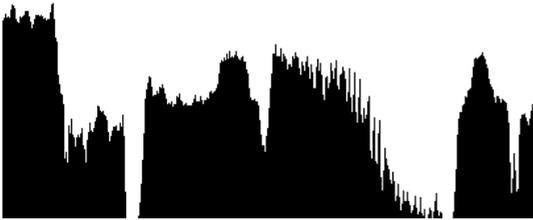


(a)



(b)

Рисунок 1.6 – Результаты обработки изображения, искаженного «белым» шумом (рисунок 1.3) среднеарифметическими фильтрами с апертурой 3×3 (а) и 9×9 (b)



(a)



(b)

Рисунок 1.7 – Результаты обработки изображения, искаженного «белым» шумом (см. рис. 1.3) среднегеометрическими фильтрами с апертурой 3×3 (a) и 9×9 (b)

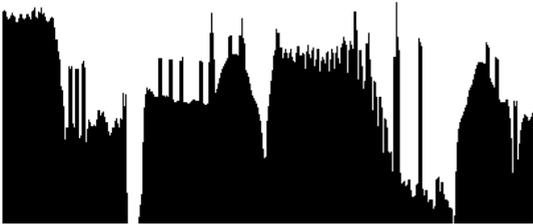


(a)

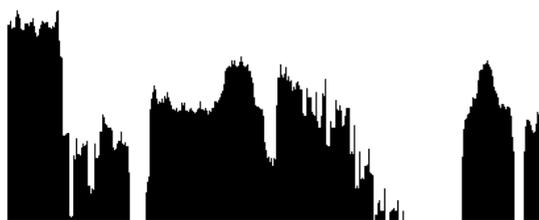


(b)

Рисунок 1.8 – Результаты обработки изображений, приведенных на рисунках 1.3 («белый» шум) и 1.4 (шум «соль и перец»), среднегармоническими фильтрами с апертурой и 9×9 (a) и 3×3 (b) соответственно



(a)

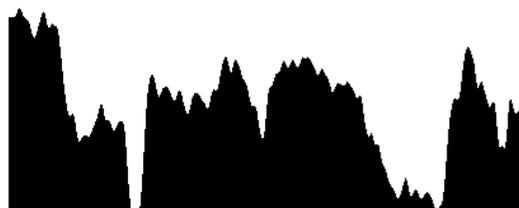


(b)

Рисунок 1.9 – Результаты обработки изображений, приведенных на рисунках 1.4 (шум «соль и перец») и 1.3 («белый» шум), контргармоническими фильтрами с апертурой 3×3 , $Q=2$ (a) и $Q=-5$ (b) соответственно

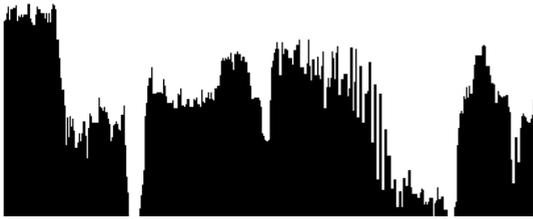


(a)

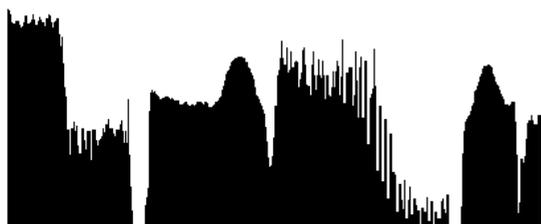


(b)

Рисунок 1.10 – Результаты обработки гауссианом с апертурой 9×9 изображений, приведенных на рисунках 1.3 («белый» шум) и 1.5 (шумы «соль и перец» и «белый»), (a) и (b) соответственно



(a)



(b)

Рисунок 1.11 – Результаты обработки медианным фильтром с апертурой 3×3 изображений, приведенных на рисунках 1.3 («белый» шум) и 1.4 (шум «соль и перец»), (a) и (b) соответственно



(a)



(b)

Рисунок 1.12 – Результат обработки бинаризованного изображения (а) последовательно минимальным и максимальным фильтрами с апертурой 3×3 (b)



(a)



(b)

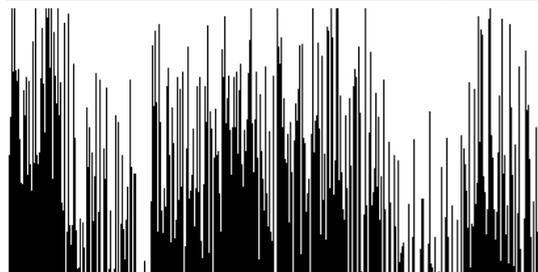


(c)

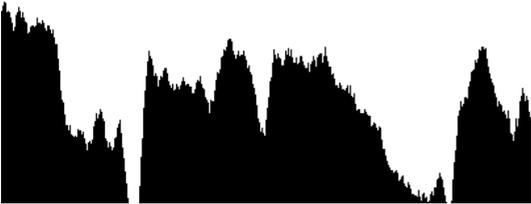
Рисунок 1.13 – Результаты обработки изображения 1.2, зашумленного белым шумом с амплитудой ± 100 (a), среднеарифметическим фильтром (b) и фильтром срединной точки (c) с апертурой 7×7 каждый

Выше Вариант 1 рисунка 1.13 исходное изображение 1.2 с затемнением 50%

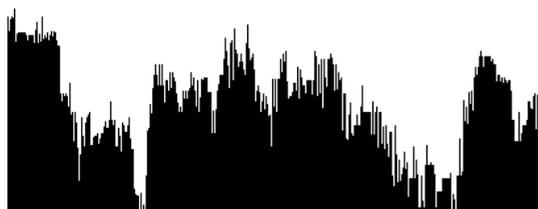
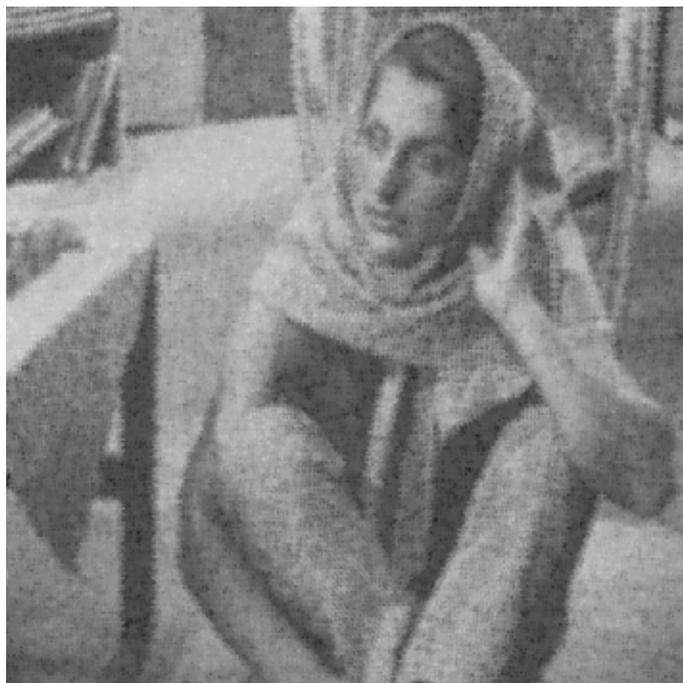
Ниже Вариант 2 рисунка 1.13 исходное изображение 1.2 без затемнения



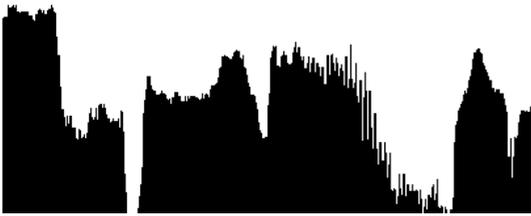
(a)



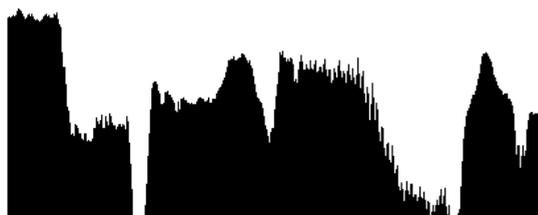
(b)



(c)

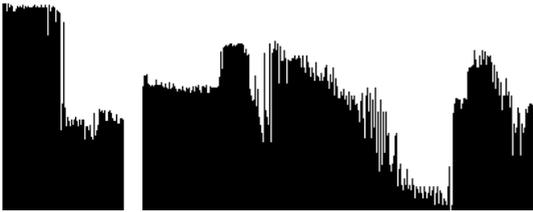


(a)



(b)

Рисунок 1.14 – Результаты обработки изображения 1.5 медианным фильтром с апертурой 5×5 (a) и фильтром усеченного среднего с апертурой 5×5 и величиной отсечения $d = 2$ (b)



(a)



(b)

Рисунок 1.15 – Результаты обработки фильтром Кувахары с апертурой 11×11 изображения 1.2 (a) и изображения 1.5 (b)

Таблица 1

Вход	Фильтры						
	положит. импульс	средне-арифметич	средне-геометрич	средне-гармонич	контргарм q=2	контргарм q=-2	гауссиан 1,2,4,2,1
10	10	10	10	10	10	10	10
10	50	18	12	208	10	30	30
10	50	18	12	208	10	50	50
210	50	18	12	208	10	90	90
10	50	18	12	208	10	50	50
10	50	18	12	208	10	30	30
10	10	10	10	10	10	10	10

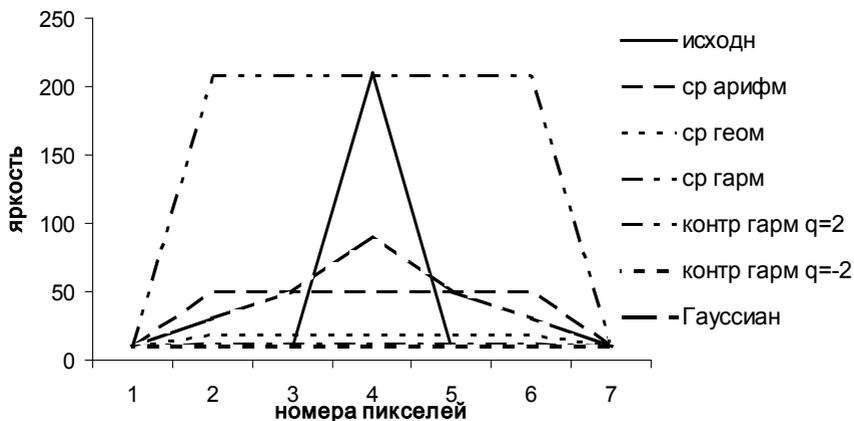


Рисунок 1.16 – Положительный импульс и результаты его обработки усредняющими фильтрами

Таблица 2

Вход	Фильтры						
	отрицат. импульс	средне-арифметич	средне-геометрич	средне-гармонич	контргарм q=2	контргарм q=-2	гауссиан 1,2,4,2,1
210	210	210	210	210	210	210	210
210	170	114	42	210	12	190	190
210	170	114	42	210	12	170	170
10	170	114	42	210	12	130	130
210	170	114	42	210	12	170	170
210	170	114	42	210	12	190	190
210	210	210	210	210	210	210	210

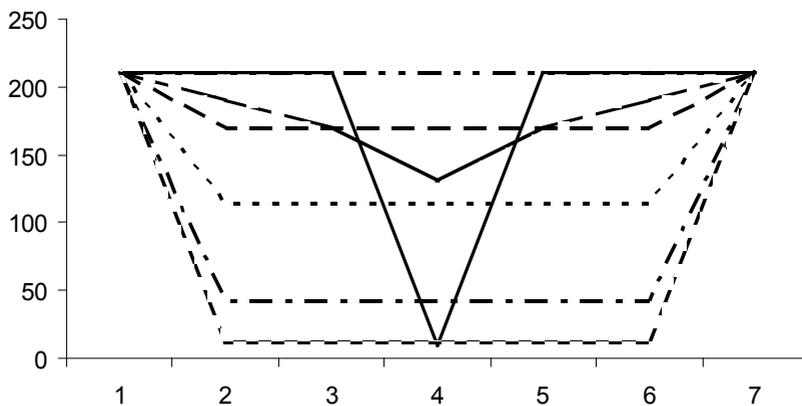


Рисунок 1.17 – Отрицательный импульс и результаты его обработки усредняющими фильтрами

Таблица 3

Вход	Фильтры						
	резкое изменен.	средне-арифметич	средне-геометрич	средне-гармонич	контргарм q=2	контргарм q=-2	гауссиан 1,2,4,2,1
10	10	10	10	10	10	10	10
10	48	18	12	12	198	10	29
10	86	33	16	16	199	10	67
200	124	60	23	23	200	11	143
200	162	110	42	42	200	12	181
200	200	200	200	200	200	200	200

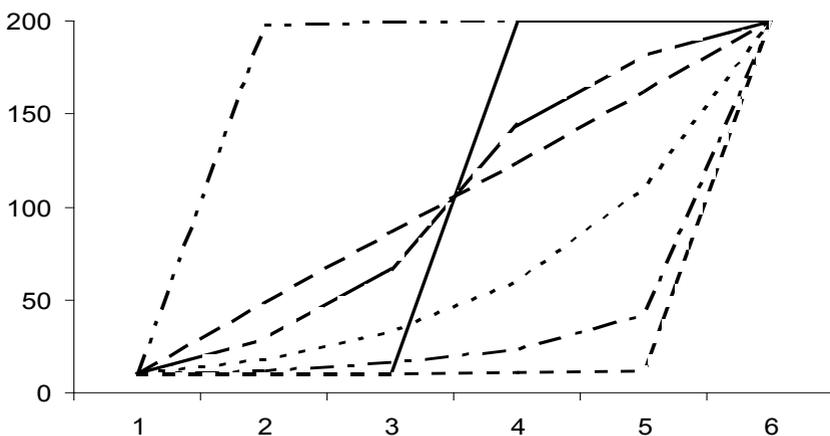


Рисунок 1.18 – Резкое изменение яркости и результаты его обработки усредняющими фильтрами

Таблица 4

Вход	Фильтры					
	средне-арифметич	средне-геометрич	средне-гармонич	контргарм q=2	контргарм q=-2	гауссиан 1,2,4,2,1
20	20	20	20	20	20	20
20	30	26	23	58	21	25
20	50	37	29	102	22	40
70	80	56	39	143	24	75
120	120	91	60	182	29	120
170	160	147	133	195	108	165
220	190	185	180	205	168	200
220	210	209	208	214	205	215
220	220	220	220	220	220	220

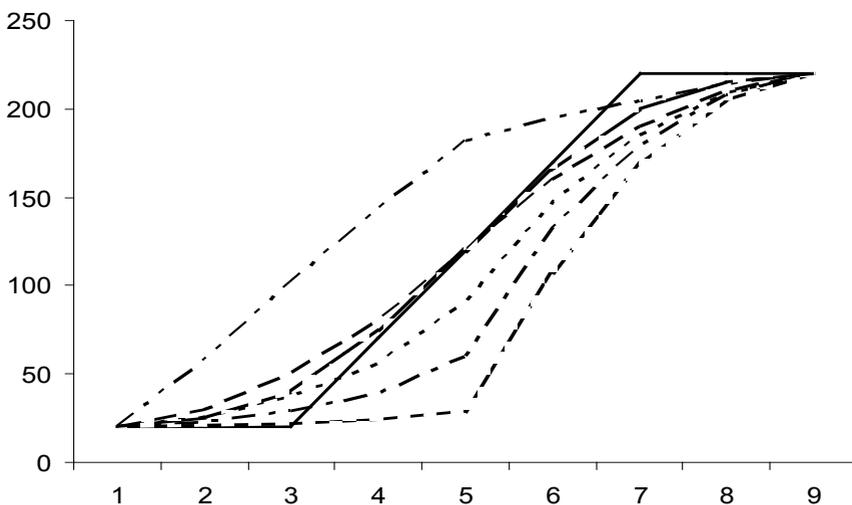


Рисунок 1.19 – Плавное изменение яркости и результаты его обработки усредняющими фильтрами

Таблица 5

Вход	Фильтры					
	средне-арифметич	средне-геометрич	средне-гармонич	контргарм q=2	контргарм q=-2	гауссиан 1,2,4,2,1
120	110	109	109	111	110	108
114	107	106	105	109	108	104
100	100	99	98	105	101	95
86	93	92	92	97	94	90
80	90	90	90	92	91	89
86	93	92	92	97	94	90
100	100	99	98	105	101	95
114	107	106	105	109	108	104
120	110	109	109	111	110	108

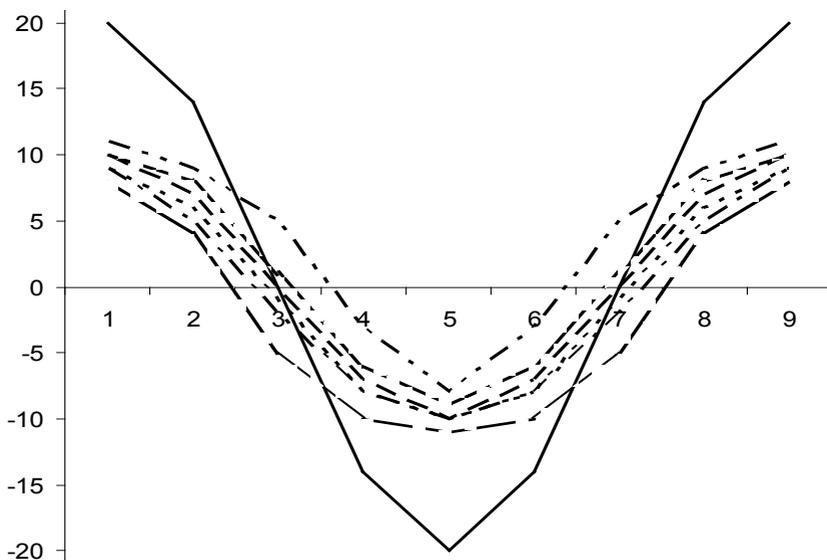


Рисунок 1.20 – Периодическое изменение яркости и результаты его обработки усредняющими фильтрами (яркость уменьшена на 100)

Таблица 6

Вход	Фильтры					
	положит. импульс	медианный n=7	минимальн. n= 5	максимальн. n= 5	средин.точ. n= 5	усечен сред n=7 d=2
10	10	10	10	10	10	10
10	10	10	10	210	110	10
10	10	10	10	210	110	10
210	10	10	10	210	110	10
10	10	10	10	210	110	10
10	10	10	10	210	110	10
10	10	10	10	10	10	10

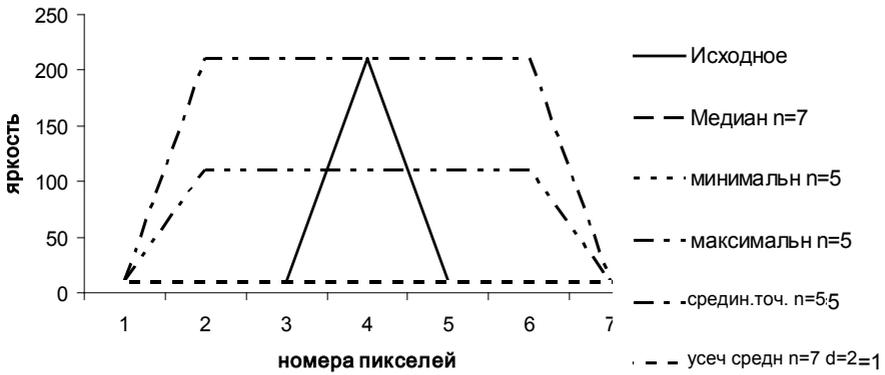


Рисунок 1.21 – Положительный импульс и результаты его обработки упорядочивающими фильтрами

Таблица 7

Вход	Фильтры				
	медианный n=7	минималн. n= 5	максималн. n= 5	средин.точ. n= 5	усечен сред n=7 d=2
210	210	210	210	210	210
210	210	10	210	110	210
210	210	10	210	110	210
10	210	10	210	110	210
210	210	10	210	110	210
210	210	10	210	110	210
210	210	210	210	210	210

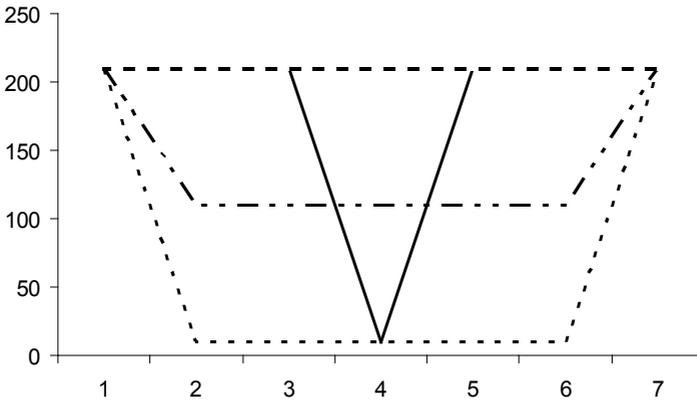


Рисунок 1.22 –Отрицательный импульс и результаты его обработки упорядочивающими фильтрами

Таблица 8

Вход резкое изменен	Фильтры				
	медианный n=7	минимальн. n= 5	максимальн. n= 5	средин.точ. n= 5	усечен сред n=7 d=2
10	10	10	10	10	10
10	10	10	200	105	48
10	10	10	200	105	86
200	200	10	200	105	124
200	200	10	200	105	162
200	200	200	200	200	200
200	200	200	200	200	200

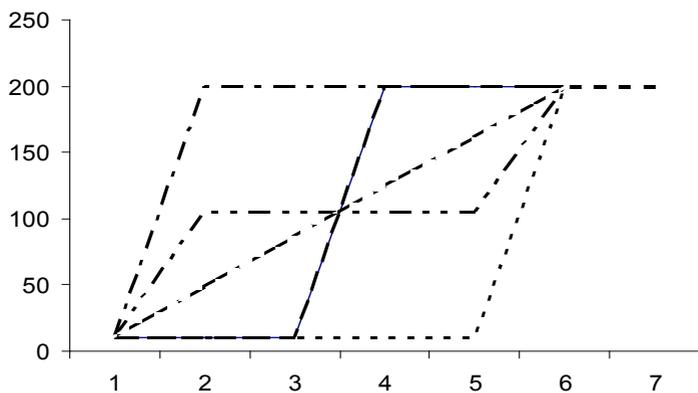


Рисунок 1.23 –Резкое изменение яркости и результаты его обработки упорядочивающими фильтрами

Таблица 9

Вход	Фильтры				
	медианный n=7	минималн. n= 5	максималн. n= 5	средин.точ. n= 5	усечен сред n=7 d=2
20	20	20	20	20	20
20	20	20	70	45	30
20	20	20	120	70	50
70	70	20	170	95	80
120	120	20	220	120	120
170	170	70	220	145	160
220	220	120	220	170	190
220	220	170	220	195	210
220	220	220	220	220	220

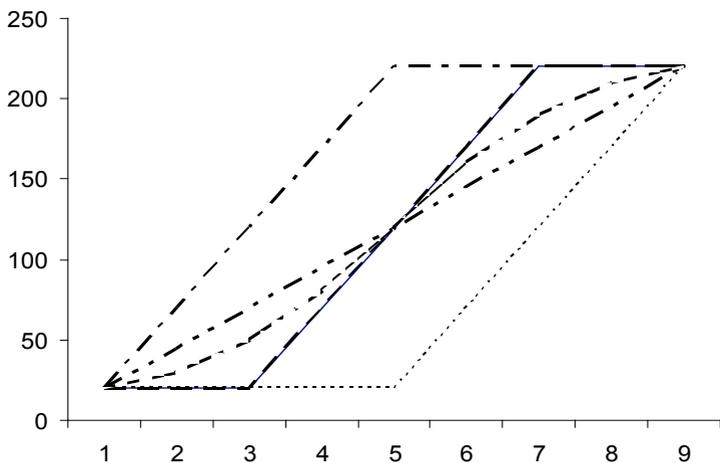


Рисунок 1.24 – Плавное изменение и результаты его обработки упорядочивающими фильтрами

Таблица 10

Вход	Фильтры				
	медианный n=7	минимальн. n= 5	максимальн. n= 5	средин.точ. n= 5	усечен сред n=7 d=2
120	100	100	120	110	103
114	100	86	120	103	103
100	100	80	120	100	100
86	100	80	114	97	97
80	100	80	100	90	97
86	100	80	114	97	97
100	100	80	120	100	100
114	100	86	120	103	103
120	100	100	120	110	103

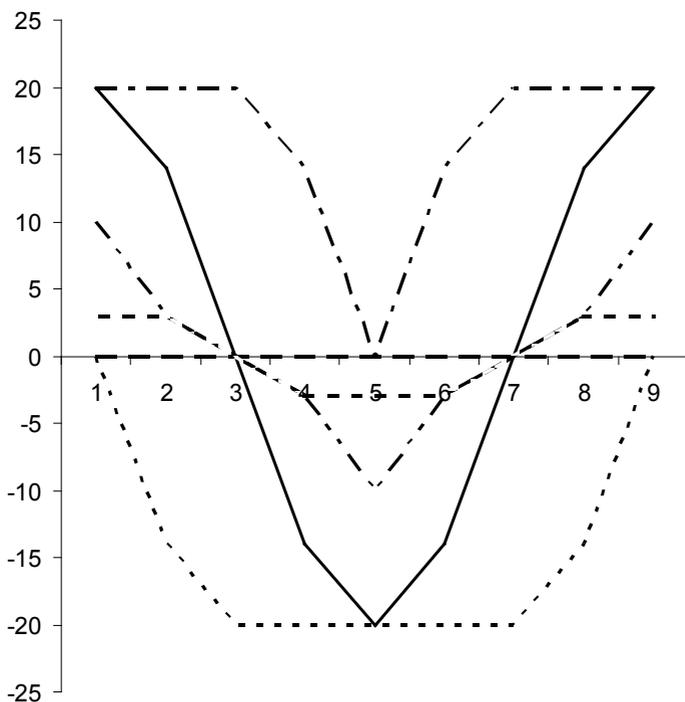
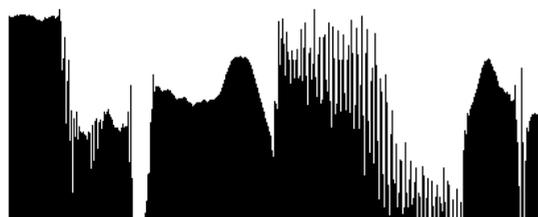


Рисунок 1.25 – Периодическое изменение яркости и результаты его обработки упорядочивающими фильтрами (яркость уменьшена на 100)

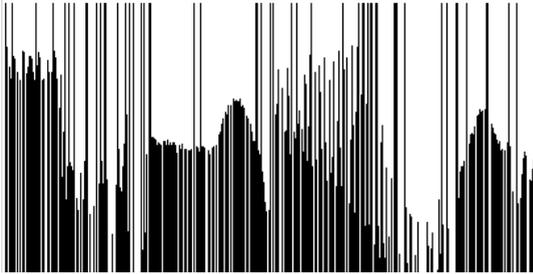


(a)



(b)

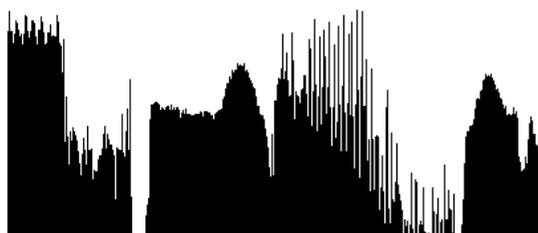
Рисунок 1.26 – Результаты обработки изображения 1.3 среднеарифметическим (а) и адаптивным усредняющим (b) фильтрами с апертурой 7×7



(a)

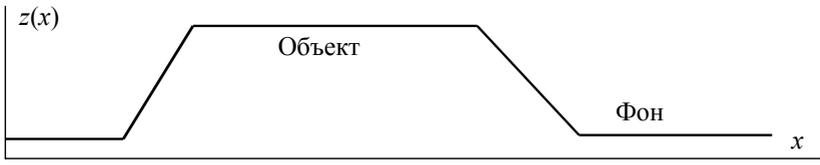


(b)

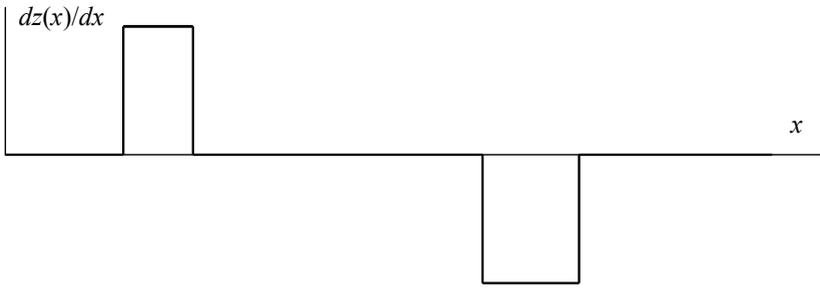


(с)

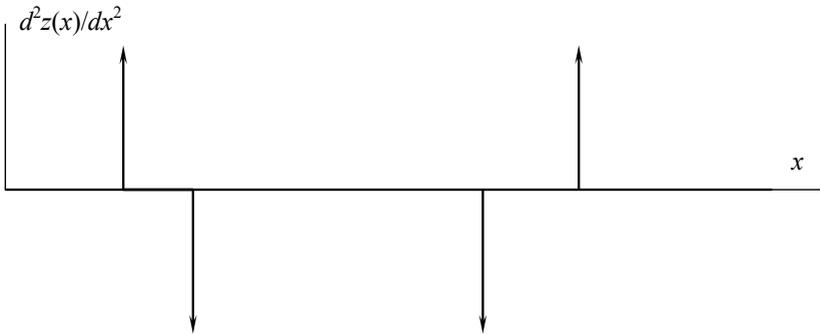
Рисунок 1.27 – Изображение, искаженное шумом «соль и перец» с частотой 40% (а), и результаты его обработки медианными фильтрами с апертурой 7×7 : обычным (б) и адаптивным (с)



(a)



(b)



(c)

Рисунок 2.1 –Разрез функции яркости модели изображения объект-фон (a) и его первая (b) и вторая (c) производные



(a)



(b)



(c)

Рисунок 2.2 – Результаты обработки изображения, приведенного на рисунке 1.2, фильтрами Робертса (a), Собеля (b) и Превитт (c). Изображения инвертированы.

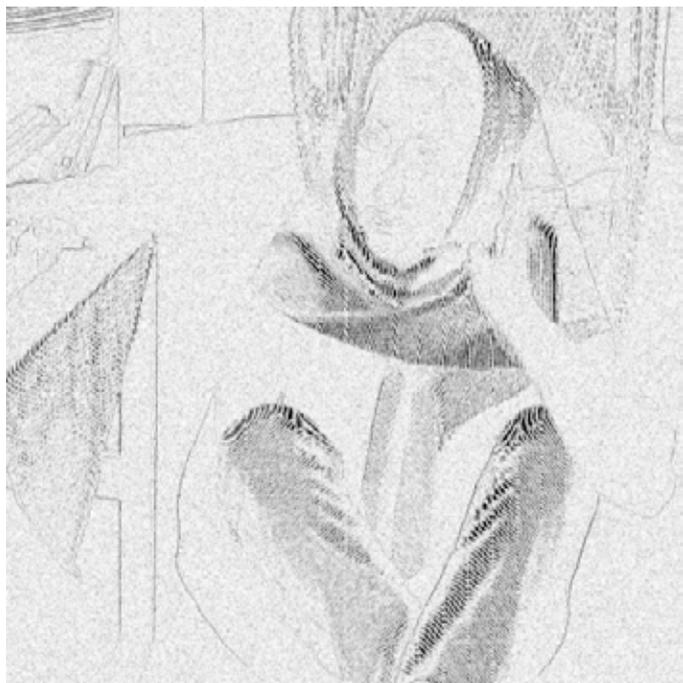


(a)



(b)

Рисунок 2.3 – Результаты обработки изображения, приведенного на рисунке 1.2, фильтрами Собеля (а) и Превитт (b). Изображения инвертированы и нормированы по коэффициенту усиления и яркости.



(a)



(b)



(с)

Рисунок 2.4 – Результаты обработки изображения, приведенного на рисунке 1.3 («белый» шум), фильтрами Робертса (а), Собеля (б) и Превитт (с). Изображения инвертированы и нормированы по коэффициенту усиления и яркости.



(a)



(b)

Рисунок 2.5 – Результаты обработки изображения, приведенного на рисунке 1.2, фильтрами Лапласа: обычным (а) и согласованным (b). Изображения инвертированы.

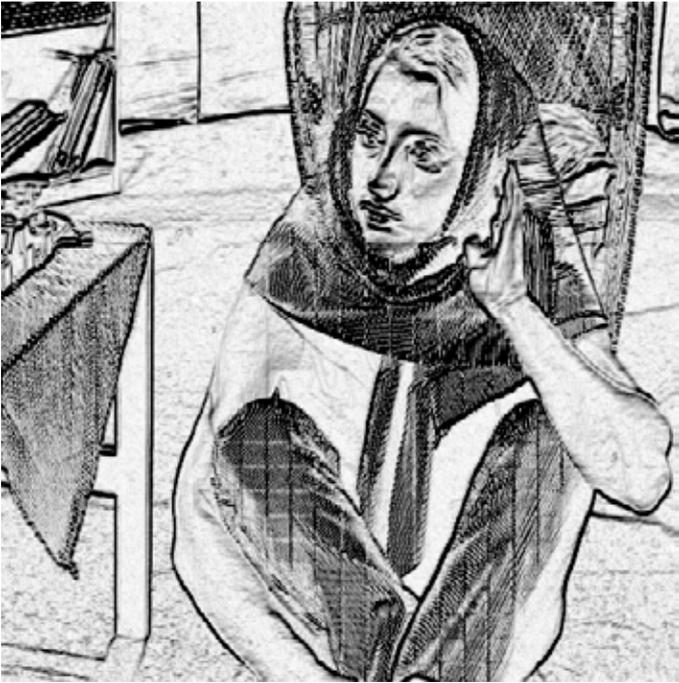


(a)



(b)

Рисунок 2.6 – Результаты обработки изображения, приведенного на рисунке 1.3 («белый» шум), фильтрами Лапласа: обычным (а) и согласованным (b). Изображения инвертированы.



(a)



(b)

Рисунок 2.7 – Результаты обработки фильтром Кирша изображений, приведенных на рисунке 1.2 (а) и 1.3 (b). Изображения инвертированы.



(a)



(b)

Рисунок 2.8 – Результаты обработки фильтром Уоллеса ($C_{norm} = 100$) изображений, приведенных на рисунке 1.2 (a) и 1.3 (b). Изображения инвертированы.



(a)



(b)

Рисунок 2.9 – Результаты обработки LoG-фильтром с маской 11×11 изображений, приведенных на рисунках 1.2 (a) и 1.3 (b). Изображения инвертированы и нормированы по коэффициенту усиления и яркости.

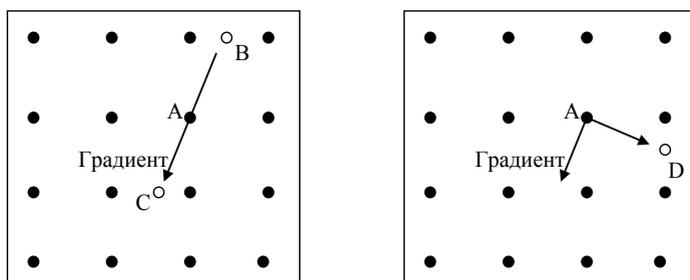


Рис.2.10 – Нахождение линии контура методом неаимаксимального подавления



(a)



(b)

Рисунок 2.11 – Результаты применения метода не максимального подавления изображения, приведенного на рисунке 1.2, без предварительного размывания (а) и с размыванием фильтром Гаусса и установкой порога отсека градиента яркости (b). Изображения инвертированы

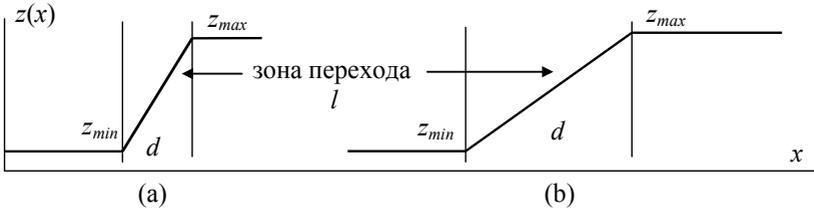


Рисунок 3.1 –Разрезы функции яркости моделей изображений объект-фон (а) и (b) с одинаковой контрастностью и разной резкостью

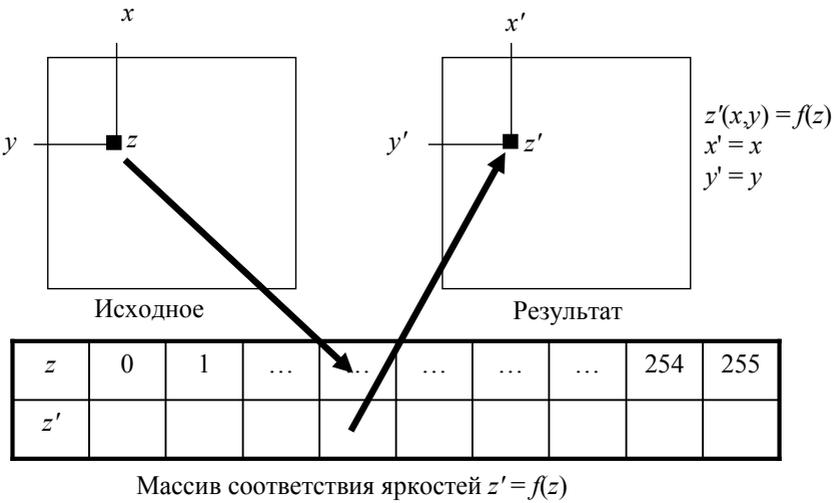
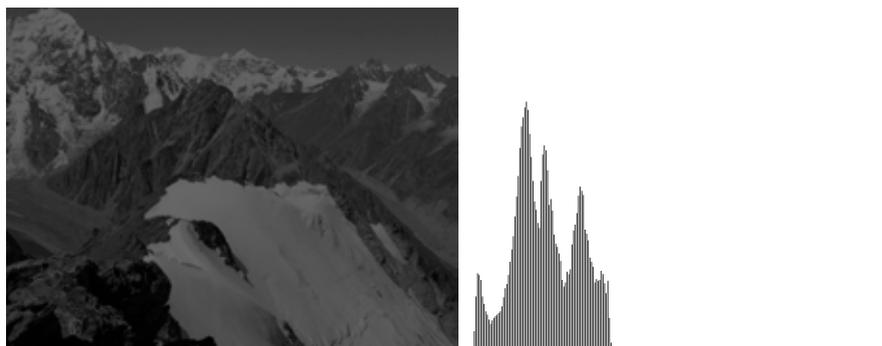
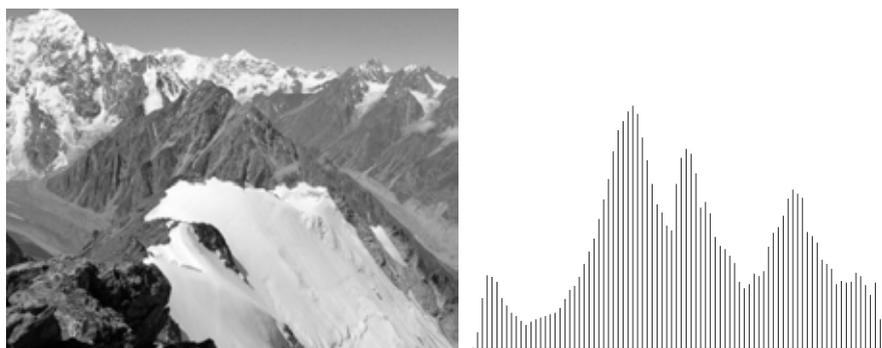


Рисунок 3.2 – Алгоритм обработки изображения преобразованием шкалы яркости



(a)

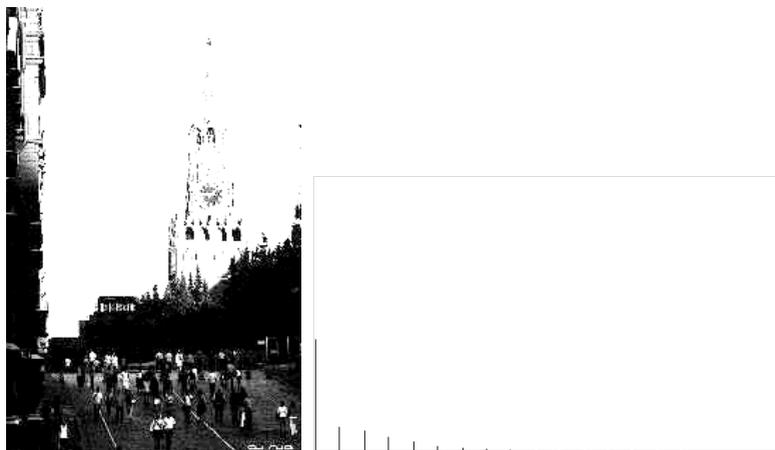


(b)

Рисунок 3.3 – Исходное изображение (a) с контрастностью $C=85$ и результат его обработки (b) линейным растяжением шкалы яркости с гистограммами яркости



(a)



(b)

Рисунок 3.4 – Исходное изображение (a) с контрастностью $C=255$ и результат его обработки (b) кусочно-линейным преобразованием шкалы яркости в диапазоне 0 – 20 с гистограммами яркости

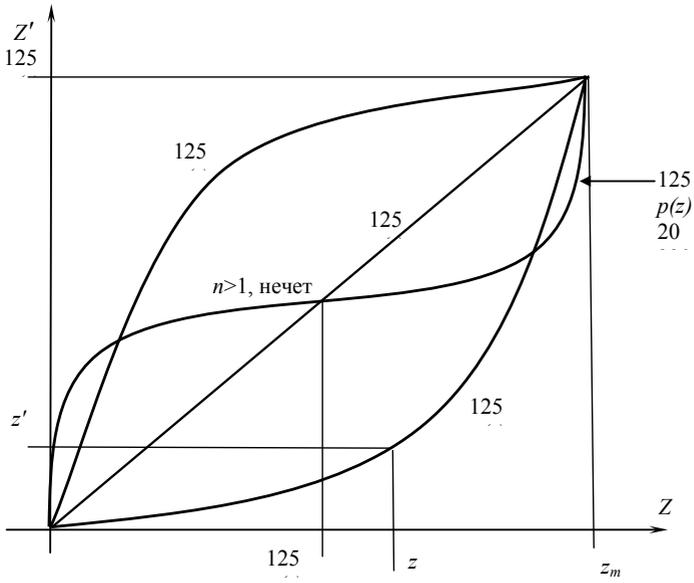
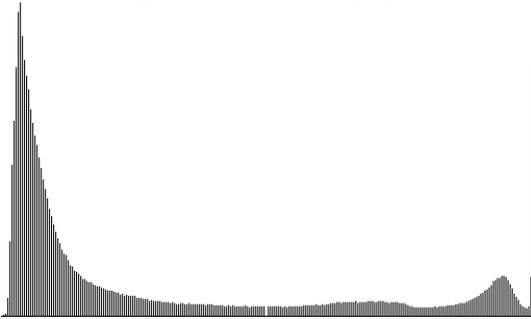


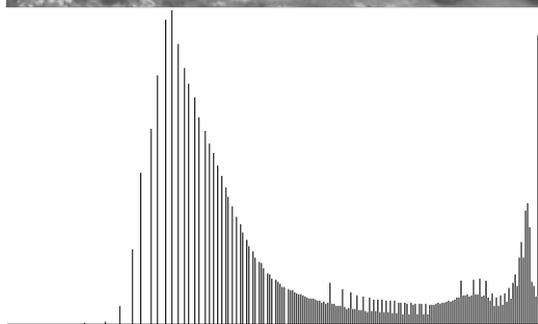
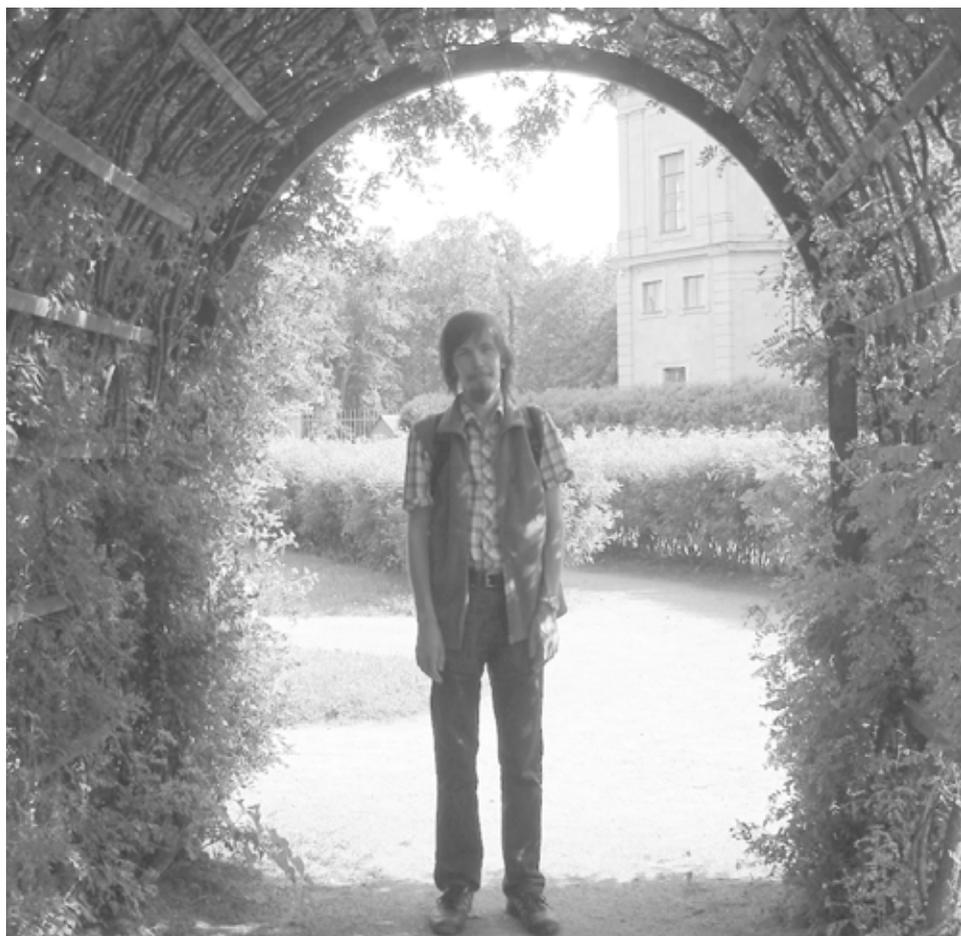
Рис. 3.5 – Степенное преобразование шкалы яркости (гамма-коррекция)



(a)



(b)



(c)

Рисунок 3.6 – Исходное изображение (а) и результаты его обработки гамма-коррекцией с $\gamma > 1$ (b) и $\gamma < 1$ (c) с гистограммами яркости

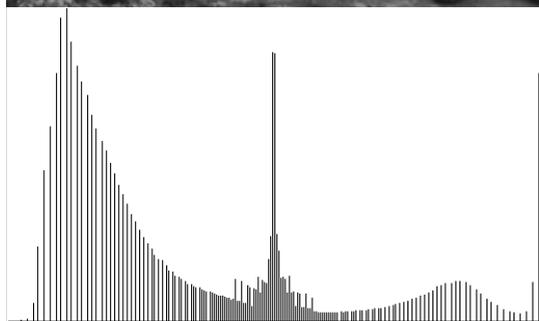
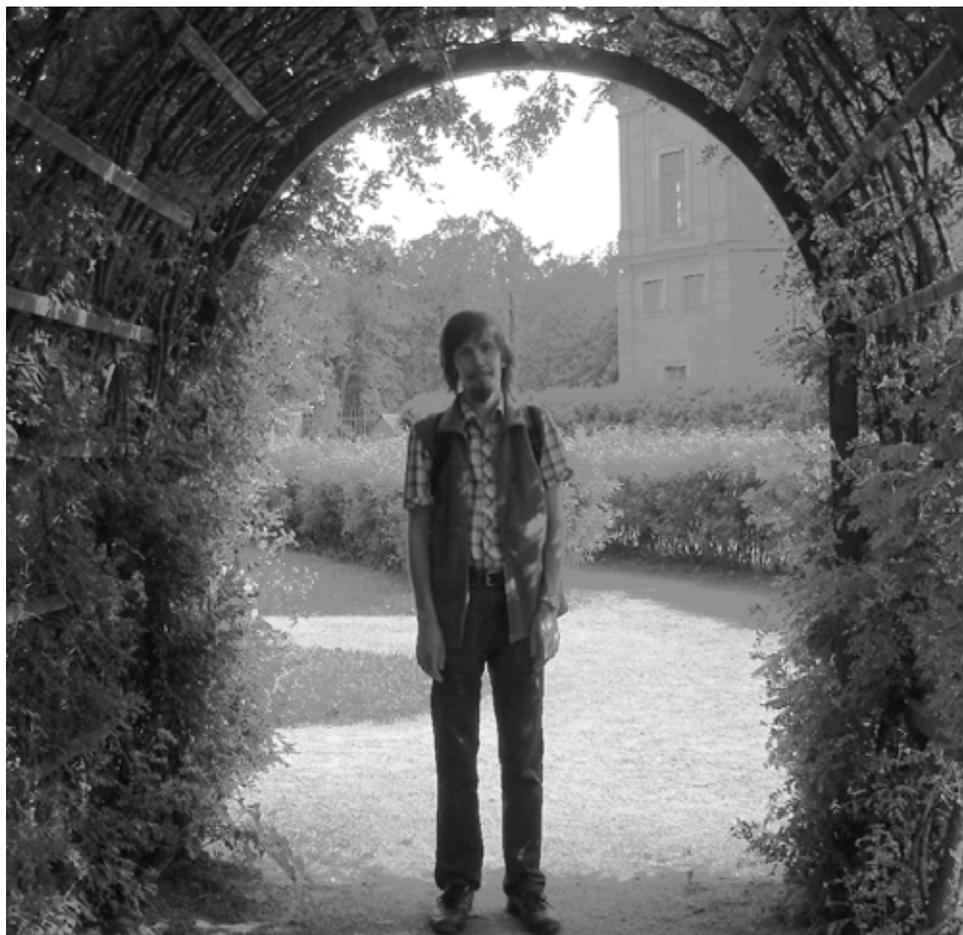


Рисунок 3.7 – Результат обработки изображения 3.6 (а) гамма-коррекцией со смещенным нулем и гистограмма яркости

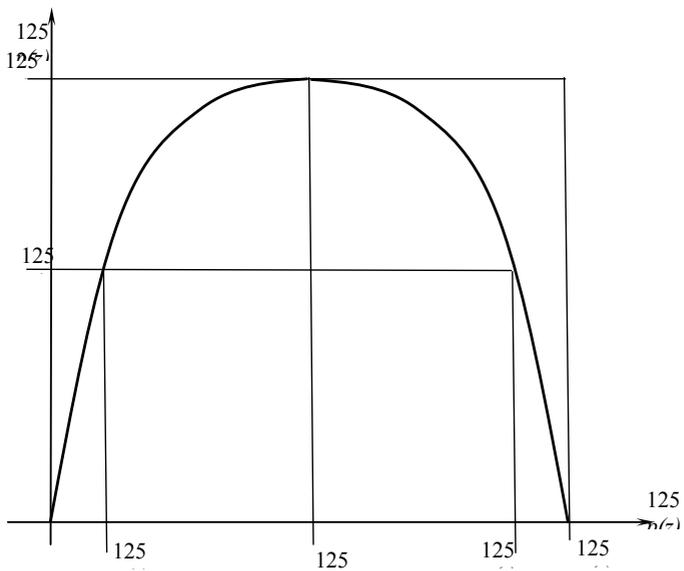


Рисунок 3.8 – График функции соляризации

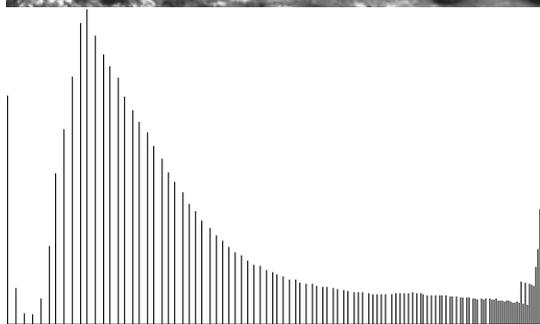
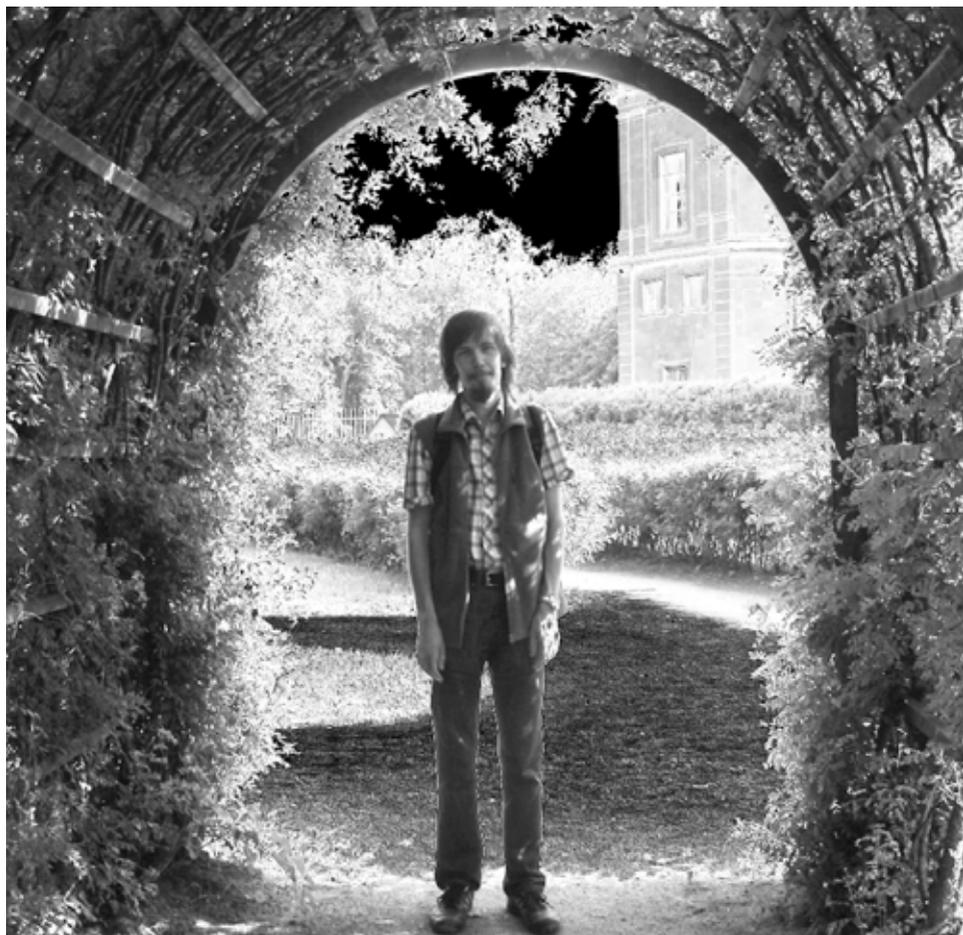
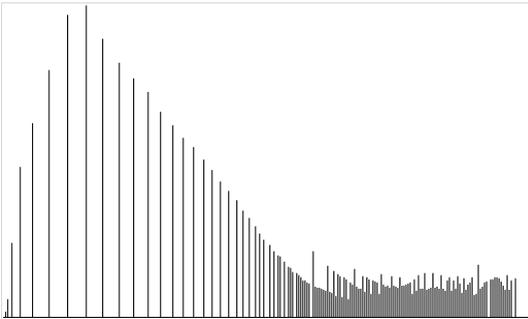
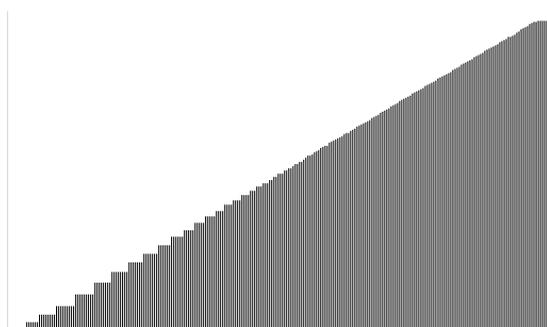
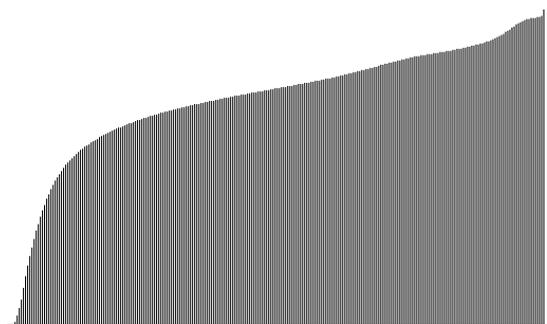


Рисунок 3.9 – Результат обработки изображения 3.6 (а) соляризацией и гистограммой яркости

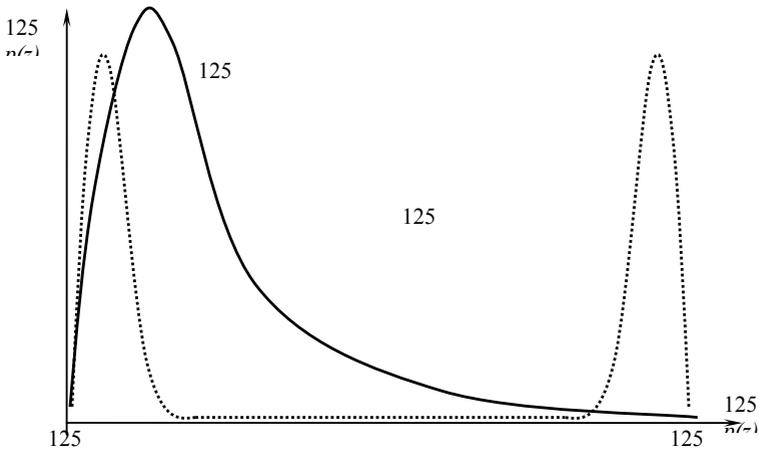


(a)



(b)

Рисунок 3.10 – Результат эквализации гистограммы яркости изображения 3.6 (а) и гистограмма яркости (а), а также кумулятивные гистограммы исходного и преобразованного изображений (b).



(a)

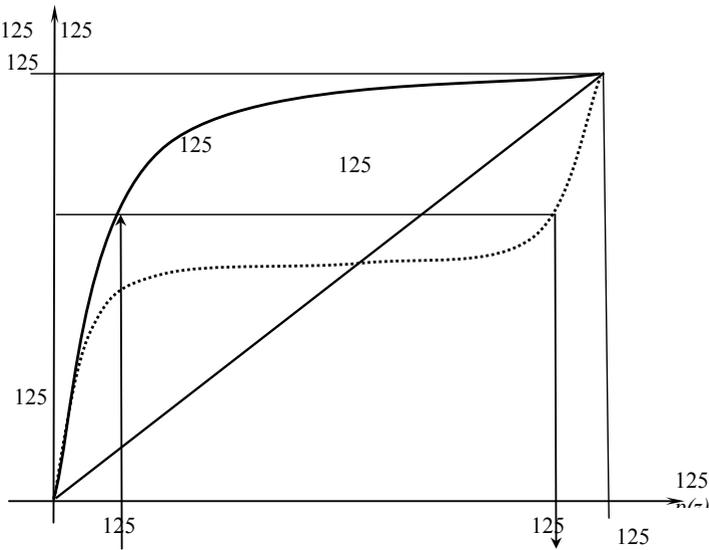
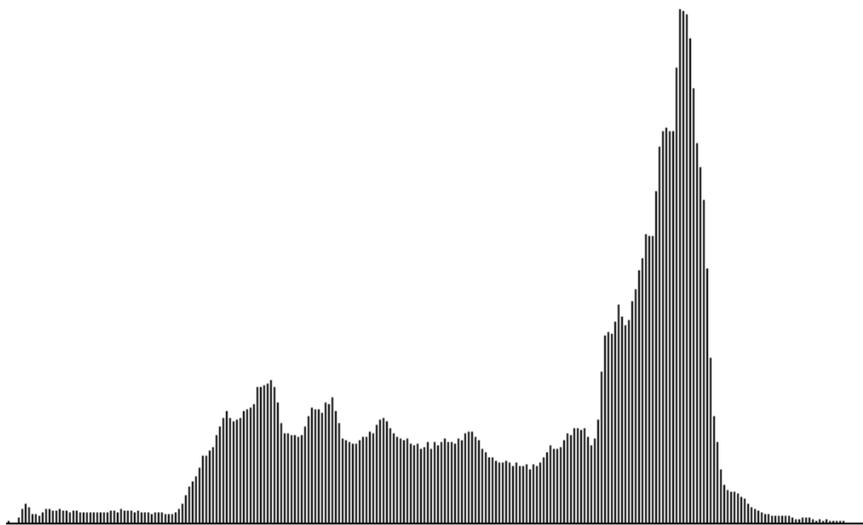
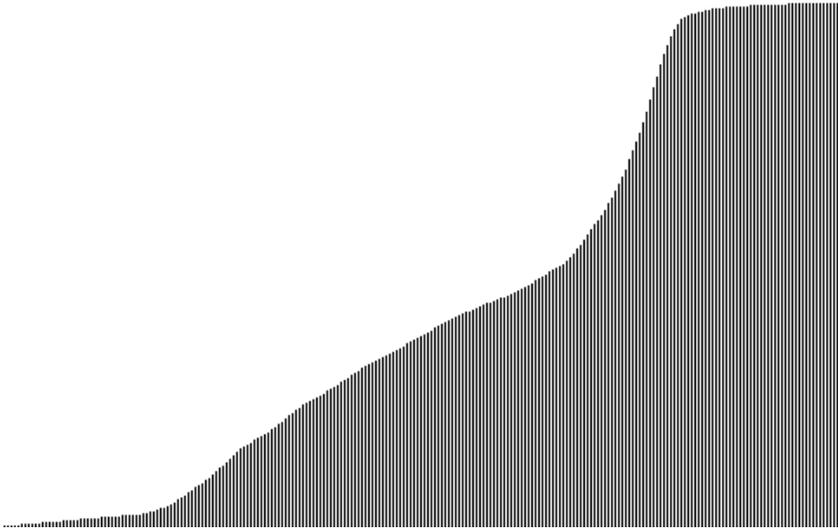
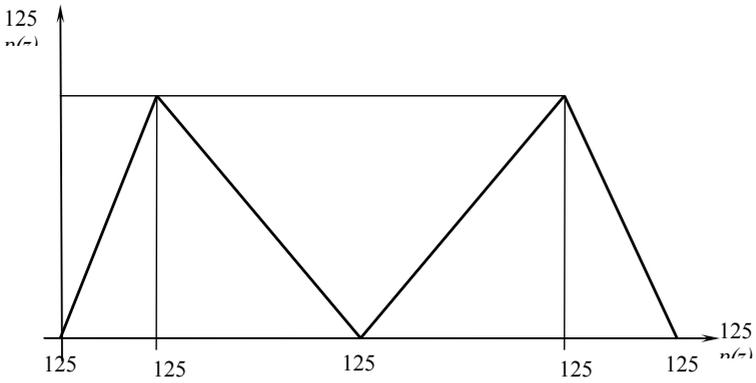


Рисунок 3.11 – Исходное изображение и желаемая гистограммы яркости (а), соответствующие им кумулятивные гистограммы (b)

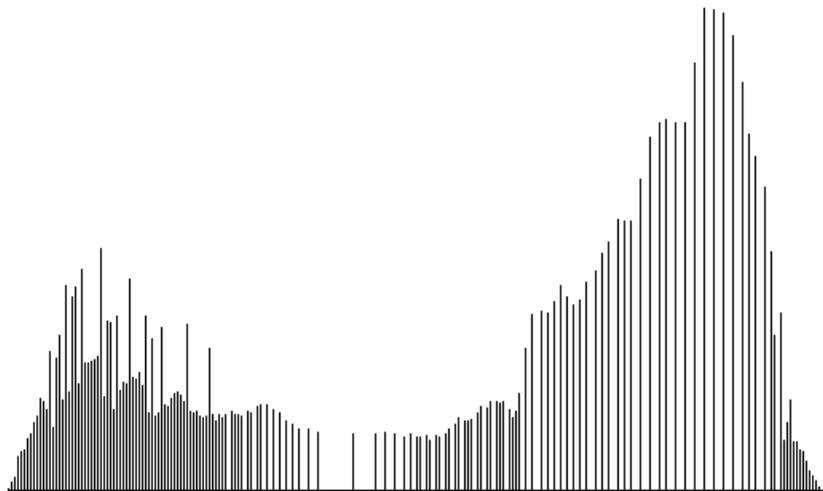


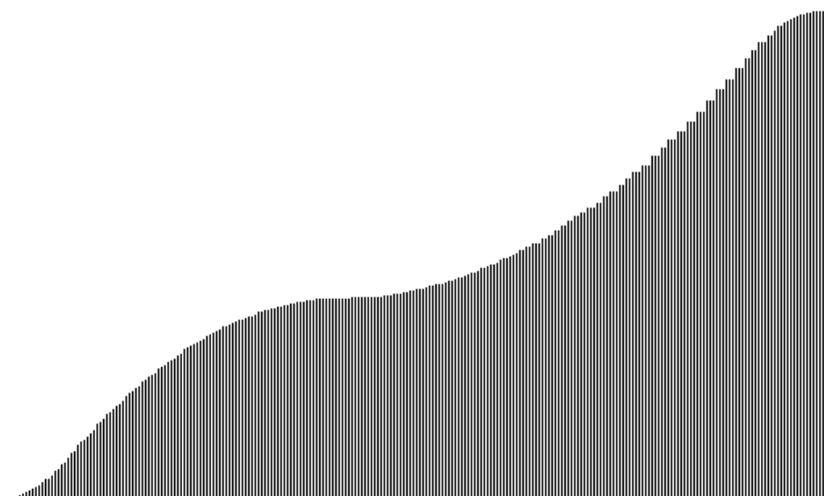


(a)



(b)





(c)

Рисунок 3.12 – Исходное изображение (а), график желаемой гистограммы яркости (b) и результат обработки (c) с гистограммами яркости и кумулятивными гистограммами.



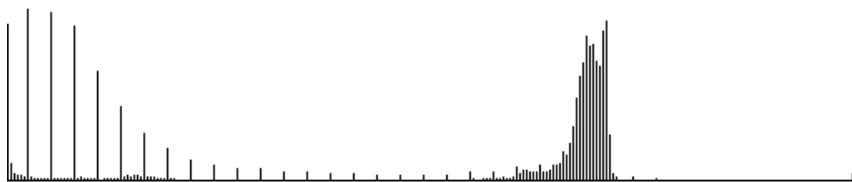


Рисунок 3.13 – Результат обработки изображения 3.4 (а) методом адаптивного увеличения контрастности. Размер окрестности – 5×5 пикселей, значения параметров: $k_0 = 7.0$, $k_1 = 0.6$, $k_2 = 0.02$, $k_3 = 0.6$.



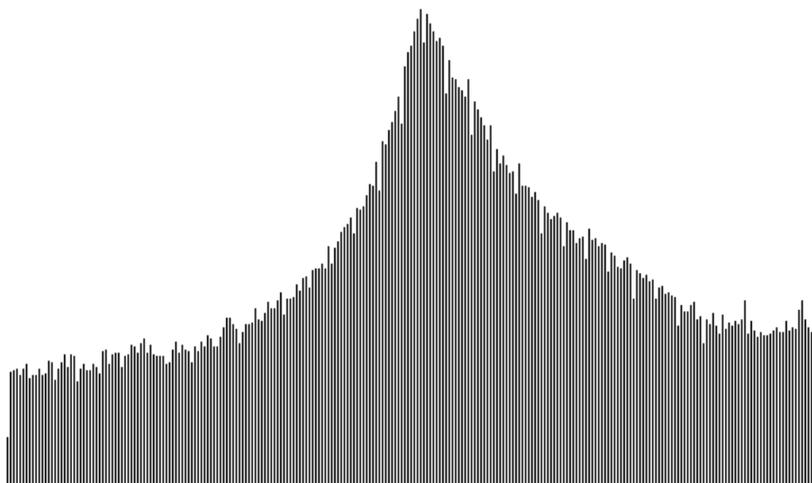


Рисунок 3.14 –Результат локального увеличения контрастности эквализацией гистограммы яркости изображения 3.12 (а) с гистограммой яркости. Размер окрестности – 55×55 пикселей.



(а)



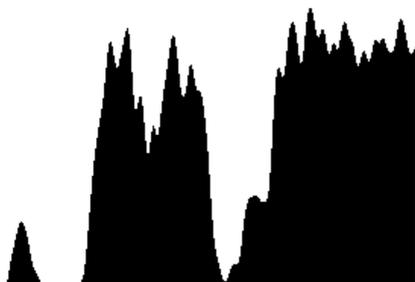
(b)



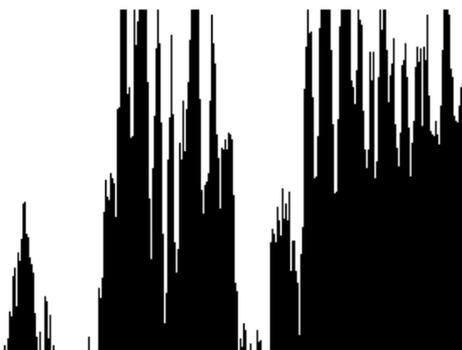
(c)

Рисунок 3.15 – Исходное изображение (a) и результат компенсации освещения (b) с последующим кусочно-линейным преобразованием шкалы яркости (c). Усред-

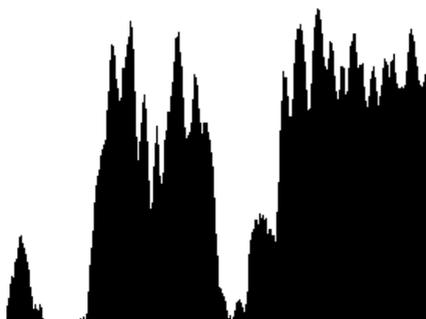
няющий фильтр при компенсации освещения – Гауссиан 11×11 пикселей, диапазон кусочно-линейного преобразования шкалы яркости 0 – 90.



(a)

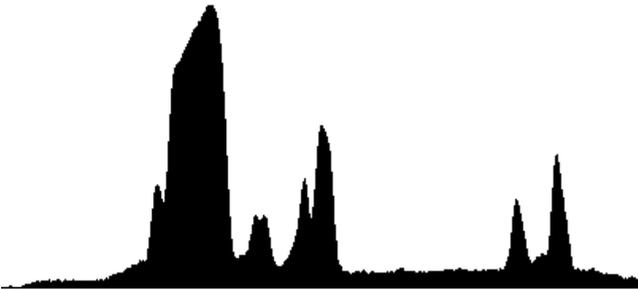


(b)



(c)

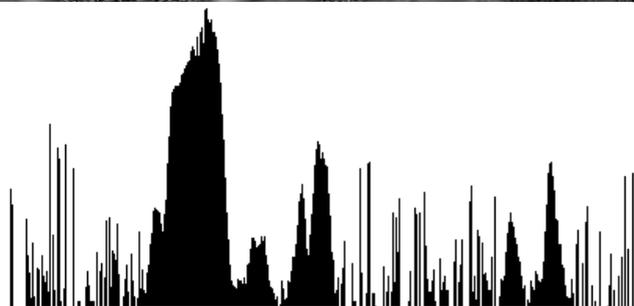
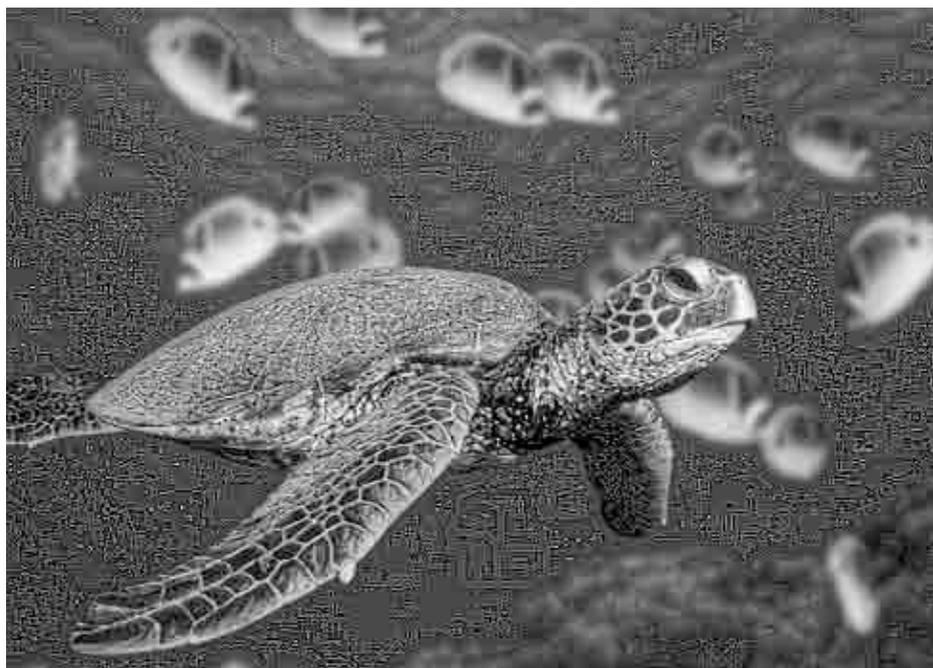
Рисунок 3.16 – Исходное размытое изображение (a) и результаты увеличения резкости с использованием фильтра Лапласа (b) и нерезким маскированием (c) с разрезами функции яркости по одной и той же строке (192). Коэффициенты усиления $k_L = 5$, $k_G = 5$



(a)



(b)



(с)

Рисунок 3.17 – Исходное изображение с разным уровнем размывания областей (а) и результат адаптивного усиления резкости с учетом дисперсии яркости окрестности (b – порог 5, с – порог 0) с разрезами функции яркости по одной и той же строке. Размер окрестности 7×7 , коэффициент 70.

Таблица 11

z	C_z	z'	R_z	z	C_z	z'	R_z
$\Delta = 4$				$\Delta = 10$			
98	0	98	0	95	0	95	0
98	0,004	87	2,76	95	0,01	79	1,6
98	0,008	83	3,71	95	0,021	74	2,09
102	0,008	120	4,5	105	0,019	133	2,87
102	0,004	115	3,19	105	0,01	125	2,04
102	0	102	0	105	0	105	0
$\Delta = 20$				$\Delta = 60$			
90	0	90	0	70	0	70	0
90	0,022	69	1,01	70	0,079	46	0,4
90	0,043	64	1,28	70	0,146	42	0,47
110	0,038	151	2,06	130	0,102	205	1,25
110	0,019	139	1,47	130	0,048	185	0,91
110	0	110	0	130	0	130	0

Таблица 12

z	z'				
	$n=0,8$	$n=0,6$	$n=0,5$	$n=0,3$	$n=0,2$
99	99	99	99	99	99
99	99	99	99	99	99
99	98	95	91	73	55
99	97	92	88	68	50
101	103	108	114	147	199
101	102	106	110	137	182
101	101	101	101	101	101
101	101	101	101	101	101

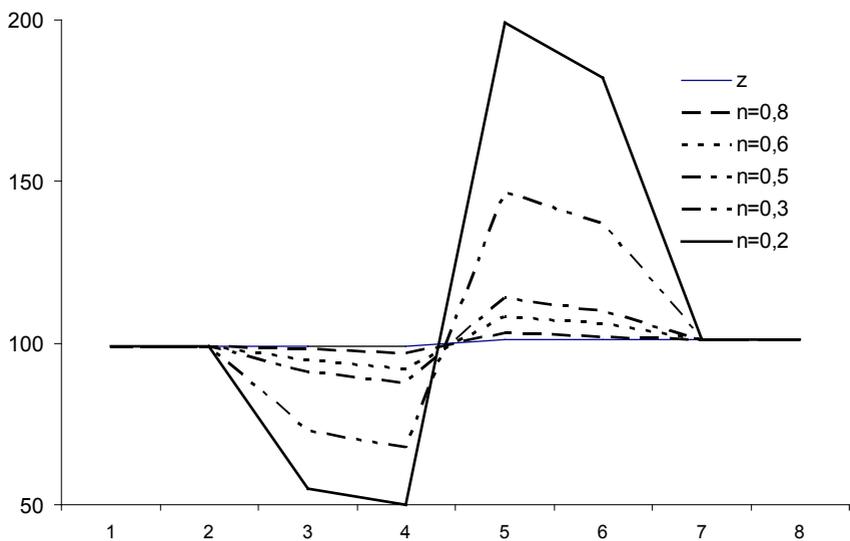


Рис. 3.18 – Результат моделирования адаптивного увеличения жесткости

Таблица 13

z	C_z	z'	R_z	z	C_z	z'	R_z
$\Delta = 4$				$\Delta = 12$			
98	0,001	92	1,47	94	0,006	81	1,07
98	0,003	88	2,43	96	0,002	88	0,67
99	0,001	93	1,48	98	0	98	0
100	0	100	0	100	0	100	0
101	0,001	107	1,59	102	0	102	0
102	0,003	113	2,76	104	0,002	113	0,76
102	0,001	108	1,59	106	0,006	121	1,33

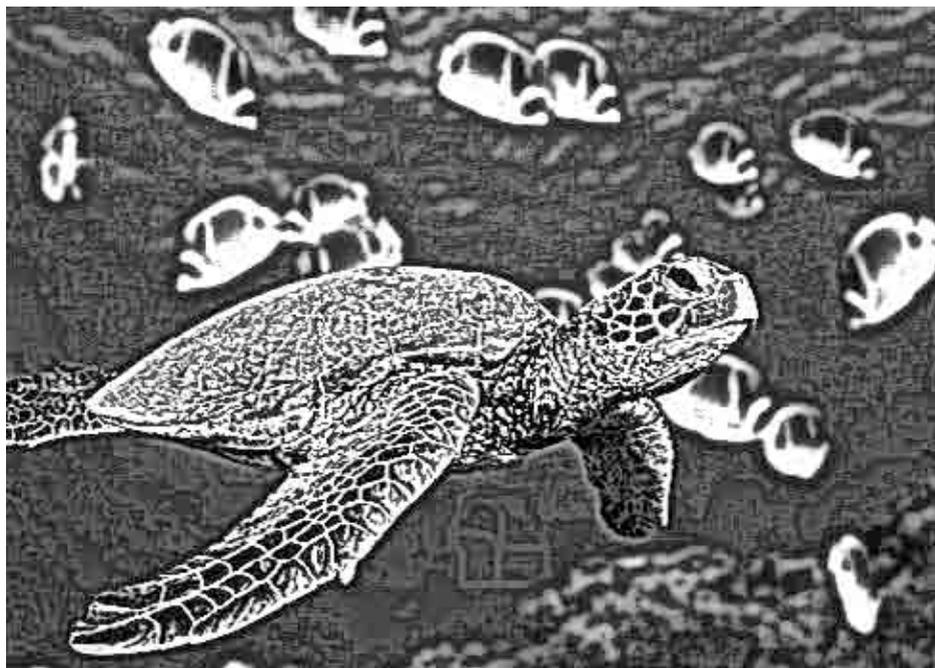
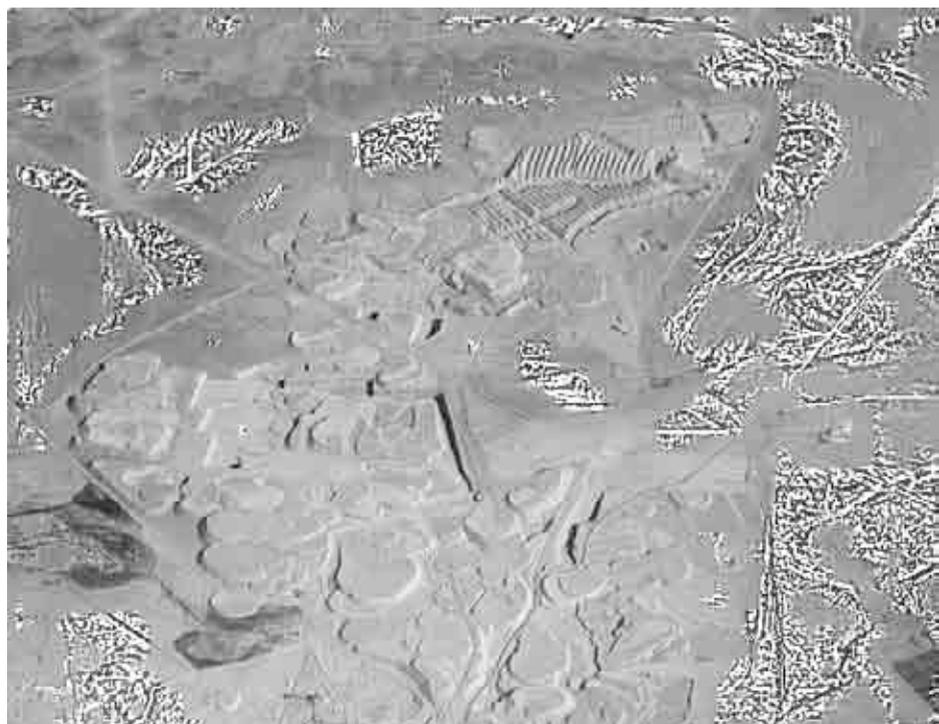


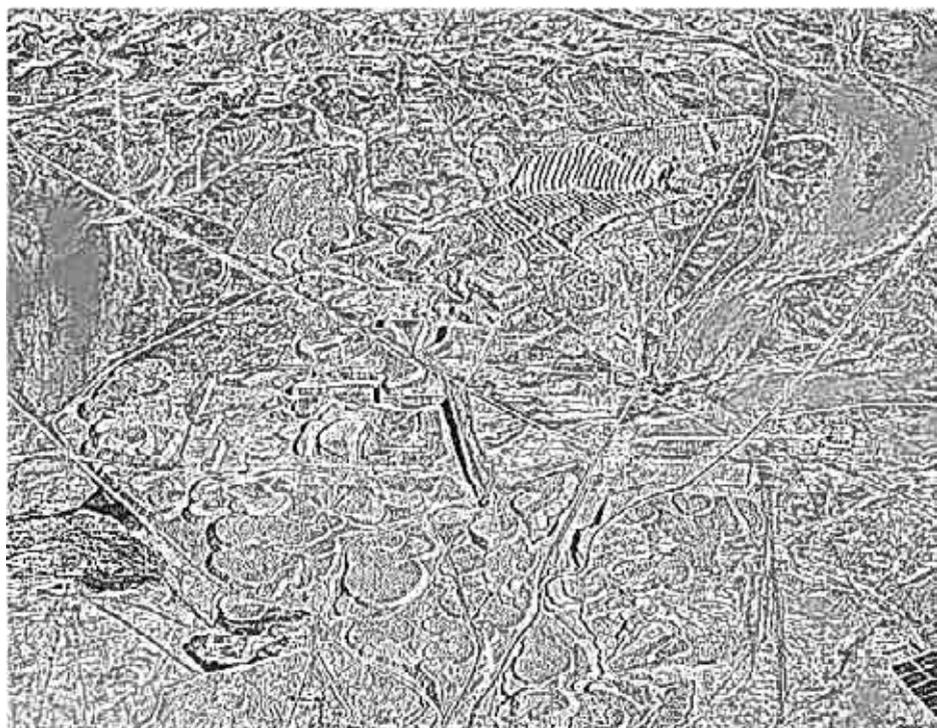
Рисунок 3.19 – Результат усиления локальной контрастности изображения 3.17 (а)



(a)



(b)



(c)



(d)

Рисунок 3.20 – Исходное изображение с мелкими малоразличимыми деталями (a) и результаты его обработки с использованием протяженности гистограммы (b), энтропии (c) и среднеквадратичного отклонения (d) яркости для нахождения значения адаптивной степени усиления локальной контрастности

Учебное пособие

Соловьев Николай Владимирович

Сергеев Александр Михайлович

Улучшение качества растровых изображений

Редактор

Корректор

Верстальщик

Сдано в набор

Подписано к печати

Формат

Бумага офсетная. Усл. печ. л.

Уч.-изд. л.

Тираж экз. Заказ
