

Е. В. МЕДВЕДЕВА

**ЦИФРОВАЯ ОБРАБОТКА ИЗОБРАЖЕНИЙ
В ВИДЕОИНФОРМАЦИОННЫХ СИСТЕМАХ**

Учебное пособие

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	5
1. ПРЕДСТАВЛЕНИЕ ИЗОБРАЖЕНИЙ	6
1.1. Типы изображений	6
1.2. Цветовые системы	6
1.3. Сэмплирование	8
2. СЖАТИЕ ИЗОБРАЖЕНИЙ	9
2.1. Требования к алгоритмам сжатия изображений	9
2.2. Оценка качества изображений	11
2.2.1. Субъективная оценка качества	11
2.2.2. Объективная оценка качества	12
2.3. Алгоритмы сжатия изображений без потерь качества	13
2.3.1. Групповое кодирование	13
2.3.2. Алгоритм Лемпеля-Зива-Велча	14
2.3.3. Алгоритм Хаффмана	16
2.3.4. Арифметическое кодирование	17
2.4. Алгоритмы кодирования с преобразованием	19
2.4.1. Дискретное косинусное преобразование	19
2.4.2. Дискретное вейвлет-преобразование	20
2.5. Стандарты сжатия статических изображений	23
2.5.1. Алгоритм JPEG	23
2.5.2. Алгоритм JPEG 2000	25
2.6. Стандарты сжатия видео	26
2.6.1. MPEG-1	27
2.6.2. MPEG-2	28
2.6.3. MPEG-4	28
2.6.4. H.261	29
2.6.5. H.263	30
2.6.6. H.264	30
2.6.7. H.265	34
2. ФИЛЬТРАЦИЯ ИЗОБРАЖЕНИЙ	35
3.1. Модели шумов на изображениях	35
3.2. Типы фильтров для обработки изображений	38
3.3. Пространственная линейная фильтрация	39
3.3.1. Масочная фильтрация изображений	39
3.3.2. Усредняющий арифметический фильтр	42
3.3.3. Рекуррентная каузальная фильтрация	43

3.3.4. Фильтрация в частотной области	44
3.4. Нелинейная фильтрация	45
3.4.1. Медианные фильтры	45
3.4.2. Усредняющие нелинейные фильтры	47
3.5. Морфологические фильтры	46
4. СЕГМЕНТАЦИЯ ИЗОБРАЖЕНИЙ	49
4.1. Градиентные методы. Вычисление первой производной	52
4.2. Метод вторых производных (Лапласиан)	56
4.3. Метод Лапласиана Гауссиана	57
4.4. Метод Канни	58
5. МНОГОМЕРНЫЕ ЦЕПИ МАРКОВА ПРИ ОБРАБОТКЕ ИЗОБРАЖЕНИЙ	59
5.1. Многомерная марковская модель изображения	59
5.1.1. Двумерная марковская модель изображения	62
5.1.3. Трехмерная марковская модель изображения	65
5.2. Многомерная нелинейная фильтрация изображения	67
5.3. Сегментация изображений на основе цепей Маркова	73
5.3.1. Сегментация изображений на основе выделения контуров	73
5.3.2. Текстурная сегментация изображений	76
5.4. Сжатие изображений	77
6. ЗАДАНИЯ К ЛАБОРАТОРНЫМ И ПРАКТИЧЕСКИМ РАБОТАМ	79
6.1. Практическое занятие 1. Представление цифрового изображения	79
6.2. Практическое занятие 2. Статистическое сжатие изображений	80
6.3. Практическое занятие 3. Обработка изображений с помощью дискретного косинусного преобразования	82
6.4. Практическое занятие 4. Обработка изображений с помощью дискретного вейвлет-преобразования	83
6.5. Практическое занятие 5. Формирование искусственных изображений на основе двумерной математической модели	85
6.6. Практическое занятие 6. Восстановление изображений по отдельным отсчетам с использованием двух- и трехмерной математической модели	87
6.7. Лабораторная работа №1. Исследование методов фильтрации изображений.	89
6.8. Лабораторная работа №2. «Исследование методов выделения контуров на изображениях	94
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	104

ВВЕДЕНИЕ

Цифровая обработка изображений является неотъемлемой частью различных видеоинформационных систем: наблюдения, связи, мониторинга, технического зрения, телевизионных и др. При обработке изображений решается широкий спектр задач, направленных на повышение качества изображений, искаженных случайными помехами, эффективное использование ресурсов при передаче и хранении изображений, выделение и распознавание объектов интереса и др.

Цифровые изображения и их последовательности представляют собой многомерные сигналы. Поэтому решение научных и инженерных задач при работе с изображениями требует особых усилий, опирающихся на знание специфических методов, поскольку традиционная идеология одномерных сигналов и систем мало пригодна в этих случаях.

В данном пособии рассматриваются методы сжатия, восстановления зашумленных изображений и выделения объектов интереса. Наряду с известными методами обработки изображений в пособие рассмотрены новые методы обработки на основе многомерных цепей Маркова. Также пособие содержит задания к практическим и лабораторным работам по дисциплине «Цифровая обработка изображений».

Пособие предназначено для изучения и усвоения принципов цифровой обработки изображений в современных видеоинформационных системах, а также для получения практических навыков по моделированию различных алгоритмов обработки изображений, оценки качества восстановленных изображений, их кодированию.

Пособие будет полезно студентам и аспирантам радиотехнических и связных специальностей, а также научно-техническими работниками, специализирующимися в области цифровой обработки изображений.

1. ПРЕДСТАВЛЕНИЕ ИЗОБРАЖЕНИЙ

1.1. Типы изображений

Бинарные изображения – изображения, состоящие только из черных и белых пикселей.

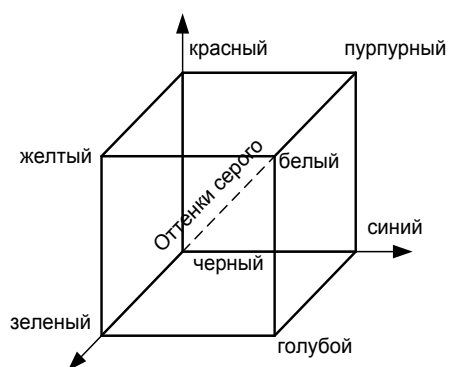
Полутонные изображения – изображения, в которых один пиксел занимает в запоминающем устройстве g бит, т. е. может принимать значение от 0 до 2^g . Это численное значение характеризует яркость элемента изображения.

Цветные изображения – изображения, требующие для точной передачи их цвета не менее трех g -разрядных чисел на один пиксел.

1.2. Цветовые системы

Цветовая система RGB

В цветовом пространстве RGB [1–5] пиксели цветного изображения представляются с помощью трех чисел, указывающих относительное соотношение красного (red), зеленого (green), голубого (blue) цветов. Поскольку красная, зелёная и синяя составляющие цвета элемента изображения задаются, численным значением от 0 до 255, для запоминания каждого элемента изображения требуется 24 бита. Такое изображение теоретически может содержать до 16,8 миллиона различных цветов.



Цветовую систему RGB можно представить в виде куба (рис. 1.1), для которого каждая его пространственная точка определяется координатами X , Y , Z . На главной диагонали куба расположены серые цвета.

Рис. 1.1. Взаимосвязь цветов в цветовой модели RGB

В цветовом пространстве RGB все три цвета считаются одинаково важными, и они обычно сохраняются с одинаковым разрешением.

При практическом применении системы RGB встречаются две проблемы: 1) это зависимость от аппаратуры; 2) невозможность получать цвета путем аддитивного цветового синтеза. Модель RGB используют в компьютерной графике.

Цветовая система YCbCr

В цветовом пространстве YCbCr (YUV) [1–4] используется компонента яркости (Y) и две хроматические компоненты (Cb, Cr). Яркостная компонента вычисляется как взвешенное усреднение компонент R, G, B по формуле:

$$Y = k_r R + k_g G + k_b B, \quad (1.1)$$

где k обозначает соответствующий весовой множитель.

По рекомендации ITU-T с идентификатором BT.601:

$$\begin{aligned} Y &= 0,299R + 0,587G + 0,114B, \\ Cb &= 0,564(B - Y), \\ Cr &= 0,713(R - Y) \end{aligned} \quad (1.2)$$

или

$$\begin{aligned} R &= Y + 1,402Cr, \\ G &= Y - 0,344Cb - 0,714Cr, \\ B &= Y + 1,772Cb. \end{aligned} \quad (1.3)$$

Преимущество пространства YCbCr по сравнению с RGB заключается в том, что компоненты Cb и Cr можно представить с меньшим разрешением, чем Y, так как глаз человека менее чувствителен к цвету предметов, чем к их яркости. Это позволяет сократить объем информации, требуемый для представления хроматических компонент, без заметного ухудшения качества передачи цветовых оттенков изображения. Модель YCbCr ориентирована на работу с цветопередающей аппаратурой.

1.3. Сэмплирование

Сэмплы [1–3] – это элементы, из которых состоят цветные (яркостные компоненты).

На рис. 1.2 показаны форматы сэмплирования для компонент Y, Cb, Cr [1–3]. Формат 4:4:4 подразумевает, что все три компоненты (Y, Cb, Cr) имеют одинаковое разрешение. То есть для каждой из четырех компонент яркости отбирается по четыре хроматической компоненты. Сэмплирование по формату 4:4:4 означает полную точность в передаче хроматических компонент.

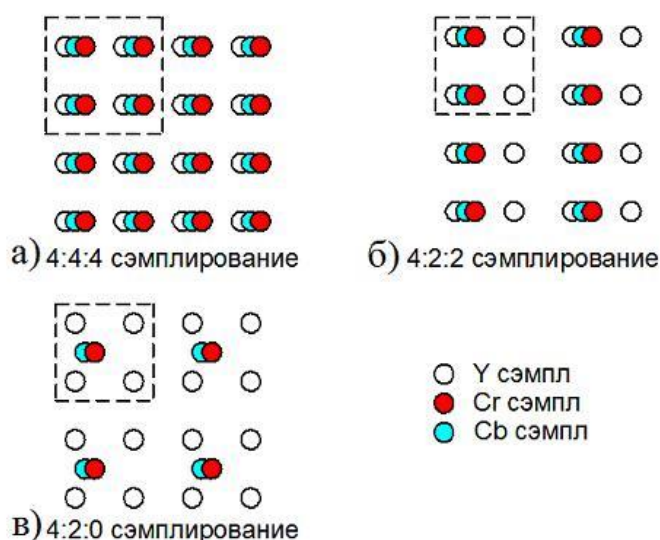


Рис. 1.2. Форматы сэмплирования

При сэмплировании по формату 4:2:2 хроматические компоненты по вертикали имеют одинаковое разрешение с яркостью, а по горизонтали они имеют половину от разрешения яркости (числа 4:2:2 означают, что на каждые четыре сэмпла яркости Y отбирается только две компоненты Cb и две компоненты Cr).

В формате сэмплирования 4:2:0 каждая компонента Cb и Cr имеет и по вертикали и по горизонтали половину разрешения по сравнению с Y. Поскольку хроматические компоненты отбираются в четыре раза реже компонент яркости, то пространство 4:2:0 YCbCr требует в два раза меньше сэмплов по сравнению с форматом 4:4:4 (или RGB).

2. СЖАТИЕ ИЗОБРАЖЕНИЙ

2.1. Требования к алгоритмам сжатия изображений

К алгоритмам сжатия изображений предъявляются следующие требования [1–4,6–9]:

- высокая степень компрессии;
- высокое качество изображений;
- высокая скорость компрессии и декомпрессии;
- симметричность по времени;
- небольшая стоимость аппаратной реализации. Эффективность программной реализации.

Степень компрессии определяется отношением количества элементов несжатого изображения к количеству элементов сжатого изображения.

Визуальное качество изображений измеряется с помощью субъективных и объективных критериев. Субъективное оценивание качества осуществляется с помощью тестовых процедур рекомендованных ITU-R BT.500-11. Наиболее популярной мерой объективной оценки качества служит пиковое соотношение сигнал/шум, которое вычисляется по средне-квадратичному отклонению исходного изображения от преобразованного видеокадра относительно квадрата наибольшего возможного значения пиксела.

Высокая скорость компрессии и декомпрессии определяется количеством вычислительных операций и применяемых технологий.

Симметричность во времени – это отношение времени кодирования к времени декодирования.

Большинство предъявляемых требований к алгоритмам сжатия являются противоречивыми. Поэтому на практике выбирают метод сжатия по 2–3 характеристикам, а остальные характеристики стараются оптимизировать.

Существует две большие группы алгоритмов сжатия изображений: алгоритмы сжатия с потерями и без потерь качества.

Алгоритмы сжатия без потери информации обнаруживают в документах избыточные или повторяющиеся данные, а затем кодируют их [1–4, б]. При этом исходная информация может быть восстановлена полностью. Однако для сжатия полутоновых и цветных изображений подобные алгоритмы дают весьма скромные результаты и поэтому мало приемлемы. Большее сжатие достигается применением алгоритмов сжатия с потерями качества ценой неполного совпадения декодированного сигнала с его оригиналом, в которых избыточная информация устраняется без визуальных изменений изображения.

Алгоритмы сжатия без потерь (методы энтропийного кодирования) делят на адаптивные и неадаптивные [1].

В неадаптивных методах способ кодирования не меняется в процессе обработки данных, а в адаптивных – кодирование подстраивается под изменяющиеся статистические характеристики потока входных данных.

При использовании неадаптивного варианта алгоритма необходимо передать на декодирующую сторону таблицу распределения вероятностей.

При адаптивном варианте – таблица распределения вероятностей пересчитывается как на передающей, так и на приемной стороне по мере поступления данных.

Алгоритмы кодирования могут быть однократными и многократными, т. е. требующими сбора предварительной информации об обрабатываемых данных до начала непосредственного процесса кодирования.

Наиболее известными алгоритмами сжатия без потерь являются следующие:

- кодирование длин серий (RLE);
- словарные методы (LZW);
- алгоритм Хаффмана;
- арифметическое кодирование.

В алгоритмах сжатия с потерями [1–4,6,9,10] сначала удаляют визуальную избыточность, используя преобразования на основе блоков (Кархунена-Лоэбэ, разложение по сингулярным числам, дискретное косинус-преобразование) или преобразования на основе изображений (например,

дискретное вейвлетное преобразование). После преобразования осуществляется квантование коэффициентов. Дальнейшее сжатие достигается за счет группового кодирования преобразованных данных (Run Length Encoding – RLE) и энтропийного кодирования.

2.2. Оценка качества изображений

2.1.1. Субъективная оценка качества

Несколько тестовых процедур субъективного оценивания качества объединено в рекомендацию ITU-R BT.500-11. Наиболее часто используемой процедурой из этого стандарта является метод непрерывной шкалы качества двойного раздражения DSCQS (Double Stimulus Continuous Quality Scale), в котором эксперту демонстрируются пары изображений коротких видеопоследовательностей эталонной и тестируемой и ему предлагается оценить качество фрагментов тестируемой последовательности.

В таблице 2.1 приведены основные шкалы субъективных оценок качества изображения [8].

Таблица 2.1

Основные шкалы субъективных оценок качества изображения

Нормализованная	Пятибальная шкала		Семибальная шкала ухудшения
	Шкала качества	Шкала ухудшения	
1	5 (отлично)	5 (незаметно)	1 (незаметно)
0,875			2 (едва заметно)
0,75	4 (хорошо)	4 (заметно, но не мешает)	3 (заметно, но лишь слегка ухудшает)
0,625	3 (удовлетворительно)	3 (заметно, немного мешает)	4 (ухудшает, но не мешает)
0,5			5 (несколько мешает)
0,25	2 (плохо)	2 (мешает)	6 (определенно мешает)
0	1 (очень плохо)	1 (сильно мешает)	7 (крайне мешает)

Такой подход дает наилучшую оценку качества изображения, но требует группы людей и дорог в практическом применении.

2.2.2. Объективная оценка качества

Основной задачей выбора объективных систем оценивания является выбор критериев, которые могли бы автоматически и с большой степенью достоверности оценить качество предложенных изображений, т. е. как усреднено оценила бы его группа экспертов.

Среди объективных критериев качества оценки изображений наиболее популярными являются следующие критерии.

1. Среднеквадратичная ошибка:

$$CKO = \frac{1}{N} \sum_{i=1}^N (x_i - y_i)^2, \quad (2.1)$$

где N – число пикселей в изображении,

x_i, y_i – значения яркостей пикселей исходного и восстановленного изображений, соответственно.

2. Пиковое отношение сигнал/шум PSNR (Peak Signal to Noise Ratio):

$$PSNR = 20 \lg \frac{255}{\sqrt{CKO}}, \quad (2.2)$$

где CKO – среднеквадратичная ошибка,

3. Минимаксный критерий, или критерий максимального правдоподобия:

$$ML = \max_{i=1, \dots, N} |x_i - y_i|. \quad (2.3)$$

4. Индекс структурного сходства SSIM (structure similarity) исходного (x) и восстановленного (y) изображений:

$$SSIM(x, y) = \frac{(2\mu_x \mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \quad (2.4)$$

где μ_x – среднее x , μ_y – среднее y ,

σ_x^2 – дисперсия x , σ_y^2 – дисперсия y , σ_{xy} – ковариация x и y ,

$$c_1 = (k_1 L)^2, \quad c_2 = (k_2 L)^2,$$

L – динамический диапазон пикселей,

$k_1=0,01$ и $k_2=0,03$ – константы.

5. Критерий $CW - SSIM$:

$$CW - SSIM(C_x, C_y) = \frac{2 \left| \sum_{i=1}^N C_{x,i} C_{y,i}^* \right| + K}{\sum_{i=1}^N C_{x,i}^2 + \sum_{i=1}^N C_{y,i}^2 + K}, \quad (2.5)$$

где $C_{x,i}$, $C_{y,i}$ – вейвлет-коэффициенты, N – число коэффициентов, K – константа, предотвращающая деление на ноль.

2.3. Алгоритмы сжатия изображений без потерь качества

2.3.1 Групповое кодирование

В групповом кодировании (Run Length Encoding – RLE) изображение вытягивается в ряд по строкам раstra [6]. Сжатие в RLE происходит за счет того, что в исходном изображении встречаются цепочки одинаковых значений отсчетов. Такие цепочки заменяются на два числа: первое – количество элементов в цепочке, второе – значение элемента. Например, если в изображении встречается цепочка 7,7,7,0,0,0,0,5,5, то она заменяется на пары (3,7), (4,0), (2,5).

Характеристики алгоритма:

- средний коэффициент сжатия 2–3;
- максимальный коэффициент сжатия – 32–64;
- возможно увеличение файла;
- высокая скорость компрессии и декомпрессии;
- симметричность алгоритма примерно единица;
- не требует дополнительной памяти при архивации и разархивации;

- рассчитан на изображения с большими областями повторяющегося цвета;

- применяется в стандартах JPEG, MPEGx (x=1,2,4), H.26x (x=1,3-5).

2.3.2. Алгоритм Лемпеля-Зива-Велча

Сжатие в алгоритме Лемпеля-Зива-Велча (Lempel-Ziv-Welch - LZW) осуществляется за счет одинаковых цепочек байт [4,6].

Сначала создается таблица, в которую записывают все значения отсчетов изображения (т. е. с 0 до 255). Кроме того, таблица включает код очистки и код конца информации. Под коды для строк остаются значения словаря от 258 до 511. Добавляемые строки записываются в таблицу последовательно, при этом номер строки в словаре становится ее кодом.

Первоначально таблица будет выглядеть следующим образом:

Таблица 2.2

Словарь	Значение	Примечание
0	0	Всевозможные строки, состоящие из одного элемента изображения
1	1	
2	2	
...	...	
255	255	
256	256	Код очистки
257	257	Код конца информации
258		Коды строк (9 бит)
...		
511		

Алгоритм сжатия LZW состоит из следующих этапов.

1. Входной поток данных разбивается на фразы. Причем каждая новая фраза начинается с последнего символа предыдущей фразы и не должна совпадать ни с одной из предшествующих фраз. Минимальная длина фразы два символа.

2. Фразы закрепляются за определенным номером словаря.

3. Выходной код формируется следующим образом:

- если фраза состоит из двух символов, то выходной код соответствует первому символу фразы (левому);

- если фразу составляют большее число символов, то выходной код соответствует номеру словаря.

Пример. Пусть исходное изображение $\{7,7,5,5,0,0,0,0,0,0,5,5,0\}$ содержит 8 бит/отсчет.

Таблица, содержащая значения строк (фраз), а также кодовые слова, будет выглядеть следующим образом:

Таблица 2.3

Словарь	Значение строк (фразы)	Выходной поток
256	256	256
258	7,7	7
259	7,5	7
260	5,5	5
261	5,0	5
262	0,0	0
263	0,0,0	262
264	0,0,0,5	263
265	5,5,0	260

То есть архивацию можно представить так:

7 – есть в таблице;

7,7 – нет. Добавляем в таблицу 258 – «7,7». В выходной поток -7; ... ;

0 – есть в таблице;

0,0 – есть в таблице;

0,0,0 нет. Добавляем в таблицу 263 – «0,0,0». В выходной поток – значение словаря похожей фразы 0,0 и т. д.

Коэффициент сжатия для рассматриваемого примера будет составлять: $K_c = 13 \cdot 9 / 9 \cdot 9 = 1,44$.

Таблицу строк можно восстановить, пользуясь только потоком кодов.

Характеристики алгоритма:

- минимальный коэффициент сжатия – $5/7$;
- средний коэффициент сжатия – 4;
- симметричность: 1,2-3;
- для определения одного элемента изображений требуется некоторое количество сравнений. Требуется память для хранения таблицы кодов;
- LZW ориентирован на 8-битные изображения. Сжимает за счет одинаковых подцепочек в потоке;
- варианты LZW используются в архиваторах GIF, TIFF.

2.3.3. Алгоритм Хаффмана

Алгоритм Хаффмана основан на использовании статистических словарей [1,2,6] и включает в себя последовательность следующих действий:

1. Все имеющиеся K сообщений располагают в один столбик в порядке убывания вероятностей.

2. Затем два самых маловероятных сообщения a_{K-1}, a_K объединяют в одно сообщение b , которое имеет вероятность $p_{K-1} + p_K$. В результате получают сообщения $a_1, a_2, \dots, a_{K-2}, b$, вероятности которых $p_1, p_2, \dots, p_{K-2}, p_{K-1} + p_K$. Эти сообщения вновь располагают в порядке убывания вероятностей.

3. Далее берут два сообщения, имеющие наименьшие вероятности, объединяют их в одно и вычисляют их общую вероятность.

4. Шаги 2 и 3 повторяют до тех пор, пока не получат единственное сообщение, вероятность которого равна единице.

5. Проводя линии, объединяющие сообщения и образующие последовательные подмножества, получают дерево, в котором отдельные сообщения являются концевыми узлами. Соответствующие им кодовые слова можно построить, приписывая ветви, которые идут вниз – «0», а вверх – «1».

Характеристики алгоритма:

- минимальный коэффициент сжатия – 1/5;
- средний коэффициент сжатия – 2;
- симметричность: близка к единице;
- используются в стандартах JPEG, MPEGx (x=1,2,4), H.26x (x=1,3-5);
- ориентирован для черно-белых изображений с большими областями повторяющегося цвета;
- алгоритм прост в реализации, быстр и может быть легко реализован аппаратно.

Пример. Дискретный источник выдает символы из ансамбля $\{a_i\}$, с вероятностями $p(a_1) = 0,3$; $p(a_2) = 0,15$; $p(a_3) = 0,25$; $p(a_4) = 0,15$; $p(a_5) = 0,1$; $p(a_6) = 0,05$. Закодировать символы данного ансамбля кодом Хаффмана. Процесс кодирования представлен на рис. 2.1.

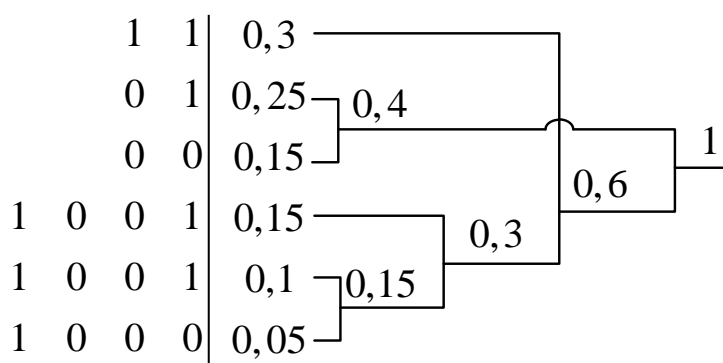


Рис. 2.1 Процесс кодирования методом Хаффмана

2.3.4. Арифметическое кодирование

При арифметическом кодировании [1,3,6] каждый символ исходного текста представляется отрезком на числовой оси с длиной, равной вероятности его появления, и началом, совпадающим с концом отрезка символа, предшествующего ему в алфавите. Результатом арифметического кодирования является некоторая двоичная дробь из интервала $[0, 1)$.

Характеристики алгоритма:

- средний коэффициент сжатия 5-6;

- требует больших вычислительных ресурсов;
- используются в формате JPEG2000.

Пример. Закодировать сообщение «ааа».

Вероятность $p(a)=3/4$; $p(b)=1/4$.

Пустому слову соответствует весь интервал $[0, 1)$. После получения каждого очередного символа арифметический кодер уменьшает интервал, выбирая ту его часть, которая соответствует вновь поступившему символу.

Процесс кодирования пояснен на рис. 2.2 и в таблице 2.4.

Таблица 2.4

Шаг	Просмотренная цепочка	Интервал (в десятичной форме)	Интервал (в двоичной форме)
0	«»	$[0, 1)$	$[0, 1)$
1	«a»	$[0, 3/4)$	$[0, 0.11)$
2	«aa»	$[0, 9/16)$	$[0, 0.1001)$
3	«aab»	$[27/64, 36/64)$	$[0.011011, 0.100100)$
4	«aaba»	$[108/256, 135/256)$	$[0.01101100, 0.10000111)$

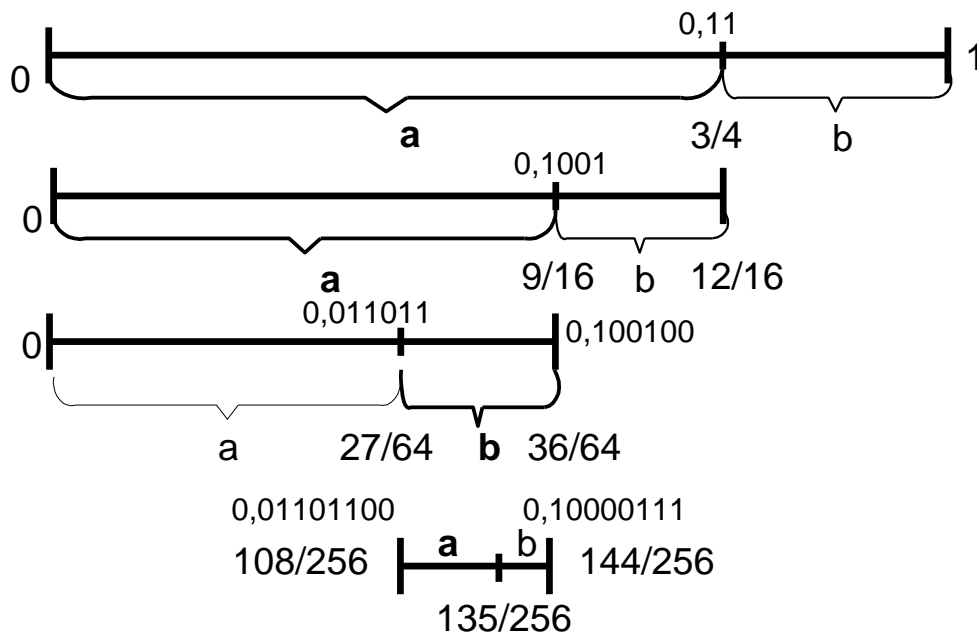


Рис. 2.2. Процесс архивирования арифметическим кодом

Интервал $[0, 1)$ разбивается на части, длина которых пропорциональна вероятностям символов.

На первом шаге берутся первые 3/4 интервала, соответствующие символу «а». На втором шаге от предыдущего интервала оставляют еще 3/4, соответствующие второму символу «а». После третьего шага от предыдущего интервала останется его правая четверть в соответствии с положением и вероятностью символа «b». На четвертом шаге оставляют лишь первые 3/4 от результата.

В качестве кода можно взять любое число из диапазона, полученного на шаге 4. Например, $0,5_{10} \Rightarrow 0,1_2$.

2.4. Алгоритмы кодирования с преобразованием

2.4.1. Дискретное косинусное преобразование

Дискретное косинус-преобразование (ДКП, DCT) оперирует с блоками размером $N \times N$, элементами которых служат яркости пикселей изображения, и порождает блок $N \times N$ коэффициентов Y [1–4].

Прямое ДКП, применимое к изображению X , задается формулой:

$$Y = AXA^T, \quad (2.6)$$

где X – матрица яркостей пикселей. Элементы матрицы A равны:

$$A_{ij} = C_i \cos \frac{(2j+1)i\pi}{2N}, \text{ где } C_i = \sqrt{\frac{1}{N}}, (i=0), C_i = \sqrt{\frac{2}{N}}, (i>0). \quad (2.7)$$

Обратное преобразование ДКП:

$$X = A^T Y A. \quad (2.8)$$

Пример. Вычислить ДКП для блока 4×4 .

$N = 4$. Матрица преобразования A (2.7) для 4×4 ДКП имеет вид:

$$A = \begin{bmatrix} \frac{1}{2} \cos(0) & \frac{1}{2} \cos(0) & \frac{1}{2} \cos(0) & \frac{1}{2} \cos(0) \\ \sqrt{\frac{1}{2}} \cos\left(\frac{\pi}{8}\right) & \sqrt{\frac{1}{2}} \cos\left(\frac{3\pi}{8}\right) & \sqrt{\frac{1}{2}} \cos\left(\frac{5\pi}{8}\right) & \sqrt{\frac{1}{2}} \cos\left(\frac{7\pi}{8}\right) \\ \sqrt{\frac{1}{2}} \cos\left(\frac{2\pi}{8}\right) & \sqrt{\frac{1}{2}} \cos\left(\frac{6\pi}{8}\right) & \sqrt{\frac{1}{2}} \cos\left(\frac{10\pi}{8}\right) & \sqrt{\frac{1}{2}} \cos\left(\frac{14\pi}{8}\right) \\ \sqrt{\frac{1}{2}} \cos\left(\frac{3\pi}{8}\right) & \sqrt{\frac{1}{2}} \cos\left(\frac{9\pi}{8}\right) & \sqrt{\frac{1}{2}} \cos\left(\frac{15\pi}{8}\right) & \sqrt{\frac{1}{2}} \cos\left(\frac{21\pi}{8}\right) \end{bmatrix}.$$

Вычислив косинусы, получаем матрицу числовых коэффициентов

$$A = \begin{bmatrix} 0,5 & 0,5 & 0,5 & 0,5 \\ 0,653 & 0,271 & -0,271 & -0,653 \\ 0,5 & -0,5 & -0,5 & 0,5 \\ 0,271 & -0,653 & 0,653 & -0,271 \end{bmatrix}.$$

Пусть X – блок яркостей пикселей изображения:

$$X = \begin{bmatrix} 5 & 11 & 8 & 10 \\ 9 & 8 & 4 & 12 \\ 1 & 10 & 11 & 4 \\ 19 & 6 & 15 & 7 \end{bmatrix},$$

тогда

$$Y' = AX = \begin{bmatrix} 17 & 17,5 & 19 & 16,5 \\ -6,981 & 2,725 & -6,467 & 4,125 \\ 7 & -0,5 & 4 & 0,5 \\ -9,015 & 2,660 & 2,679 & -4,414 \end{bmatrix},$$

$$Y = AXA^T = \begin{bmatrix} 35,0 & -0,079 & -1,5 & 1,115 \\ -3,299 & -4,768 & 0,443 & -9,010 \\ 5,5 & 3,029 & 2,0 & 4,699 \\ -4,045 & -3,010 & -9,384 & -1,232 \end{bmatrix}.$$

2.4.2. Дискретное вейвлет-преобразование

Вейвлеты – это семейство математических функций определенной формы, локальных как во временной, так и в частотной областях [1,9,10]. При преобразовании и кодировании сигналов применяют дискретное вейвлет преобразование (ДВП, DWT – Discrete Wavelet Transform). Вейвлет-функция $\psi(t)$ определяется той или иной базисной функцией $\psi_0(t)$ с применением к ней операций смещения во времени и масштабирования:

$$\psi(t) \equiv \psi_{a,b}(t) = a^{-1/2} \psi_0\left(\frac{t-b}{a}\right), \quad (2.9)$$

где a и b – действительные числа, параметр $a > 0$ определяет масштаб функции $\psi(t)$, а b – ее смещение по временной оси.

В качестве примера аналитически заданных базисных вейвлет-функций $\psi_0(t) \equiv \psi_{1,0}(t)$ можно привести вейвлет Хаара:

$$\psi_H(t) = \begin{cases} 1, & \text{при } 0 \leq t < 1/2, \\ -1, & \text{при } 1/2 \leq t < 1, \\ 0, & \text{иначе.} \end{cases} \quad (2.10)$$

ДВП включает в себя следующие этапы [10]:

1. Одномерный сигнал $\{c_n^0\}_{n=0}^{N-1}$ (верхний индекс показывает уровень разложения, нижний – конкретную точку сигнала) подается на низкочастотный h и высокочастотный g фильтры разложения.

2. Результат обработки фильтров поступает на блоки децимации или прореживания. В классическом алгоритме Малла прореживание происходит в два раза.

3. Результатом обработки п.1 и п.2 являются два набора вейвлет-коэффициентов $\{c_n^1\}$ и $\{d_n^1\}$. Высокочастотные коэффициенты $\{d_n^1\}$ остаются без изменения, а низкочастотные $\{c_n^1\}$ преобразуются аналогично преобразованию исходного сигнала.

Сказанное выше можно представить математически:

$$\begin{cases} c_n^m = \sum_l h_{l-2n} c_l^{m-1} \\ d_n^m = \sum_l g_{l-2n} c_l^{m-1} \end{cases}, \quad (2.11)$$

где индекс m указывает на уровень разложения, а выражение (2.11) представляет собой свертку сигнала или его низкочастотных компонент с импульсными характеристиками низкочастотных и высокочастотных фильтров.

Для вейвлета Хаара коэффициенты фильтров h и g соответственно равны:

$$\left\{ h_0 = \frac{1}{2}, h_1 = \frac{1}{2} \right\} \text{ и } \left\{ g_0 = \frac{1}{2}, g_1 = -\frac{1}{2} \right\}.$$

Вычисление последовательностей коэффициентов в соответствии с (2.11) можно представить выражениями:

$$c_1^m = (c_{i,j} + c_{i+1,j} + c_{i,j+1} + c_{i+1,j+1})/4;$$

$$d_2^m = (c_{i,j} + c_{i+1,j} - c_{i,j+1} - c_{i+1,j+1})/4;$$

$$d_3^m = (c_{i,j} - c_{i+1,j} + c_{i,j+1} - c_{i+1,j+1})/4;$$

$$d_4^m = (c_{i,j} - c_{i+1,j} - c_{i,j+1} + c_{i+1,j+1})/4.$$

Схема вейвлет – преобразования представлена на рис. 2.3.

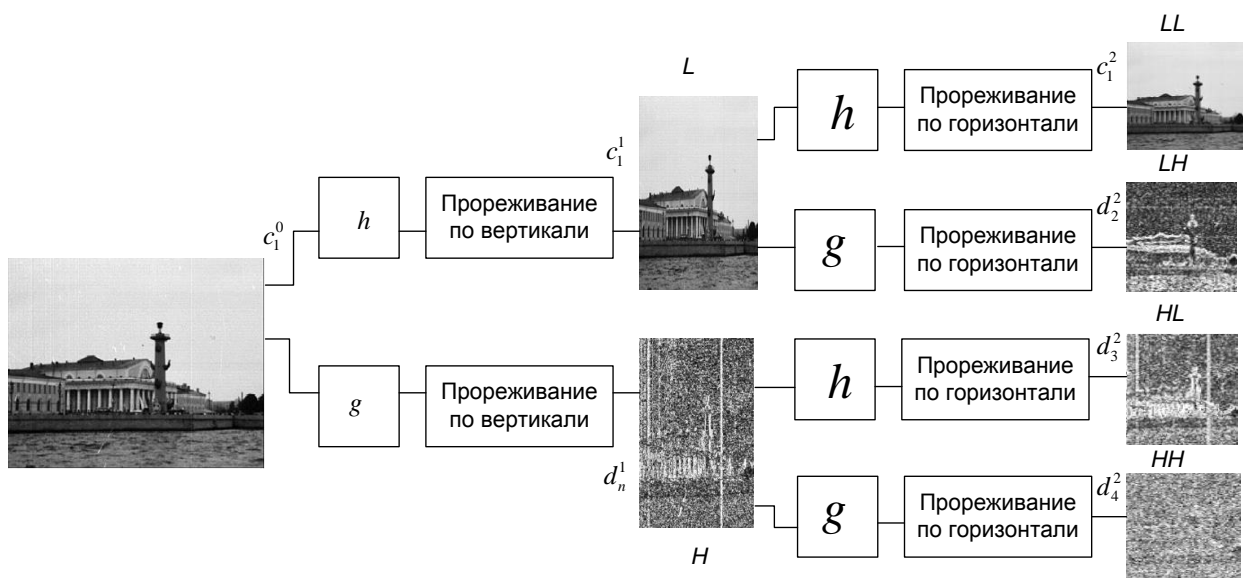


Рис. 2.3. Схема вейвлет-преобразования

Пояснения к схеме:

1. Каждая строка двумерного изображения фильтруется низкочастотным и высокочастотным фильтром.
2. Выход каждого фильтра прореживается с шагом 2, при этом получаются 4 субизображения.

Четыре «субполосных» изображений можно объединить так, чтобы получить равно столько же пикселей, что и в исходном изображении. LL – это исходное изображение, из которого удалены высокочастотные составляющие по горизонтали и вертикали и сделано прореживание с шагом 2; HL – это высокочастотная составляющая по горизонтали и оставшаяся низкочастотная по вертикали; LH – это низкочастотная составляющая по горизонтали и высокочастотная по вертикали; HH – это высокочастотные составляющие по горизонтали и вертикали.

3. Для дальнейшего сжатия двумерное вейвлет – разложение повторно применяется к части LL , из которой формируется четыре новых субполосных изображения.

Большинство коэффициентов высокочастотных составляющих близки к нулю и эффект сжатия можно получить, удалив малозначимые коэффициенты.

2.5. Стандарты сжатия статических изображений

2.5.1. Алгоритм JPEG

Алгоритм основан на дискретном косинусном преобразовании, применяемом к матрице изображения для получения некоторой новой матрицы коэффициентов. Для получения исходного изображения применяется обратное преобразование.

Алгоритм JPEG включает в себя следующую последовательность операций.

1. Изображение из цветового пространства RGB переводится в цветовое пространство YCrCb (YUV):
2. Каждая компонента изображения разбивается на блоки 8x8. При этом к цветоразностным компонентам предварительно может быть применено сэмплирование.

3. К каждой рабочей матрице применяется ДКП. В полученной матрице коэффициенты, расположенные в левом верхнем углу, соответствуют низкочастотной составляющей изображения, а в правом нижнем – высокочастотной.

4. Выполняют квантование, в результате которого малозначимые коэффициенты в блоке заменяют нулями.

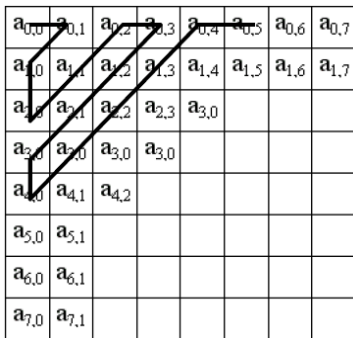


Рис. 2.4. «Зигзаг» – сканирование

5. Выполняют «зигзаг» – сканирование, т. е. матрицу 8x8 переводят в 64-элементный вектор (рис. 2.4). В результате такого сканирования в начале вектора получают коэффициенты, соответствующие низким частотам, а в конце – высоким.

6. Вектор свертывают с помощью алгоритма группового кодирования. При этом получаем пары типа (пропустить, число), где «пропустить» – счетчик пропускаемых нулей, а «число» – значение, которое необходимо поставить в следующую ячейку. Так, вектор 42 3 0 0 0 -2 0 0 0 0 1... будет свернут в пары (0,42) (0,3) (3,-2) (4,1)...

7. Получившиеся пары свертывают кодированием по Хаффману с фиксированной таблицей.

На рис. 2.5 представлена последовательность операций, применяемых в алгоритме JPEG.



Рис. 2.5. Последовательность операций в алгоритме JPEG

Характеристики алгоритма:

- коэффициенты компрессии: 2-200, средний 5-10;
- класс изображений: полноцветные 24 битные изображения или изображения в градациях серого без резких переходов цветов;

- симметричен;
- в некоторых случаях, алгоритм создает «ореол» вокруг резких горизонтальных и вертикальных границ в изображении (эффект Гиббса);

2.5.2. Алгоритм JPEG 2000

Базовая схема JPEG 2000 аналогична базовой схеме JPEG. Основные отличия заключаются в следующем [1–4]:

- вместо дискретного косинусного преобразования, используется дискретное вейвлет-преобразование;
- вместо кодирования по Хаффману используется арифметическое сжатие;
- в алгоритм изначально заложено управление качеством областей изображения;
- не используется явно дискретизация компонент U и V после преобразования цветовых пространств.

Последовательность операций, применяемых в алгоритме JPEG 2000 показана на рис. 2.6.



Рис. 2.6. Последовательность операций в алгоритме JPEG 2000

Характеристики алгоритма JPEG 2000:

- степень сжатия: 2-200;
- класс изображений: полноцветные 24-битные изображения; изображения в градациях серого без резких переходов цветов; однобитные изображения;
- симметричность: 1-1,5;

- требует больших вычислительных ресурсов, чем JPEG;
- искажения: при большом сжатии – артефакты в виде окантовок и посторонних узоров.

2.6. Стандарты сжатия видео

Сжатие видео достигается удалением статистической и визуальной избыточности. Видеокодер состоит из трех основных функциональных единиц [1,3,12–15]: временной модели, пространственной модели и энтропийного кодера (рис. 2.7). Временная модель сокращает временную избыточность, используя схожесть между последовательными видеокадрами. Отличия между кадрами могут появляться из-за движения объектов, перемещения видеокамеры, изменения освещенности. Для увеличения схожести между соседними кадрами во всех современных стандартах цифрового видеокодирования используется метод компенсации движения, который увеличивает корреляцию между соседними кадрами за счет перемещения прямоугольных блоков в текущем кадре, тем самым уменьшая энергию в разностном кадре. Выходом временной модели служит остаточный кадр и некоторое семейство числовых параметров модели. Остаточный кадр является входом для пространственной модели, которая использует схожесть соседних элементов этого кадра, сокращает тем самым пространственную избыточность. Выходом пространственной модели выступает семейство квантованных коэффициентов преобразования. Параметры временной модели и пространственной модели сжимаются энтропийным кодером. При этом удаляется статистическая избыточность данных и формируется битовый поток или файл, который можно передавать по сетям или хранить на цифровых носителях данных.

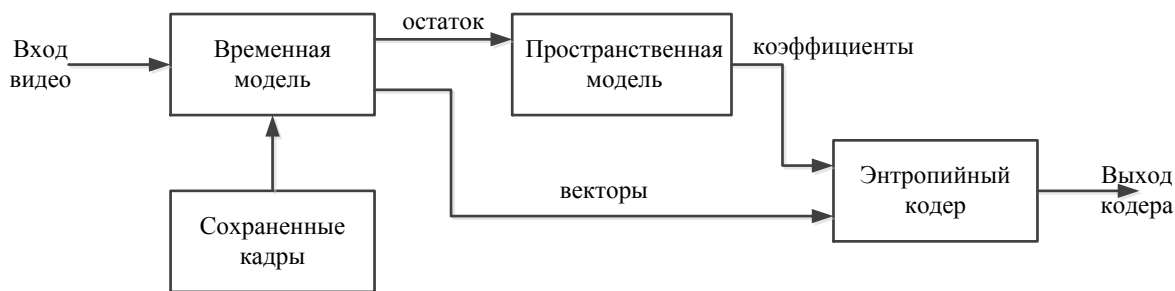


Рис. 2.7. Блок-схема видеокодера

Базовой основой стандартов сжатия видео MPEG-х (х=1,2,4), H.26х (х=1,3,4,5) является модель на основе ДКП, квантования, энтропийного кодирования и компенсации движения.

2.6.1. MPEG-1

MPEG-1 был разработан с целью сжатия мультимедийного контекста, поставляемого на CD-ROM при скорости передачи 1,5 Мбит/с [1,3,11,12].

Для того чтобы удовлетворить противоречивым требованиям и увеличить гибкость алгоритма, рассматриваются 4 типа кадров:

- I-кадры – кадры, сжатые независимо от других кадров (I-Intra pictures);
- P-кадры – сжатые с использованием ссылки на одно изображение (P-Predicted);
- B-кадры- сжатые с использованием ссылки на два изображения (B-Bidirection);
- DC-кадры – независимо сжатые с большой потерей качества.

Характеристики MPEG-1:

- поток, разрешение: 1.5 Мбит/с, 352x240x30, 352x288x25;
- малая гибкость формата;
- средний коэффициент сжатия: 8-30;

– алгоритм сравнительно прост в аппаратной реализации, содержит преобразования, поддерживаемые на аппаратном уровне большим количеством видеокарт.

2.6.2. MPEG-2

Основное отличие MPEG-2 от MPEG-1 – это эффективное кодирование черезстрочного видео [1,3,11,12].

Имеет широкий спектр скоростей передачи, видеоформатов, уровней качества и функциональности. Чтобы уменьшить стоимость реализации и сохранить функциональную совместимость, в MPEG-2 введены понятия профиля и уровня, так, что кодеры и декодеры, совместимые с одним и тем же профилем и уровнем, могут успешно взаимодействовать друг с другом, не поддерживая полный синтаксис. Профиль описывает необходимые функции. Уровень указывает возможность обработки, определяя верхние пределы пространственного разрешения, частоты кадров и скорости передачи.

Характеристики MPEG-2:

- поток, разрешение: 4-100 Мбит/с, 352x288, 720x576, 1440x1152, 1920x1152;
- форматы изображений: 4:2:0, 4:2:2, 4:4:4;
- высокая универсальность, масштабируемость, сравнительная простота аппаратной реализации;
- невысокая степень сжатия, недостаточная гибкость.

2.6.3. MPEG-4

Основные отличия стандарта MPEG-4 [1,3,11-13] от предыдущих стандартов:

1. Кодирование видеообъектов. Видеообъект берется как последовательность снимков объекта произвольной формы. Его временные слои называются видеообъектными плоскостями (VOP). Поскольку видеообъектные плоскости имеют высокую степень корреляции, предсказание с компенсацией движения является весьма эффективным. Понятия I-, P-, B-кадров в MPEG-4 расширяется до I-, P-, B- VOP, соответственно.

2. Кодирование текстур. Для кодирования текстур применяется ДВП.

3. Улучшение эффективного кодирования. MPEG-4 представляет собой новые методы для улучшения эффективности кодирования, включая INTRA-предсказание AC/DC, горизонтальное сканирование, глобальную компенсацию движения (GMC), размер блока 8x8 для компенсации движения и $\frac{1}{4}$ – пиксельное предсказание.

Характеристики MPEG-4 :

- поток: 5Кбит/с – 50 Мбит/с;
- при одинаковом коэффициенте сжатия качество видео лучше, чем в MPEG-2;
- масштабируемость;
- требует больших вычислительных ресурсов, чем MPEG-2.

2.6.4. H.261

H.261 был первым широко распространенным стандартом видеокодирования, разработанный ИТУ-Т поддержки видеотелефонии и видеоконференции через сети с коммутацией каналов ISDN [1,12].

В H.261 применяется цветовой формат 4:2:0. Анализ и компенсации движения подвергаются макроблоки размером 16×16 пикселей с компенсацией движения.

В выходной поток записываются два типа кадров: INTRA – сжатые независимо (соответствуют I-кадрам) и INTER – сжатые со ссылкой на предыдущий кадр (соответствуют P-кадрам). Сжатие в INTRA-кадрах осуществляется по схеме сжатия отдельного изображения. В INTER-кадрах

производится аналогичное сжатие разности каждого передаваемого макро-блока с «наиболее похожим» макроблоком из предыдущего кадра.

Характеристики H.261:

- поток: $p \times 64$ Кбит, $p=1..30$;
- прост в аппаратной реализации;
- невысокая степень сжатия;
- ограничения на формат.

2.6.5. H.263

Данный стандарт является расширением, дополнением и значительным усложнением H.261 [1,12]. Обеспечивает передачу видео по сетям с коммутацией каналов и пакетов.

Отличие базового варианта H.263 от H.261 состоит в использовании полупиксельного предсказания векторов движения, кодировании коэффициентов преобразования и т. д.

Характеристики H.263:

- поток: 0.04-20 Мбит/с;
- поддерживает значительное количество видеоформатов: sub-QCIF, QCIF, CIF, 4CIF и 16CIF.
- быстрая аппаратная реализация;
- имеет большую степень сжатия по сравнению с H.261 при том же качестве.

2.6.6. H.264

Стандарт H.264 ориентирован на эффективное сжатие и устойчивость к сетевым средам для универсального видеоконтекста и для кадра прямоугольной формы [1,3,12,14]. Некоторые тесты оценивают в 4 раза лучшее объективное качество H.264, чем у MPEG-4.

Стандарт H.264 имеет следующие усовершенствования и возможности.

Обладает большой гибкостью в выборе размеров и форм блоков при компенсации движения по сравнению со всеми предшествующими стандартами, причем минимальный размер блока составляет 4×4 пиксела и позволяет точнее выделять области движения. Стандарт повышает точность компенсации движения до четверти пиксела.

В стандарте расширяется методика увеличенного выбора опорного изображения, что поднимает эффективность кодирования, так как позволяет кодеру для компенсации движения делать выбор между большим количеством изображений, декодированных и сохраненных в декодере.

Стандарт обеспечивает кодеру во многих случаях возможность использовать для предсказания движения изображение, близкое по содержанию к кодируемому. Улучшена обработка пропущенных (skipped) блоков и блоков с прямым (direct) предсказанием движения.

В стандарте используется контекстно-адаптивное двоичное арифметическое кодирование САВАС (Context-adaptive binary arithmetic coding). Предусмотрено также использование контекстно-адаптивного кодирования кодами переменной длины (экспоненциальными кодами Голомба) – САVLC (Context-adaptive variable-length coding). САVLC имеет меньшую сложность, чем САВАС, но, тем не менее, сложнее и эффективнее, чем алгоритмы, применяемые для тех же целей в более ранних технологиях сжатия видео (алгоритм Хаффмана).

Повышена устойчивость к ошибкам и потерям данных и гибкость работы на множестве сетевых структур за счет структуры набора параметров, гибкий размер слоя, избыточного представления областей изображения, разбиения данных и т. д.

В таблице 2.5 приведены параметры исходных и компрессированных видеосигналов для некоторых приложений [1,11].

Таблица 2.5

Параметры исходных и компрессированных видеосигналов для некоторых приложений

Приложения	Формат и размер кадра	Модель цветности и разрядность, бит/пкл	Частота кадров, Гц и тип развертки	Скорость несжатого потока, Мбит/с	Стандарты компрессии	Скорость сжатого потока Мбит/с
Видеотелефония	subQCIF, QSIF, QCIF; 4:3	4:2:0, 12	1..5; прогрессивная	0,141..1,45	H.261, H.263, H.264, MPEG-4	0,009..0,13
Видеоконференцсвязь	QSIF, QCIF; SIF, CIF; 4:3	4:2:0, 12	5..30; прогрессивная	1,10..34,8	H.261, H.263, H.264, MPEG-4	0,032..1,50
Домашнее видео, видео по запросу	SIF, CIF; 4:3	4:2:0, 12	30/SIF; 25/SIF; прогрессивная		MPEG-1/2/4	0,7..1,5
Потоковое видео в Интернете и беспроводных сетях 3G	QSIF, CIF; 4:3/16:9	4:2:0, 12	25, 30; черезстрочная	0,141..34,8	Фирменные, H.263, PEG-4, H.264/AVC, VC-1	0,02..0,6
Стандартное ТВ	640x480/NTSC	4:2:0, 12 4:2:2, 16	30- NTSC, 25-	105..141	MPEG-2/4	2..20
	720x576 PAL/SECAM		PAL/SECAM черезстрочная	119..158		
ТВ повышенной четкости	1280x720 16:9	4:2:0, 12 4:2:2, 16	25/30/50/60; прогрессивная	264..844	MPEG-2/4	4..40

Продолжение таблицы 2.5

Приложения	Формат и размер кадра	Модель цветности и разрядность, бит/пкс	Частота кадров, Гц и тип развертки	Скорость несжатого потока, Мбит/с	Стандарты компрессии	Скорость сжатого потока Мбит/с
DVD, HD DVD, Blu-Ray	720x576, 4:3	4:2:0, 12	25/30; черезстрочная; 25, прогрессивная	10..949	MPEG-2, H.264/AVC, VC-1	1,858..9,8
	1920x1080 16:9					10..20
ТВЧ 1080	1920x1080; 16:9	4:2:2, 16	25/30; черезстрочная	791..949	MPEG-2/4, H.264/AVC	10..60
			25/30/50/60; прогрессивная	791..1900		
Цифровое кино	1920x1080 16:9	4:2:2, 20	60; прогрессивная	2370	MPEG-4, H.264/AVC	60..80
	3840x2160 16:9			9490		100..150
	5760x3240 16:9	4:4:4, 36	60; прогрессивная	38400	HEVC-High Efficiency Video Coding	150..600
	7680x4320 16:9			68300		
ТУВЧ	7680x4320 16:9	4:4:4, 36	60; прогрессивная	72000	HEVC-High Efficiency Video Coding	100
Трехмерное ТВ-SldexSlde; 2x1080	1920x1080 (2x960)x1800; 16:9	4:2:0, 12	50/60; черезстрочная	0,8..0,95	MPEG-2, MP&HL	0,065
					MPEG-4/ AVC HP	0,035

2.6.7. H.265

Стандарт H.265 (одобрен членами ITU-T 13 апреля 2013г.) разработан в связи с растущей потребностью в более высокой степени сжатия движущихся изображений для самых разных приложений, таких как потоковая передача в интернете, передача данных, видеоконференцсвязь, телевизионное вещание [15]. Основные отличия H.265 от стандарта H.264 заключаются в следующем.

Вместо макроблоков 16x16 используются блоки с древовидной структурой кодирования (от 64x64 до 8x8), размер которых выбирается кодером в зависимости от требований соответствующих приложений.

Увеличено количество режимов межкадрового и внутрикадрового предсказания до 35.

Тип преобразования для внутрикадрового кодирования – целочисленное с точным восстановлением. Адаптивный размер блока от 32x32 до 4x4.

Поддерживает схему энтропийного кодирования – CABAC (контекстно-адаптивное двоичное арифметическое кодирование).

Предусмотрено меньшее количество профилей (3 вместо 7) для обеспечения возможности взаимодействия между различными приложениями и устройствами.

Поддерживаемое разрешение – от 128x96 до 8192x4320.

3. ФИЛЬТРАЦИЯ ИЗОБРАЖЕНИЙ

Изображения, сформированные различными информационными системами, искажаются действием помех. Ослабление действия помех достигается фильтрацией. Задача фильтрации заключается в том, чтобы найти минимальное в среднем отличие результата обработки от идеального сигнала.

При цифровой обработке изображений принята декартовая система координат с началом в левом верхнем углу кадра и с положительными направлениями из этой точки вправо и вниз. На рис. 3.1 показаны примеры окрестностей различных типов, изображенные в виде совокупностей точек (пикселей). Рабочей точкой, в которой осуществляется обработка, является точка с координатами i, j . В зависимости от типа окрестности различают каузальную (а), некаузальную (б) и полукаузальную (в) фильтрацию изображений.

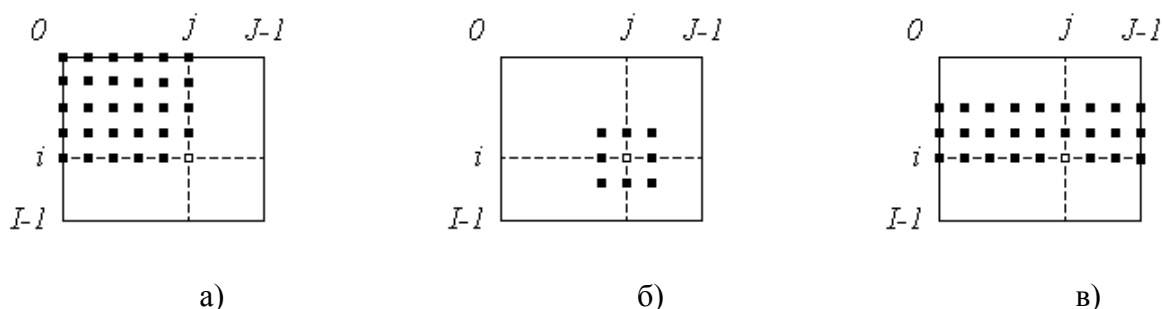


Рис. 3.1. Примеры окрестностей различных видов

При обработке сигналов в реальном масштабе времени следует применять каузальную фильтрацию. А некаузальную – при обработке ранее сформированных изображениями, уже хранящимися в памяти устройства обработки.

3.1. Модели шумов на изображениях

Сигналы изображений могут быть искажены случайными помехами, имеющими различную физическую природу. Наиболее часто рассматриваются следующие виды шумов [1,2,4,9,16].

Аддитивный шум

Изображение, зашумленное аддитивным шумом:

$$y(i, j) = x(i, j) + \eta(i, j), \quad (3.1)$$

где $x(i, j)$ – исходное изображение;

$\eta(i, j)$ – аддитивный шум с гауссовым (или другим) распределением функции плотности вероятности.

Мультипликативный шум

Описывается соотношением:

$$y(i, j) = x(i, j)\eta(i, j). \quad (3.2)$$

Импульсный шум

Под импульсным шумом понимается искажение сигнала импульсами, т. е. выбросами с очень большими положительными или отрицательными значениями и малой длительностью.

Импульсный шум может иметь фиксированные (соль-перец) и случайные значения импульсов.

$$y(i, j) = (1 - p)x(i, j) + p \cdot g(i, j), \quad (3.3)$$

где $g(i, j)$ – модель импульсного шума;

p – бинарный параметр, принимающий значения 0 и 1.

Шум квантования

Зависит от сигнала и характеризуется выбранным шагом квантования. Может производить нежелательные артефакты, например, ложные контуры вокруг объектов или устранять низкоконтрастные детали на изображении.

Спекл-шум

В силу того, что поверхность облучаемых деталей вызывают случайные фазовые отклонения отраженного излучения, это приводит к появлению гранулированного шума на изображении – спекл-шума.

Такой шум считается мультипликативным.

Функции плотности распределения вероятностей для описания модели шума

Гауссов шум

Возникает на изображении в результате возникновения таких факторов, как шум в электрических цепях, шум сенсоров – из-за недостатка освещения или высокой температуры.

Функция плотности распределения вероятностей случайной величины z :

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(z-\mu)^2/2\sigma^2}, \quad (3.4)$$

где z определяет уровень яркости из всевозможных значений;

μ – среднее значение случайной величины z ;

σ – среднеквадратичное отклонение.

Дисперсия σ^2 определяет мощность вносимого шума.

Если случайная величина z описывается выражением (3.4), то около 70 % ее значений сосредоточено в диапазоне $[(\mu - \sigma), (\mu + \sigma)]$ и около 95 % в диапазоне $[(\mu - 2\sigma), (\mu + 2\sigma)]$ (рис. 3.2).

Шум Релея

Распределение Релея полезно при моделировании шума, который возникает на снимках, снятых с большого расстояния.

Функция плотности распределения вероятностей для шума Релея:

$$p(z) = \begin{cases} \frac{2}{b}(z-a)e^{-(z-a)^2/b}, & z \geq a \\ 0, & z < a \end{cases}, \quad (3.5)$$

где a, b – некоторые положительные постоянные ($b \neq 0$), определяющие статистические характеристики распределения данного типа (рис. 3.3).

Математическое ожидание и дисперсия шума:

$$\mu = a + \sqrt{\pi b/4}, \quad \sigma^2 = \frac{b(4 - \pi)}{4}.$$

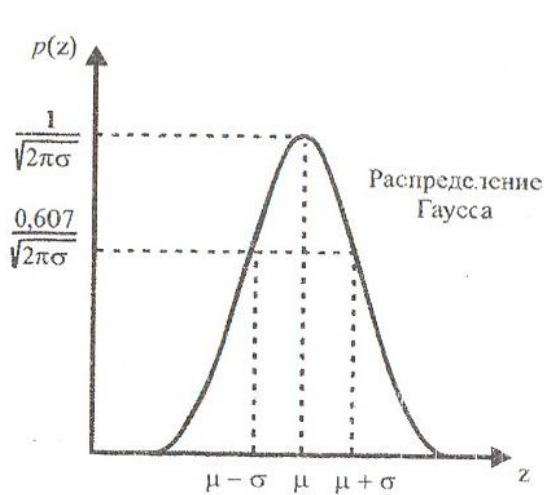


Рис. 3.2. Плотность распределения Гаусса

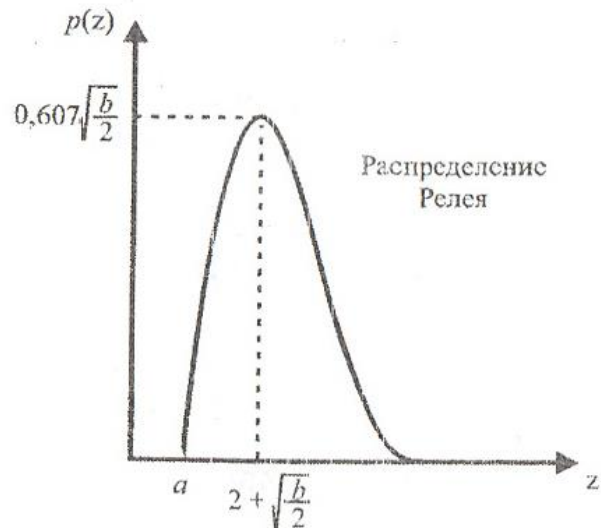


Рис. 3.3. Плотность распределения Релея

3.2. Типы фильтров для обработки изображений

Любой фильтр может быть определен как некоторый оператор $L[\cdot]$, который ставит в соответствие сигналу x некий сигнал y : $y = L[x]$.

Линейные и нелинейные фильтры [2,7, 16–18]

Для линейных фильтров: 1) линейный оператор $L[\cdot]$ должен удовлетворять принципу суперпозиции; 2) выходной сигнал является сверткой входной последовательности и импульсной характеристики.

Для нелинейных фильтров [2]: 1) оператор $L[\cdot]$ является нелинейным; 2) выходной сигнал не является сверткой входной последовательности и импульсной характеристики.

Пространственные и частотные [2,7, 16–18]

Пространственная фильтрация выполняется непосредственно над элементами изображения.

Частотная фильтрация связана с применением преобразования Фурье и обработкой сигнала в частотной области.

3.3. Пространственная линейная фильтрация

3.3.1. Масочная фильтрация изображений

Процесс основан на перемещении маски (фильтра, ядра, шаблона, окна) от точки к точке. В каждой точке (i, j) отклик фильтра вычисляется с использованием предварительно заданных связей. В случае линейной пространственной фильтрации отклик задается суммой произведений коэффициентов фильтров на соответствующие значения пикселей в области, покрытой маской фильтра.

Фильтрация изображения $y(i, j)$, имеющего размеры $M \times N$, задается выражением, определяющего свертку маски с изображением [2,4,16–18]:

$$x^*(i, j) = \sum_{(i_1, j_1) \in S} a(i_1, j_1) y(i - i_1, j - j_1), \quad (3.6)$$

где $x^*(i, j)$ – результат фильтрации полезного сигнала $x(i, j)$; S – множество точек, образующих окрестность; $a(i_1, j_1)$ – весовые коэффициенты, совокупность которых представляет собой двумерную импульсную характеристику (ИХ).

Наиболее распространенным критерием оптимальности, применяемым для оценки качества обработки, является критерий минимума среднего квадрата ошибок [16,17]:

$$E \left\{ [x(i, j) - x^*(i, j)]^2 \right\} = \min_{a(\cdot)}, \quad (3.7)$$

где $E\{x\}$ – математическое ожидание, вычисляемое по всем случайным величинам, содержащимся в (3.7).

Выполнив дифференцирование выражения (3.7) и, заменив, входящие в него математические ожидания отсчетами корреляционных функций, получим следующие выражение [16]:

$$B_{xy}(k, l) = \sum_{i_1, j_1 \in S} \sum a_{i_1, j_1} \cdot B_y(k - i_1, l - j_1), \quad (3.8)$$

где $k, l \in S$,

$B_{xy}(k, l) = E\{x_{i,j} \cdot y_{i-k, j-l}\}$ – взаимнокорреляционная функция,

$B_y(k - i_1, l - j_1) = E\{y_{i-i_1, j-j_1} \cdot y_{i-k, j-l}\}$ – автокорреляционная функция.

Алгебраическое уравнение (3.8) называют уравнением Винера-Хопфа. Если разрешить уравнение (3.8) относительно всех точек в окрестности S неизвестных $a_{i,j}$, то будет найдена искомая импульсная характеристика линейного фильтра, минимизирующего средний квадрат ошибок фильтрации.

Распространенным видом помехи является белый шум, аддитивно воздействующий на изображение. Наблюдаемое в этом случае изображение имеет вид:

$$y_{i,j} = x_{i,j} + n_{i,j}, \quad i = \overline{0, I-1}, \quad j = \overline{0, J-1}, \quad (3.9)$$

а корреляционная функция шума $n_{i,j}$ описывается выражением:

$$B_n(i, j) = D_n \cdot \delta_i \cdot \delta_j, \quad (3.10)$$

где D_n – дисперсия шума, а $\delta_k = \begin{cases} 1, & \text{при } k = 0 \\ 0, & \text{при } k \neq 0 \end{cases}$ – символ Кронекера.

Считаем, что входной сигнал имеет нулевое математическое ожидание, а изображение $x(i, j)$ и шум $n(i, j)$ взаимно независимы, поэтому для корреляционной функции входного сигнала справедливо:

$$B_y(i, j) = D_x \cdot R_x(i, j) + D_n \cdot \delta_i \cdot \delta_j,$$

где D_x – дисперсия, а $R_x(i, j)$ – нормированная корреляционная функция полезного сигнала.

Взаимная корреляционная функция $B_{xy}(o)$ в этих условиях совпадает с корреляционной функцией полезного сигнала $B_x(i, j)$. Поэтому уравнение Винера-Хопфа (3.8) приводится к виду [16]:

$$\left\{ \begin{array}{l} q^2 \cdot R_x(k, l) = q^2 \cdot \sum_{i_1, j_1 \in S} a_{i_1, j_1} \cdot R_x(k - i_1, l - j_1) + a_{k, l}, \\ k, l \in S \end{array} \right. , \quad (3.11)$$

где $q^2 = D_x/D_n$ – отношение дисперсий сигнала и шума, S – окрестность.

Рассмотрим примеры некоторых фильтров сглаживающих шум и фильтров подчеркивающих контуры [2,4,7].

Низкочастотные пространственные фильтры

Низкочастотные пространственные фильтры оставляют низкочастотные компоненты изображения нетронутыми и ослабляют высокочастотные компоненты. Сумма значений маски для всех низкочастотных фильтров равна 1. Одним из эвристических вариантов является равномерная маска, все девять весовых коэффициентов которой равны 1/9 (рис. 3.4а).

Низкочастотные фильтры усредняют быстрые изменения интенсивности, т. е. понижают содержание ВЧ-компонент, приводят к появлению размытости.

Высокочастотные пространственные фильтры

Высокочастотные фильтры используются для подчеркивания границ и выделения небольших объектов. Большое значение коэффициента в центре маски определяет действие высокочастотного фильтра (рис. 3.4б). По мере того, как центр с большим коэффициентом перемещается по изображению с высокой пространственной частотой (что означает большие изменения интенсивности элементов изображения), выходные значения пикселей многократно увеличиваются.

Меньшие отрицательные значения коэффициентов ядра, сгруппированные вокруг центра, уменьшают эффект большого весового фактора. В конечном итоге изменения интенсивности изображения усиливаются, а области постоянной интенсивности пикселей остаются неизменными.

$$\begin{array}{c} \left| \begin{array}{ccc} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{array} \right| \\ \cdot \\ \left| \begin{array}{ccc} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{array} \right| \end{array} \text{ а) } \qquad \begin{array}{c} \left| \begin{array}{ccc} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{array} \right| \\ \cdot \\ \left| \begin{array}{ccc} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{array} \right| \end{array} \text{ б) }$$

Рис. 3.4. Примеры масок

3.3.2. Усредняющий арифметический фильтр

Метод арифметического усреднения [2,4] заключается в том, что текущему пикселю изображения присваивается усредненное по всей окрестности текущего пикселя значение яркости:

$$x^*(i, j) = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N y(i, j), \quad (3.12)$$

где $x^*(i, j)$ – текущее значение пикселя выходного изображения, $y(i, j)$ – текущее значение пикселя входного изображения в области примыкания, M и N – ширина и высота области примыкания.

Арифметический метод может применяться с целью устранения высокочастотных помех и сглаживания изображений. Эффективен при достаточно слабом зашумлении исходного изображения. Применение арифметического фильтра оправдано в случае фильтрации изображений, зашумленных БГШ.

3.3.3. Рекуррентная каузальная фильтрация

Ограниченность размера окрестности, используемой масочным фильтром, приводит к его ограниченной способности подавления шума.

А увеличение размера маски приводит к увеличению вычислительной трудоемкости.

Примером рекуррентной фильтрации является калмановская фильтрация однородной стационарной гауссовской последовательности, имеющей корреляционную функцию экспоненциального вида

$$B_x(r) = D_x \cdot \exp(-\alpha \cdot |r|) = D_x \cdot F^{|r|}, \quad (3.13)$$

где D_x – дисперсия последовательности, а $F = \exp(-\alpha)$, $\alpha > 0$ – коэффициент ее одношаговой корреляции, определяемый параметром α , имеющим смысл ширины спектра последовательности. При ее наблюдении на фоне БГШ оптимальный каузальный фильтр реализуется рекуррентным алгоритмом, который в стационарном (установившемся) режиме фильтрации для двумерных ИХ имеет вид:

$$x_{i,j}^* = B_x \cdot F \cdot (x_{i-1,j}^* + x_{i,j-1}^*) - (B_x \cdot F)^2 \cdot x_{i-1,j-1}^* + A \cdot y_{i,j}. \quad (3.14)$$

где A – коэффициент усиления фильтра Калмана ($0 < A < 1$).

Первые два слагаемых в алгоритме (3.14) определяют вклад в оценку сигнала на текущем i -м шаге фильтрации, вносимый оценкой предыдущего шага. Третье учитывает влияние текущего наблюдения y_i . При высоком уровне шума параметр A приближается к нулю. При этом увеличивается параметр $B = 1 - A$, приближаясь к единице. Это означает удлинение импульсной характеристики и, следовательно, сужение полосы пропускаемых фильтром частот. Эти свойства ИХ способствуют эффективной фильтрации шума.

3.3.4. Фильтрация в частотной области

Приняв в уравнении Винера-Хопфа (3.8) область S импульсной характеристики неограниченной и заменив реальные функции $B_{xy}(\cdot), B_y(\cdot), a_{i_1, j_1}$ на периодические $\tilde{B}_{xy}(\cdot), \tilde{B}_y(\cdot), \tilde{a}_{i_1, j_1}$, имеющими двумерный период R_{I_1, I_2} , получим:

$$\begin{cases} \tilde{B}_{xy}(k, l) = \sum_{i_1=0}^{I_1-1} \sum_{j_1=0}^{I_2-1} \tilde{a}_{i_1, j_1} \tilde{B}_y(k - i_1, l - j_1), \\ k, l \in R_{I_1, I_2} \end{cases} \quad (3.15)$$

Периодичность функций, входящих в уравнение (3.8), позволяет применить к его обеим частям двумерное ДПФ:

$$\tilde{G}_{x,y}(k_1, k_2) = \tilde{A}_{k_1, k_2} \cdot \tilde{G}_y(k_1, k_2), \quad (3.16)$$

где $\tilde{G}_{x,y}(k_1, k_2)$, $\tilde{G}_y(k_1, k_2)$ – спектральные плотности мощности, представляющие собой ДПФ соответствующих корреляционных функций, а \tilde{A}_{k_1, k_2} – ДПФ искомой импульсной характеристики, являющееся частотным коэффициентом передачи фильтра Винера.

Решение уравнения (3.19) для частотной характеристики, имеет вид:

$$\tilde{A}_{k_1, k_2} = \frac{\tilde{G}_{x,y}(k_1, k_2)}{\tilde{G}_y(k_1, k_2)} \quad (3.17)$$

и дает эффективный способ осуществления оптимальной линейной фильтрации изображения. Его выполнение требует знания спектральных плотностей мощности и опирается на применение к обрабатываемому изображению дискретного преобразования Фурье.

3.4. Нелинейная фильтрация

2.4.1. Медианные фильтры

Все линейные алгоритмы фильтрации приводят к сглаживанию мелких деталей, размытию границ объектов на изображениях и эффективны при наличии БГШ небольшой интенсивности.

Однако, часто приходится иметь дело с изображениями, искаженными помехами других типов. Одной из них является импульсная помеха. При ее воздействии на изображении наблюдаются белые или (и) черные

точки, хаотически разбросанные по кадру. Применение линейной фильтрации в этом случае неэффективно – каждый из входных импульсов (по сути – дельта-функция) дает отклик в виде импульсной характеристики фильтра, а их совокупность способствует распространению помехи на всю площадь кадра.

Удачным решением перечисленных проблем является применение *медианной* фильтрации [1,4, 7,16, 18]. Медианная фильтрация представляет собой эвристический метод обработки, ее алгоритм не является математическим решением строго сформулированной задачи.

При медианной фильтрации используется двумерное окно (апертура фильтра), обычно имеющее центральную симметрию, при этом его центр располагается в текущей точке фильтра:

$$x^* = med(y_1, y_2, \dots, y_n). \quad (3.18)$$

Рассмотрим пример. Предположим, что упорядоченная последовательность в окне размером 3x3 имеет вид: $Y = \{99, 140, 97, 150, 255, 155, 158, 99, 175\}$, где элемент 255 соответствует центру окна (x, y) . Большое значение яркости в этой точке кадра является результатом воздействия импульсной помехи. Упорядоченная по возрастанию выборка имеет вид: $\{97, 99, 99, 140, 150, 155, 158, 175, 255\}$, следовательно, на выходе медианного фильтра получаем $med = 150$. Таким образом, учет яркостей элементов окрестности при фильтрации в текущей точке привел к подавлению импульсной помехи.

Если импульсная помеха не является точечной, а занимает некоторую область, то она также может быть подавлена, если размер этой локальной области будет меньше, чем половина размера апертуры медианного фильтра. Поэтому для подавления импульсных помех, поражающих локальные участки изображения, следует увеличивать размеры апертуры медианного фильтра.

3.4.2. Усредняющие нелинейные фильтры

Усредняющие фильтры различают в зависимости от способа нахождения среднего значения входных данных [2,4].

Гармонический

$$x^*(i, j) = \frac{MN}{\sum_{i=1}^M \sum_{j=1}^N \frac{1}{y(i, j)}}. \quad (3.19)$$

Фильтр хорошо работает с импульсными шумами типа «соль» и не работает с шумами «перец».

Геометрический

Геометрическое усреднение имеет нелинейную структуру:

$$x^*(i, j) = \left[\prod_{i=1}^M \prod_{j=1}^N y(i, j) \right]^{\frac{1}{MN}}. \quad (3.20)$$

Приводит к сглаживанию изображений, ухудшению резкости. Однако при использовании геометрического усреднения отдельные объекты входного изображения искажаются в меньшей степени. Фильтр используется для подавления высокочастотного аддитивного шума и импульсного шума типа «соль».

Контргармонический

$$x^*(i, j) = \frac{\sum_{i=1}^M \sum_{j=0}^N y(i, j)^{Q+1}}{\sum_{i=1}^M \sum_{j=0}^N y(i, j)^Q}. \quad (3.21)$$

где Q – порядок фильтра.

Контргармонический фильтр устраняет импульсные шумы типа «соль» и «перец», причем при положительных значениях Q подавляются шумы «перец», а при отрицательных – соль. Неэффективен для устранения БГШ.

3.5. Морфологические фильтры

Класс морфологических фильтров оперирует с теорией множеств [2]. В двоичных изображениях множества являются подмножествами двумерного целочисленного пространства Z^2 с элементами в виде пар чисел, т. е. двумерных векторов, координаты которых указывают на черный или белый пиксель. Полутоновые изображения могут быть описаны множествами, состоящими из элементов пространства Z^3 . В этом случае две координаты элемента множества указывают координаты пикселя, а третья соответствует дискретному значению яркости.

Морфологические операции выполняются над изображением и некоторым структурным элементом или примитивом (рис. 3.5).

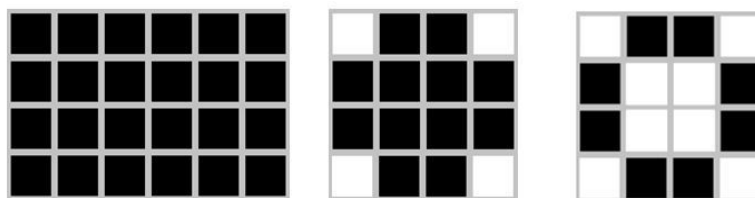


Рис. 3.5. Структурные элементы

Основные морфологические операции:

1. *Дилатация (Расширение)*. Пусть A и B – множества пространства Z^2 . Дилатация множества A по множеству B обозначается $A \oplus B$ и определяется как

$$A \oplus B = \left\{ z \mid (\hat{B})_z \cap A \neq \emptyset \right\}. \quad (3.22)$$

В основе этого соотношения лежит получение центрального отражения множества B относительно его центра и затем сдвиг полученного множества в точку z из пространства изображения A . При этом дилатация множества A по B – это множество всех таких смещений z , при которых множества A и \hat{B} совпадают по меньшей мере в одном элементе.

Дилатация может применяться для устранения разрывов линий путем их перекрытия.

2. *Эрозия (Сжатие)*. Для множеств A и B из пространства Z^2 эрозия A и B , обозначаемая $A \ominus B$, определяется как

$$A \ominus B = \{z | (B)_z \cap A \subseteq A\}. \quad (3.23)$$

То есть эрозия множества A по B – это множество всех таких точек z , при сдвиге которых множество B целиком содержится в A .

Эрозия может применяться для удаления небольших по размеру объектов, например, импульсных помех.

3. *Размыкание.* Размыкание множества A по примитиву B обозначается $A \circ B$ и определяется выражением

$$A \circ B = (A \ominus B) \oplus B. \quad (3.24)$$

Иначе говоря, размыкание множества A по примитиву B строится как эрозия A по B с последующей дилатацией по тому же примитиву. Размыкание сглаживает контуры объекта, обрывает узкие перешейки и ликвидирует выступы небольшой ширины.

4. *Замыкание.* Замыкание множества A по примитиву B обозначается $A \bullet B$ и определяется равенством

$$A \bullet B = (A \oplus B) \ominus B. \quad (3.25)$$

Замыкание также проявляет тенденцию к сглаживанию участков контуров, но в отличие от размыкания заполняет узкие разрывы, ликвидирует небольшие отверстия и заполняет промежутки контура.

Применяются морфологические фильтры для дактилоскопии, распознавания образов, нелинейной фильтрации, векторизации, обнаружения краев, подавления шума, улучшения изображений и т. д. Морфологический фильтр способен устранять импульсные шумы любой полярности и не пригоден для устранения БГШ.

К недостаткам можно отнести зависимость результата фильтрации от содержания изображения: если изображение содержит элементы, меньшие по размеру, чем структурный элемент, то эти элементы будут удалены как шум.

4. СЕГМЕНТАЦИЯ ИЗОБРАЖЕНИЙ

Сегментация изображений является предварительным этапом любой системы анализа и обработки изображений [2,19–29]. Она позволяет оперативно выделить интересующие объекты на изображении от фона и других объектов, определить размер, форму, положение объекта, преобразовать изображение к виду удобному для дальнейшей обработки автоматическими системами и эффективного решения задач более высокого уровня, таких как распознавание образов и анализ сцен. Качество сегментации изображений оказывает значительное влияние на качество финальной стадии обработки изображения. Неточная, недостаточная или избыточная сегментация может привести к возникновению ошибок на следующих этапах обработки изображения.

Один из возможных методов определения частей объектов основан на выделении контуров или границ на изображении, поскольку именно контуры являются наиболее информативными составляющими изображения.

Контурная сегментация изображений включает в себя следующие этапы обработки:

1. Выделение контуров.
2. Затравка выделенных областей (после выделения контура необходимо пометить каким-либо образом все пиксели внутри каждой из областей, ограниченных контуром).
3. Определение основных характеристик сегментов (размер и положение объекта, ориентация, средний уровень яркости, периметр и т. д.).

Существуют различные методы выделения контуров на изображении. Наиболее известными из них являются: градиентный метод, контрастный, метод вторых производных, Превитта – Кирша, Марра – ХилдERTA, фрактальный и др. [2,19-30]. В результате всех методов вычисляется величина (первая или вторая производная, контрастность), которая может принимать широкий спектр значений. Наиболее простым способом выделения контура является сравнение вычисленной величины с порогом. Если значе-

ние вычисленной величины больше порогового значения, то данная точка является точкой контура, иначе – в данной точке контур не обнаружен.

На рис. 4.1 приведена модель наклонного перепада яркости изображения, а на рис. 4.2 показаны первая и вторая производные указанного профиля яркости изображения. То есть значение первой или второй производной изображения может использоваться для обнаружения наличия перепада яркости в каждой точке изображения.

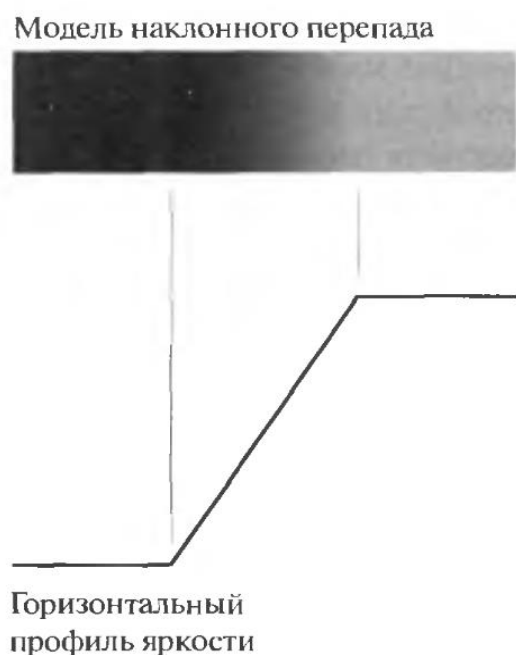


Рис. 4.1. Модель наклонного перепада яркости



Рис. 4.2. Горизонтальный профиль яркости вблизи перепада яркости, а так же первая и вторая производные этой зависимости

Качество методов выделения контуров оценивается непрерывностью и толщиной линии, низким процентом ложных и пропущенных контуров. Кроме того, метод выделения контуров должен быть простым, иметь низкую вычислительную сложность.

Наиболее общим способом нахождения контуров является обработка изображения с помощью скользящей маски. При использовании маски отклик в каждой точке изображения задаётся выражением:

$$R = w_1 z_1 + w_2 z_2 + \dots + w_9 z_9 = \sum_{i=1}^9 w_i z_i \quad (4.1)$$

где z_i – значение яркости пикселя (рис. 4.3), соответствующего коэффициенту w_i маски (рис 4.4).

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Рис. 4.3. Значение яркостей в окрестности некоторого элемента изображения

Рис. 4.4. Скользящая маска размером 3×3

Точка будет являться точкой контура, если

$$|R| \geq H, \quad (4.2)$$

где H – неотрицательный порог.

Пример маски для обнаружения точек показан на рис. 4.5.

Сумма коэффициентов маски должна равняться нулю, чтобы на областях постоянной яркости получить нулевой отклик.

-1	-1	-1
-1	8	-1
-1	-1	-1

Рис. 4.5. Маска для обнаружения точек

Пример масок для обнаружения линий приведен на рис. 4.6.

-1	-1	-1
2	2	2
-1	-1	-1

а) Горизонтальная

-1	-1	2
-1	2	-1
2	-1	-1

б) $+45^\circ$

-1	2	-1
-1	2	-1
-1	2	-1

в) Вертикальная

2	-1	-1
-1	2	-1
-1	-1	2

г) -45°

Рис. 4.6. Маски для обнаружения линий

При скольжении первой маски (а) по изображению, наиболее сильный отклик будет на горизонтальных линиях толщиной в один пиксель; вторая маска (б) дает наибольший отклик на линиях, проходящих под углом $+45^\circ$; третья (в) – на вертикальных линиях; четвертая (г) – на проходящих под углом -45° . Эти направления можно выявить и по тому признаку, что предпочтительные направления каждой из масок характеризуются большими значениями весовых коэффициентов, чем любые другие направления.

4.1. Градиентные методы. Вычисление первой производной

Вычисление первой производной цифрового изображения основано на дискретных приближениях двумерного градиента. Для каждого элемента изображения вычисляется величина градиента по следующему выражению:

$$G = \sqrt{(G_x^2 + G_y^2)}, \quad (4.3)$$

где G_x , G_y – разности яркостей пикселей в двух взаимно перпендикулярных направлениях.

Для сокращения количества вычислений используют упрощенный вариант вычисления градиента:

$$G = |G_x| + |G_y|. \quad (4.4)$$

Полученное значение величины градиента сравнивается с порогом, и делается вывод о принадлежности точки контуру.

Реализацией градиентных методов для дискретных изображений являются соответствующие градиентные операторы (маски): Робертса, Собелла, Превитта [2, 4, 19, 23].

Для вычисления градиента вычисляется операция свёртки матрицы оператора с изображением. Операция свёртки представляет собой умножение каждого элемента двумерной апертуры размерностью равной размерности матрицы оператора на ему соответствующий элемент оператора (4.1). Далее маска оператора сдвигается по строке на один пиксель. По достижении маской конца строки, происходит её переход на начало строки и сдвиг вниз по столбцам на один пиксель. Сдвиги происходят циклически, пока маска не достигнет крайнего правого нижнего положения на изображении. Начальной точкой движения является левый верхний край изображения.

Оператор Робертса

Первая производная в точке z_5 (рис. 4.3) при применении оператора Робертса вычисляется по формулам:

$$G_x = (z_9 - z_5); \quad (4.5)$$

$$G_y = (z_8 - z_6). \quad (4.6)$$

Для вычисления используются маски, приведенные на рис. 4.7:

-1	0
0	1

0	-1
1	0

Рис. 4.7. Маски оператора Робертса

Оператор Робертса требует небольших вычислительных затрат, но он чувствителен к шуму. Он даёт линии тоньше, чем другие методы выделения границ.

Метод Превитта

В методе Превитта используются маски 3×3 (рис. 4.8). Первая производная в точке z_5 (рис. 4.3) вычисляется по формулам:

$$G_x = (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3); \quad (4.7)$$

$$G_y = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7). \quad (4.8)$$

-1	-1	-1
0	0	0
1	1	1

-1	0	1
-1	0	1
-1	0	1

Рис. 4.8. Маски оператора Превитта

Оператор Собела

Реализация оператора Собела аналогична реализации метода Превитта. Различия методов – в коэффициентах масок и в выражениях их задания.

Маски оператора Собела задаются выражениями:

$$G_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3); \quad (4.9)$$

$$G_y = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7). \quad (4.10)$$

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

Рис. 4.9. Маски оператора Собела

Большие значения коэффициентов масок используются для уменьшения эффекта сглаживания за счёт придания большего веса средним точкам.

Оператор Превитта реализовать проще, чем оператор Собела, однако у последнего оператора влияние шума угловых элементов несколько меньше, что является существенным при работе с производными.

Метод Превитта-Кирша

В методе Превитта-Кирша для выделения контуров используется серия масок, где для масок Превитта: $a = 1, b = -1, c = -2$, для масок Кирша: $a = 3, b = -5, c = 0$ (рис. 4.10).

При реализации метода Превитта-Кирша операция свёртки выполняется восемь раз, поэтому данный метод является одним из самых ресурсоёмких методов сегментации. При каждой последующей свёртке изображения с маской оператора в ней происходит сдвиг элементов по часовой стрелке на один пиксел.

Изображение обрабатывается поочередно каждой из масок, и для каждой точки изображения ищется максимальное по модулю из восьми возможных значений.

$$H_1 = \begin{array}{|c|c|c|} \hline a & a & b \\ \hline a & c & b \\ \hline a & a & b \\ \hline \end{array} \quad \dots \quad H_8 = \begin{array}{|c|c|c|} \hline a & b & b \\ \hline a & c & b \\ \hline a & a & a \\ \hline \end{array}$$

Рис. 4.10. Маски для метода Превитта-Кирша

Градиентные методы в целом более защищены от воздействия шума по сравнению с методами второй производной, и, следовательно, создают более низкий процент ложных контуров.

Существенным недостатком градиентных методов является то, что неоднородности освещения, а также прочие эффекты, которые нарушают непрерывность яркостной картины, могут приводить к разрывам выделяемых контуров. Поэтому для обеспечения требуемой точности и замкнутости границы их необходимо дополнять процедурами связывания контуров, что в целом усложняет метод сегментации изображений.

4.2. Метод вторых производных (Лапласиан)

В методе второй производной контурные точки находятся, как точки пересечения нуля второй производной. Метод реализуется с помощью оператора Лапласа [2,4,19,20].

Лапласиан двумерной функции представляет собой $f(x, y)$ производную второго порядка.

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}. \quad (4.11)$$

Цифровая аппроксимация Лапласиана применительно к окрестности 3x3 может быть представлена в виде:

$$d^2 f = 4z_5 - (z_2 + z_4 + z_6 + z_8). \quad (4.12)$$

Дискретное приближение с использованием диагональных соседних элементов имеет вид:

$$d^2 f = 8z_5 - (z_1 + z_2 + z_3 + z_4 + z_6 + z_7 + z_8 + z_9). \quad (4.13)$$

Для реализации формул используются маски:

0	-1	0
-1	4	-1
0	-1	0

-1	-1	-1
-1	8	-1
-1	-1	-1

Рис. 4.11. Маски Лапласиана

Оператор Лапласа излишне чувствителен к шуму и приводит к удвоению контуров, что дает увеличение процента ложных контуров и усложняет последующую сегментацию.

4.3. Метод Лапласиана Гауссиана

Метод Лапласиана Гауссиана предполагает предварительную обработку сегментируемого изображения гауссовским сглаживающим фильтром и применение к отфильтрованному изображению оператора Лапласа [2, 4, 19, 23]. Это позволяет увеличить четкость выделения контуров и снизить влияние шума.

Рассмотрим гауссову функцию:

$$h(r) = -e^{-\frac{r^2}{2\sigma^2}}, \quad (4.14)$$

где $r^2 = x^2 + y^2$, а σ – стандартное отклонение. Свертка изображения с такой функцией приводит к его расфокусировке, причем степень расфокусировки определяется значением σ . Лапласиан функции h задается выражением:

$$\nabla^2 h = -\left(\frac{r^2 - \sigma^2}{\sigma^4}\right)e^{-\frac{r^2}{2\sigma^2}}. \quad (4.15)$$

Эту функцию называют Лапласианом Гауссиана.

Аппроксимирующая маска Лапласиана Гауссиана при $\sigma = 1,4$ имеет вид:

0	1	1	2	2	2	1	1	0
1	2	4	5	5	5	4	2	1
1	4	5	3	0	3	5	4	1
2	5	3	-12	-24	-12	3	5	2
2	5	0	-24	-40	-24	0	5	2
2	5	3	-12	-24	-12	3	5	2
1	4	5	3	0	3	5	4	1
1	2	4	5	5	5	4	2	1
0	1	1	2	2	2	1	1	0

Рис. 4.12. Маска Лапласиана Гауссиана

4.4. Метод Канни

Метод Канни предполагает выполнение нескольких последовательных шагов, первым из которых является сглаживание исходного изображения с помощью гауссовского фильтра [2]. Затем определяется величина и направление градиента в каждой точке сглаженного изображения, для чего изображение дифференцируется по двум ортогональным направлениям. Точки, которые не являются локальными максимумами величины градиента, подавляются путем обнуления величины скачка интенсивности в точке, если она не превышает величину скачка интенсивности в двух соседних точках по направлению градиента. Принадлежность точки контуру определяется путем сравнения величины интенсивности в полученных точках с двумя порогами методом гистерезиса. Согласно этому методу, точки, величина интенсивности в которых меньше нижнего порога, не являются точками контура. Точки, величина интенсивности которых выше нижнего порога присоединяются к соседним точкам, величина интенсивности в которых превышает верхний порог.

Применение двух порогов чувствительности позволяет выделить контуры более точно и полученные данным методом границы являются замкнутыми. Однако, несмотря на применение гистерезиса, низкая помехоустойчивость операции численного дифференцирования приводит к выделению ложных контуров.

5. МНОГОМЕРНЫЕ ЦЕПИ МАРКОВА ПРИ ОБРАБОТКЕ ИЗОБРАЖЕНИЙ

Развитие видеосистем требует разработки новых методов обработки многомерных изображений, представляющих собой статические изображения (2D-сигналы), видео (3D-сигналы), меняющиеся во времени томографические снимки (4D-сигналы). Для разработки таких методов необходимы многомерные математические модели, адекватные реальным изображениям и обладающие высокой вычислительной эффективностью, чтобы алгоритмы, синтезируемые на их основе, требовали небольших вычислительных ресурсов. Перспективным решением является использование в качестве таких моделей многомерных цепей Маркова с несколькими состояниями [31–39].

5.1. Многомерная марковская модель изображения

Цифровые полутоновые изображения (ЦПИ), представленные g -разрядными двоичными числами, разбиваются на g разрядных двоичных изображений (РДИ). Предполагается, что двоичные изображения представляют собой случайное марковское поле с разделимой автокорреляционной функцией. В этом случае многомерную (Q -мерную) математическую модель (ММ) двоичного изображения можно представить суперпозицией Q простых одномерных цепей Маркова с двумя равновероятными состояниями $M_1^{(l)}$ и $M_2^{(l)}$ и матрицами вероятностей переходов по каждому измерению. При этом двоичный элемент $v_D^{(l)}$ будет одновременно принадлежать Q цепям Маркова (рис. 5.1).

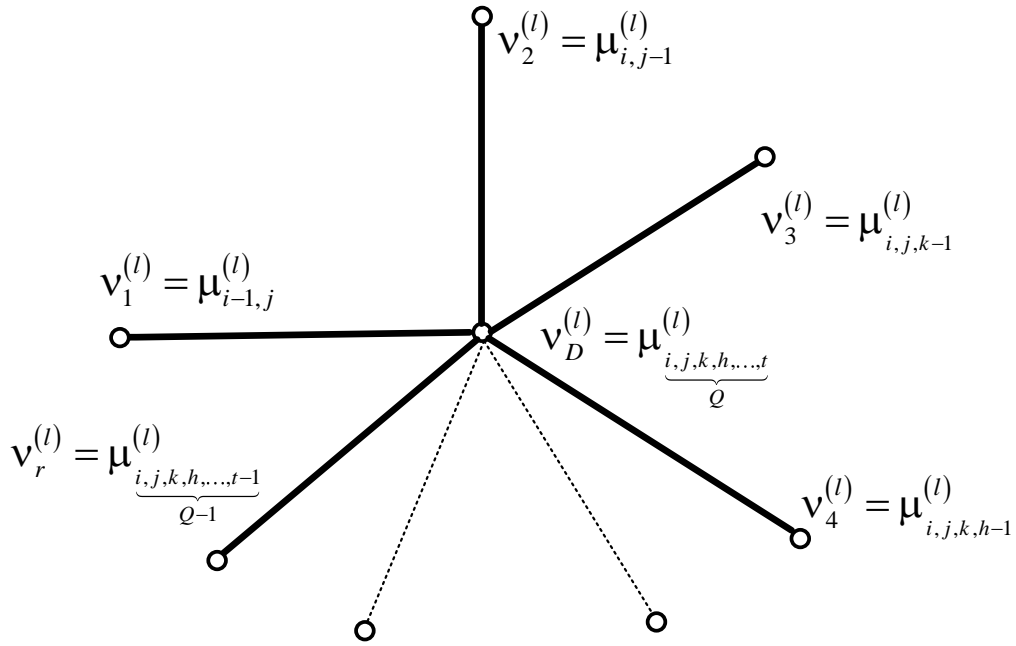


Рис. 5.1. Модель Q -мерной цепи Маркова

Для получения алгоритма формирования состояний двоичных элементов Q -мерной цепи Маркова использован энтропийный подход, который заключается в следующем. Условная энтропия элемента $v_D^{(l)}$ относительно состояний элементов окрестности $\Lambda_Q^{(l)}$ определится как разность безусловной энтропии элемента $v_D^{(l)}$ и взаимной информации полученной от элементов окрестности $\Lambda_{\underbrace{i,j,k,\dots,t-1}_Q}^{(l)}$ [32]:

$$\begin{aligned}
 H\left(v_D^{(l)} \mid v_1^{(l)}, \dots, v_t^{(l)}\right) &= H\left(v_D^{(l)}\right) - I\left(v_1^{(l)}, \dots, v_D^{(l)}\right) = \\
 &= -\log \left(\frac{P\left(v_D^{(l)}\right) \prod P\left(\underbrace{v_i^{(l)}, \dots, v_r^{(l)}}_L\right)}{\prod_{i=1}^D P\left(v_i^{(l)}\right) \prod P\left(\underbrace{v_i^{(l)}, \dots, v_r^{(l)}}_M\right)} \right) =
 \end{aligned} \tag{5.1}$$

$$= -\log \left(\frac{\prod w \left(v_D^{(l)} \mid \underbrace{v_i^{(l)}, \dots, v_r^{(l)}}_N \right)}{\prod w \left(v_D^{(l)} \mid \underbrace{v_i^{(l)}, \dots, v_r^{(l)}}_S \right)} \right), \quad Q \geq 2,$$

где $L = 2b$, $M = 2m + 1$, $N = 2k + 1$, $S = 2d$;

множества индексов для четного Q : $b = m = 1, \frac{Q}{2}$; $k = 0, \frac{Q-2}{2}$; $d = 1, \frac{Q}{2}$;

для нечетного Q : $b = 1, \frac{Q+1}{2}$; $m = 1, \frac{Q-1}{2}$; $k = 0, \frac{Q-1}{2}$; $d = 1, \frac{Q-1}{2}$.

Произведения, обозначенные в (5.1) символом \prod , вычисляются по всем возможным несовпадающим комбинациям различных подстрочных индексов Q -мерного случайного поля; $p(v_i^{(l)}, \dots, v_r^{(l)})$ – безусловные многомерные вероятности состояний двоичных элементов Q -мерного случайного поля; $p(v_i^{(l)})$ – безусловные плотности вероятностей состояний элементов; $w(v_D^{(l)} \mid v_i^{(l)}, \dots, v_r^{(l)})$ – плотности вероятностей переходов в многомерных цепях Маркова.

Плотность вероятности перехода в Q -мерной цепи Маркова $w(v_D^{(l)} \mid v_i^{(l)}, \dots, v_r^{(l)})$ можно представить через элементы многомерной матрицы вероятностей переходов (МВП) в форме:

$$w \left(v_D^{(l)} \mid \underbrace{v_i^{(l)}, \dots, v_t^{(l)}}_Q \right) = \sum_{i, \dots, k=1}^2 \pi(v_D^{(l)} = M_i^{(l)} \mid v_1^{(l)} = M_j^{(l)}; \dots; v_t^{(l)} = M_k^{(l)}) \times \delta(v_1^{(l)} - M_j^{(l)}); \dots; \delta(v_t^{(l)} - M_k^{(l)}), \quad (5.2)$$

где $\delta(\cdot)$ – дельта функция.

Q -мерная модель является универсальной и служит основой для построения моделей изображений любой размерности.

5.1.1. Двумерная марковская модель изображения

Пусть $\mu^{(l)} = \{\mu_{i,j}^{(l)}\}$ – случайное поле, заданное на прямоугольной сетке $L = \{(i, j); 1 \leq i \leq m, 1 \leq j \leq n\}$ размером $m \times n$ элементов (рис. 5.2).

Будем считать, что каждый элемент $\mu_{i,j}^{(l)}$ случайного поля $\mu^{(l)}$ принадлежит двум независимым цепям Маркова с равновероятными состояниями $M_1^{(l)}$ и $M_2^{(l)}$ и МВП по горизонтали и вертикали соответственно:

$${}^1\Pi^{(l)} = \left\| \begin{array}{cc} {}^1\pi_{11}^{(l)} & {}^1\pi_{12}^{(l)} \\ {}^1\pi_{21}^{(l)} & {}^1\pi_{22}^{(l)} \end{array} \right\|, \quad {}^2\Pi^{(l)} = \left\| \begin{array}{cc} {}^2\pi_{11}^{(l)} & {}^2\pi_{12}^{(l)} \\ {}^2\pi_{21}^{(l)} & {}^2\pi_{22}^{(l)} \end{array} \right\|. \quad (5.3)$$

Если условная зависимость l -го РДИ определена от левого верхнего сегмента, то элемент $\mu_{i,j}^{(l)}$ зависит от случайных величин только из некоторого подмножества $\Lambda_{i,j}^{(l)}$ этого сегмента, называемого окрестностью. Лучшим образом удовлетворяет условию каузальности окрестность вида:

$$\Lambda_{i,j}^{(l)} = \{\mu_{i,j-1}^{(l)}; \mu_{i-1,j}^{(l)}; \mu_{i-1,j-1}^{(l)}\}. \quad (5.4)$$

Разделим условно двумерное l -е РДИ на четыре части, каждая из которых имеет свои особенности моделирования (рис. 5.2) [32–35].

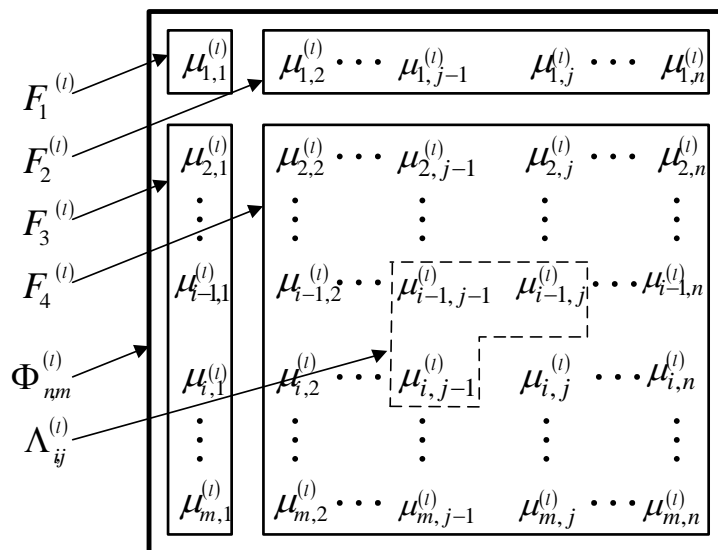


Рис. 5.2. Области l -го РДИ

На рис. 5.3 представлен фрагмент двумерного l -го РДИ, соответствующего области $F_4^{(l)}$ марковского случайного поля (рис. 5.2), где приняты обозначения $v_1^{(l)} = \mu_{i,j-1}^{(l)}$; $v_2^{(l)} = \mu_{i-1,j}^{(l)}$; $v_3^{(l)} = \mu_{i,j}^{(l)}$; $v_3'^{(l)} = \mu_{i-1,j-1}^{(l)}$.

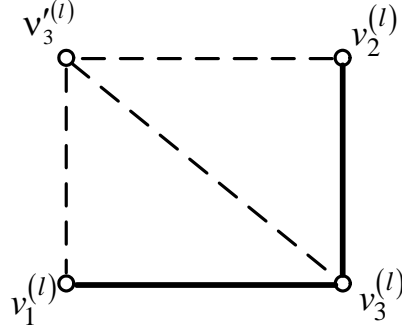


Рис. 5.3. Фрагмент области $F_4^{(l)}$ l -го с окрестностью $\Lambda_{ij}^{(l)}$

Вероятность состояния $M_1^{(l)}$ или $M_2^{(l)}$ элемента $v_3^{(l)}$ в области F_4 (рис. 5.3) полностью определяется энтропией элемента $v_3^{(l)}$ относительно состояний элементов ближайшей окрестности $\Lambda_{i,j}^{(l)}$ как разность безусловной энтропии элемента $v_3^{(l)}$ и взаимной информации между элементами $v_1^{(l)}, v_2^{(l)}, v_3'^{(l)}$ в виде:

$$\begin{aligned} H(v_3^{(l)} | v_2^{(l)}, v_1^{(l)}) &= H(v_3^{(l)}) - I(v_1^{(l)}, v_2^{(l)}, v_3'^{(l)}) = \\ &= -\log \frac{w(v_3^{(l)} | v_1^{(l)}) w(v_3^{(l)} | v_2^{(l)})}{w(v_3^{(l)} | v_1^{(l)}, v_2^{(l)})} = -\log \frac{w(v_3^{(l)} | v_1^{(l)}) w(v_3^{(l)} | v_2^{(l)})}{w(v_3^{(l)} | v_3'^{(l)})}, \end{aligned} \quad (5.5)$$

где $w(v_3^{(l)} | v_1^{(l)})$, $w(v_3^{(l)} | v_2^{(l)})$ – одномерные плотности вероятностей перехода между соседними значениями; $w(v_3^{(l)} | v_1^{(l)}, v_2^{(l)})$ – плотность вероятности перехода в двумерной цепи Маркова.

Вероятности перехода от комбинаций состояний элементов окрестности $\Lambda_{i,j}^{(l)}$ к состоянию элемента $v_3^{(l)}$ численно определяются аргументом выражения (5.6) и образуют сложную МВП вида:

$$\mathbf{\Pi} = \begin{pmatrix} \pi_{ii}^{(l)} & \pi_{ij}^{(l)} \\ \pi_{ji}^{(l)} & \pi_{jj}^{(l)} \\ \pi_{ji}^{(l)} & \pi_{jj}^{(l)} \\ \pi_{ji}^{(l)} & \pi_{jj}^{(l)} \end{pmatrix} = \begin{pmatrix} \alpha_1^{(l)} & \alpha_1'^{(l)} \\ \alpha_2^{(l)} & \alpha_2'^{(l)} \\ \alpha_3^{(l)} & \alpha_3'^{(l)} \\ \alpha_4^{(l)} & \alpha_4'^{(l)} \end{pmatrix}; i, j = \overline{1, 2}; i \neq j, l = \overline{1, g}. \quad (5.6)$$

Элементы матрицы $\mathbf{\Pi}$ (5.6) связаны с элементами матриц (5.3) следующими соотношениями

$$\begin{aligned} \alpha_1^{(l)} &= \pi_{iii}^{(l)} = \pi(v_3^{(l)} = M_1^{(l)} | v_1^{(l)} = M_1^{(l)}; v_2^{(l)} = M_1^{(l)}) = \frac{{}^1\pi_{ii}^{(l)} {}^2\pi_{ii}^{(l)}}{{}^3\pi_{ii}^{(l)}}, \\ \alpha_2^{(l)} &= \pi_{iji}^{(l)} = \pi(v_3^{(l)} = M_1^{(l)} | v_1^{(l)} = M_1^{(l)}; v_2^{(l)} = M_2^{(l)}) = \frac{{}^1\pi_{ii}^{(l)} {}^2\pi_{ij}^{(l)}}{{}^3\pi_{ij}^{(l)}}, \\ \alpha_3^{(l)} &= \pi_{jii}^{(l)} = \pi(v_3^{(l)} = M_1^{(l)} | v_1^{(l)} = M_2^{(l)}; v_2^{(l)} = M_1^{(l)}) = \frac{{}^1\pi_{ij}^{(l)} {}^2\pi_{ii}^{(l)}}{{}^3\pi_{ij}^{(l)}}, \\ \alpha_4^{(l)} &= \pi_{jji}^{(l)} = \pi(v_3^{(l)} = M_1^{(l)} | v_1^{(l)} = M_2^{(l)}; v_2^{(l)} = M_2^{(l)}) = \frac{{}^1\pi_{ij}^{(l)} {}^2\pi_{ij}^{(l)}}{{}^3\pi_{ij}^{(l)}}, \end{aligned} \quad (5.7)$$

где ${}^3\pi_{ij}^{(l)}$ ($i, j = \overline{1, 2}; i \neq j; l = \overline{1, g}$) – элемент дополнительной матрицы

$${}^3\mathbf{\Pi} = {}^1\mathbf{\Pi} \times {}^2\mathbf{\Pi} = \begin{pmatrix} {}^3\pi_{ii}^{(l)} & {}^3\pi_{ij}^{(l)} \\ {}^3\pi_{ji}^{(l)} & {}^3\pi_{jj}^{(l)} \end{pmatrix}. \quad (5.8)$$

Элементы матрицы (5.6) удовлетворяют условию нормировки, т. е.

$$\alpha_q + \alpha_q'^{(l)} = 1, \quad q = \overline{1, 4}.$$

Примем ${}^1\pi_{ii}^{(l)} = {}^2\pi_{ii}^{(l)} = 0,9$. Тогда в соответствии (5.8) получим

$$\alpha_1 = \pi(v_3^{(l)} = M_1^{(l)} | v_1^{(l)} = M_1^{(l)}; v_2^{(l)} = M_1^{(l)}) = \frac{{}^1\pi_{ii}^{(l)} {}^2\pi_{ii}^{(l)}}{{}^3\pi_{ii}^{(l)}} = \frac{0,9 \cdot 0,9}{0,82} = 0,9878;$$

$$\alpha_2^{(l)} = \pi(v_3^{(l)} = M_1^{(l)} | v_1^{(l)} = M_1^{(l)}; v_2^{(l)} = M_2^{(l)}) = \frac{{}^1\pi_{ii}^{(l)} {}^2\pi_{ij}^{(l)}}{{}^3\pi_{ij}^{(l)}} = \frac{0,9 \cdot 0,1}{0,18} = 0,5;$$

$$\alpha_3^{(l)} = \pi(v_3^{(l)} = M_1^{(l)} | v_1^{(l)} = M_2^{(l)}; v_2^{(l)} = M_1^{(l)}) = \frac{{}^1\pi_{ij}^{(l)} {}^2\pi_{ii}^{(l)}}{{}^3\pi_{ij}^{(l)}} = \frac{0,1 \cdot 0,9}{0,18} = 0,5;$$

$$\alpha_4^{(l)} = \pi(v_3^{(l)} = M_1^{(l)} | v_1^{(l)} = M_2^{(l)}; v_2^{(l)} = M_2^{(l)}) = \frac{{}^1\pi_{ij}^{(l)} {}^2\pi_{ij}^{(l)}}{{}^3\pi_{ii}^{(l)}} = \frac{0,1 \cdot 0,1}{0,82} = 0,0121.$$

Пусть ${}^1\Pi = \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix}$, ${}^2\Pi = \begin{vmatrix} {}^2\pi_{ii}^{(l)} & {}^2\pi_{ij}^{(l)} \\ {}^2\pi_{ji}^{(l)} & {}^2\pi_{jj}^{(l)} \end{vmatrix}$, тогда

$$\pi_{iii}^{(l)} = \frac{{}^1\pi_{ii}^{(l)} \cdot {}^2\pi_{ii}^{(l)}}{{}^3\pi_{ii}^{(l)}} = 1.$$

Если ${}^1\Pi = \begin{vmatrix} 0,5 & 0,5 \\ 0,5 & 0,5 \end{vmatrix}$, ${}^2\Pi = \begin{vmatrix} {}^2\pi_{ii} & {}^2\pi_{ij} \\ {}^2\pi_{ji} & {}^2\pi_{jj} \end{vmatrix}$, то

$$\pi_{iii}^{(l)} = \frac{{}^1\pi_{ii}^{(l)} \cdot {}^2\pi_{ii}^{(l)}}{{}^3\pi_{ii}^{(l)}} = \frac{0,5 \cdot {}^2\pi_{ii}^{(l)}}{0,5} = {}^2\pi_{ii}^{(l)}.$$

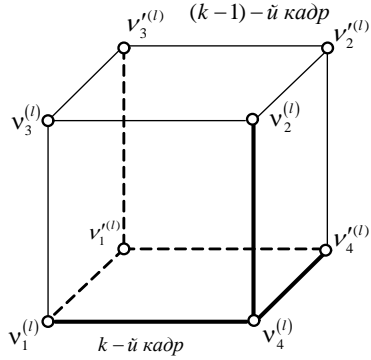
При моделирование РДИ по известным матрицам ${}^1\Pi^{(l)}$ и ${}^2\Pi^{(l)}$ вычисляются элементы $\alpha_i^{(l)}$ ($i = \overline{1,4}$), соответствующие состояниям элементов окрестности, сравниваются с выбранным случайным числом ξ , равномерно распределённом на интервале $[0,1]$. Если число $\xi \leq \alpha_i^{(l)}$, то элемент изображения v_3 принимает значение ноль, иначе – $v_3=1$. Для моделирования ЦПИ необходимо использовать g ММ РДИ с МВП соответствующих разрядов ЦПИ.

5.1.3. Трёхмерная марковская модель изображения

Каждый элемент $\mu_{i,j,k}^{(l)}$ трёхмерного случайного марковского поля принадлежит трем одномерным двоичным цепям Маркова с равновероятными состояниями $M_1^{(l)}$ и $M_2^{(l)}$ и МВП по каждой из координат:

$${}^1\Pi^{(l)} = \begin{vmatrix} {}^1\pi_{11}^{(l)} & {}^1\pi_{12}^{(l)} \\ {}^1\pi_{21}^{(l)} & {}^1\pi_{22}^{(l)} \end{vmatrix}, \quad {}^2\Pi^{(l)} = \begin{vmatrix} {}^2\pi_{11}^{(l)} & {}^2\pi_{12}^{(l)} \\ {}^2\pi_{21}^{(l)} & {}^2\pi_{22}^{(l)} \end{vmatrix}, \quad {}^4\Pi^{(l)} = \begin{vmatrix} {}^4\pi_{11}^{(l)} & {}^4\pi_{12}^{(l)} \\ {}^4\pi_{21}^{(l)} & {}^4\pi_{22}^{(l)} \end{vmatrix}. \quad (5.9)$$

Фрагмент трехмерного случайного марковского поля на ортогональной сетке с двумя равновероятными состояниями $M_1^{(l)}$ и $M_2^{(l)}$ представлен на рис. 5.4, где приняты обозначения:



$$\begin{aligned} v_1^{(l)} &= \mu_{i,j-1,k}^{(l)}, & v_2^{(l)} &= \mu_{i-1,j,k}^{(l)}, \\ v_3^{(l)} &= \mu_{i-1,j-1,k}^{(l)}, & v_4^{(l)} &= \mu_{i,j,k}^{(l)}, \\ v_1'^{(l)} &= \mu_{i,j-1,k-1}^{(l)}, & v_2'^{(l)} &= \mu_{i-1,j,k-1}^{(l)}, \\ v_3'^{(l)} &= \mu_{i-1,j-1,k-1}^{(l)}, & v_4'^{(l)} &= \mu_{i,j,k-1}^{(l)}. \end{aligned}$$

Рис. 5.4. Фрагмент трехмерного
двоичного марковского поля

Условная энтропия элемента $v_4^{(l)}$ относительно состояний элементов окрестности $\Lambda_{i,j,k} = \{v_1^{(l)}, v_2^{(l)}, v_4'^{(l)}\}$ в соответствии с формулой (5.1) будет иметь вид:

$$H(v_4^{(l)} | v_1^{(l)}, v_2^{(l)}, v_4'^{(l)}) = -\log \frac{w(v_4^{(l)} | v_1^{(l)})w(v_4^{(l)} | v_2^{(l)})w(v_4^{(l)} | v_4'^{(l)})w(v_4^{(l)} | v_1^{(l)}, v_2^{(l)}, v_4'^{(l)})}{w(v_4^{(l)} | v_1^{(l)}, v_2^{(l)})w(v_4^{(l)} | v_1^{(l)}, v_4'^{(l)})w(v_4^{(l)} | v_2^{(l)}, v_4'^{(l)})}, \quad (5.10)$$

На основании аргумента выражения (5.10) МВП Π' для различных сочетаний состояний элементов окрестности $\Lambda_{ij}^{(l)}$ имеет вид:

$$\Pi' = \begin{Bmatrix} \pi_{iii}^{(l)} & \pi_{iiij}^{(l)} \\ \pi_{iiji}^{(l)} & \pi_{ijjj}^{(l)} \\ \vdots & \vdots \\ \pi_{jjji}^{(l)} & \pi_{jjjj}^{(l)} \end{Bmatrix} = \begin{Bmatrix} \alpha_1^{(l)} & \alpha_1'^{(l)} \\ \alpha_2^{(l)} & \alpha_2'^{(l)} \\ \vdots & \vdots \\ \alpha_8^{(l)} & \alpha_8'^{(l)} \end{Bmatrix}, \quad i, j = \overline{1, 2}; \quad i \neq j. \quad (5.11)$$

Например, элементы первой строки матрицы Π' можно вычислить по формулам:

$$\alpha_1^{(l)} = \pi_{iii}^{(l)} = \frac{1 \pi_{ii}^{(l)} \cdot 2 \pi_{ii}^{(l)} \cdot 4 \pi_{ii}^{(l)} \cdot 7 \pi_{ii}^{(l)}}{3 \pi_{ii}^{(l)} \cdot 5 \pi_{ii}^{(l)} \cdot 6 \pi_{ii}^{(l)}}, \quad \alpha_1'^{(l)} = \pi_{iiij}^{(l)} = \frac{1 \pi_{ij}^{(l)} \cdot 2 \pi_{ij}^{(l)} \cdot 4 \pi_{ij}^{(l)} \cdot 7 \pi_{ii}^{(l)}}{3 \pi_{ii}^{(l)} \cdot 5 \pi_{ii}^{(l)} \cdot 6 \pi_{ii}^{(l)}}, \quad (5.12)$$

где ${}^r\pi_{ii}^{(l)} (r = \overline{1,7})$ – элементы МВП в одномерных цепях Маркова с двумя состояниями по горизонтали ${}^1\Pi^{(l)}$, вертикали ${}^2\Pi^{(l)}$, времени ${}^4\Pi^{(l)}$ и четырех сопутствующих МВП:

$${}^3\Pi^{(l)} = {}^1\Pi^{(l)} \times {}^2\Pi^{(l)}; {}^5\Pi^{(l)} = {}^1\Pi^{(l)} \cdot {}^4\Pi^{(l)}; {}^6\Pi^{(l)} = {}^2\Pi^{(l)} \cdot {}^4\Pi^{(l)}; {}^7\Pi^{(l)} = {}^3\Pi^{(l)} \cdot {}^4\Pi^{(l)}.$$

Вычисление остальных элементов матрицы $\Pi^{(l)}$ осуществляется в соответствии со значениями элементов окрестности $\Lambda_{i,j,k}^{(l)}$.

Например, если ${}^1\Pi^{(l)} = {}^2\Pi^{(l)} = {}^4\Pi^{(l)} = \begin{vmatrix} 0,9 & 0,1 \\ 0,1 & 0,9 \end{vmatrix}$, а все элементы окрестности принимают одинаковые состояния: $M_1^{(l)} = 0$ или $M_2^{(l)} = 1$, тогда вероятности того, что элемент $v_4^{(l)}$ примет состояние, совпадающее и несовпадающее с элементами окрестности $v_4^{(l)} = M_1^{(l)} = 0$, соответственно равны:

$$\pi_{iii}^{(l)} = \frac{0,9 \cdot 0,9 \cdot 0,9 \cdot 0,756}{0,82 \cdot 0,82 \cdot 0,82} = 0,999558,$$

$$\pi_{iij}^{(l)} = \frac{0,1 \cdot 0,1 \cdot 0,1 \cdot 0,244}{0,82 \cdot 0,82 \cdot 0,82} = 0,000442.$$

Алгоритм генерации искусственных трехмерных ЦПИ аналогичен алгоритму генерации двумерных ЦПИ.

Оценки ${}^r\hat{\pi}_{ii}^{(l)}$ значений элементов МВП, вычисленные по искусственным четырехмерным РДИ с погрешностью менее 3 %, совпадают с заданными, что подтверждает высокую адекватность многомерной ММ реальным процессам.

5.2. Многомерная нелинейная фильтрация изображения

При синтезе алгоритма фильтрации предполагалось, что двоичные символы разрядов ЦПИ, передаются бинарными импульсными сигналами по g – двоичным каналам независимо друг от друга в присутствии аддитивного БГШ $n(t)$ с нулевым средним и дисперсией σ_n^2 .

На основе двумерной ММ и теории фильтрации условных марковских процессов уравнение двумерной нелинейной фильтрации элементов l -го РДИ синтезируется на основе уравнения для апостериорной вероятности состояний элемента $v_3^{(l)}$ (рис. 5.3), выраженное через одномерные апостериорные вероятности и вероятности перехода состояний элементов окрестности $\Lambda_{i,j}$ к состоянию элемента $v_3^{(l)}$ [35,36]:

$$p_j(v_3^{(l)}) = c \exp\left\{f\left(M_j(v_3^{(l)})\right)\right\} \frac{p(v_1^{(l)})^1 \pi_{ij}^{(l)} p(v_2^{(l)})^2 \pi_{ij}^{(l)}}{p(v_3^{(l)})^3 \pi_{ij}^{(l)}}, \quad (5.13)$$

где c – коэффициент нормировки,

$f\left(M_j(v_3^{(l)})\right)$ – логарифм функции правдоподобия элемента $v_3^{(l)}$ l -го РДИ.

Разделив уравнение (5.13) при $j=1$ на уравнение при $j=2$ и прологарифмировав слева и справа, получим систему из g рекуррентных уравнений нелинейной фильтрации элементов l -го РДИ [35,36]:

$$\begin{aligned} u(v_3^{(l)}) = & \left[f\left(M_1(v_3^{(l)})\right) - f\left(M_2(v_3^{(l)})\right) \right] + \\ & + u(v_1^{(l)}) + z_1\left(u(v_1^{(l)}), {}^1\pi_{ij}^{(l)}\right) + u(v_2^{(l)}) + z_2\left(u(v_2^{(l)}), {}^2\pi_{ij}^{(l)}\right) - \\ & - u(v_3^{(l)}) - z_3\left(u(v_3^{(l)}), {}^3\pi_{ij}^{(l)}\right) > H, \end{aligned} \quad (5.14)$$

где $u(v_3^{(l)}) = \ln \frac{p_1(v_3^{(l)})}{p_2(v_3^{(l)})}$ – логарифм отношения апостериорных вероятностей значений двоичных элементов l -го РДИ в элементе $v_3^{(l)}$; $[f(M_1(v_3^{(l)})) - f(M_2(v_3^{(l)}))]$ – разность логарифмов функций правдоподобия состояний двоичных элементов l -го РДИ в элементе $v_3^{(l)}$; H – порог, выбранный в соответствии с критерием идеального наблюдателя (для данного алгоритма $H = 0$);

$$z_r(\cdot) = \ln \frac{{}^r \pi_{ii} + {}^r \pi_{ji} \exp\{-u^{(l)}(v_q^{(l)})\}}{{}^r \pi_{jj} + {}^r \pi_{ij} \exp\{u^{(l)}(v_q^{(l)})\}}; \quad q = \overline{1,3}; r = \overline{1,3}; \quad (5.15)$$

Приемное устройство (рис.5.5) состоит из детектора – $(D^{(l)})$, вычисляющего разность логарифмов функций правдоподобия состояний дискретного параметра двоичных сигналов $[f(M_1(v_3^{(l)})) - f(M_2(v_3^{(l)}))]$; нелинейного фильтра (НФ), включающего в себя сумматор – $(\Sigma^{(l)})$, ячейки памяти для хранения значений элементов изображения окрестности $\Lambda_{i,j,k}^{(l)} = \{v_1^{(l)}, v_2^{(l)}, v_3^{(l)}\}$ – $(\Pi_1^{(l)})$ и значений элементов матриц вероятностей перехода ${}^r \pi_{ij}^{(l)}$, $r = \overline{1,3}$ – $(\Pi_2^{(l)})$, трех колец обратной связи, каждое из которых содержит блок, реализующий нелинейную функцию $z_r(\cdot)$, $(r = \overline{1,3})$; порогового устройства (ПУ).

При формировании логарифма отношения апостериорной вероятности в $(k+1)$ -м такте $u(v_3^{(l)})$, входные данные, определяемые первым слагаемым в уравнении (5.14) складываются в сумматоре $\Sigma^{(l)}$ с ранее вычисленными данными об элементах окрестности $\Lambda_{i,j}$ l -го РДИ и вычислен-

ными значениями нелинейной функции $z_r(\cdot)$, которая содержит в себе все априорные данные о фильтруемом процессе. За счет априорных сведений о фильтруемом процессе, знания об истинном значении параметра $\mu_{k+1}^{(l)}$ в принятом сигнале увеличиваются, а средняя вероятность ошибочных решений $p_{ош}$ на выходе приемного устройства будет уменьшаться.

В случае статистической независимости импульсных сигналов ($r\pi_{ij}^{(l)} = 0,5$) функция

$$z_r(\cdot) = -u(v_r)$$

и логарифм отношения апостериорных вероятностей будет определяться только разностью логарифмов функций правдоподобия.

$$u(v_3^{(l)}) = \left[f(M_1(v_3^{(l)})) - f(M_2(v_3^{(l)})) \right].$$

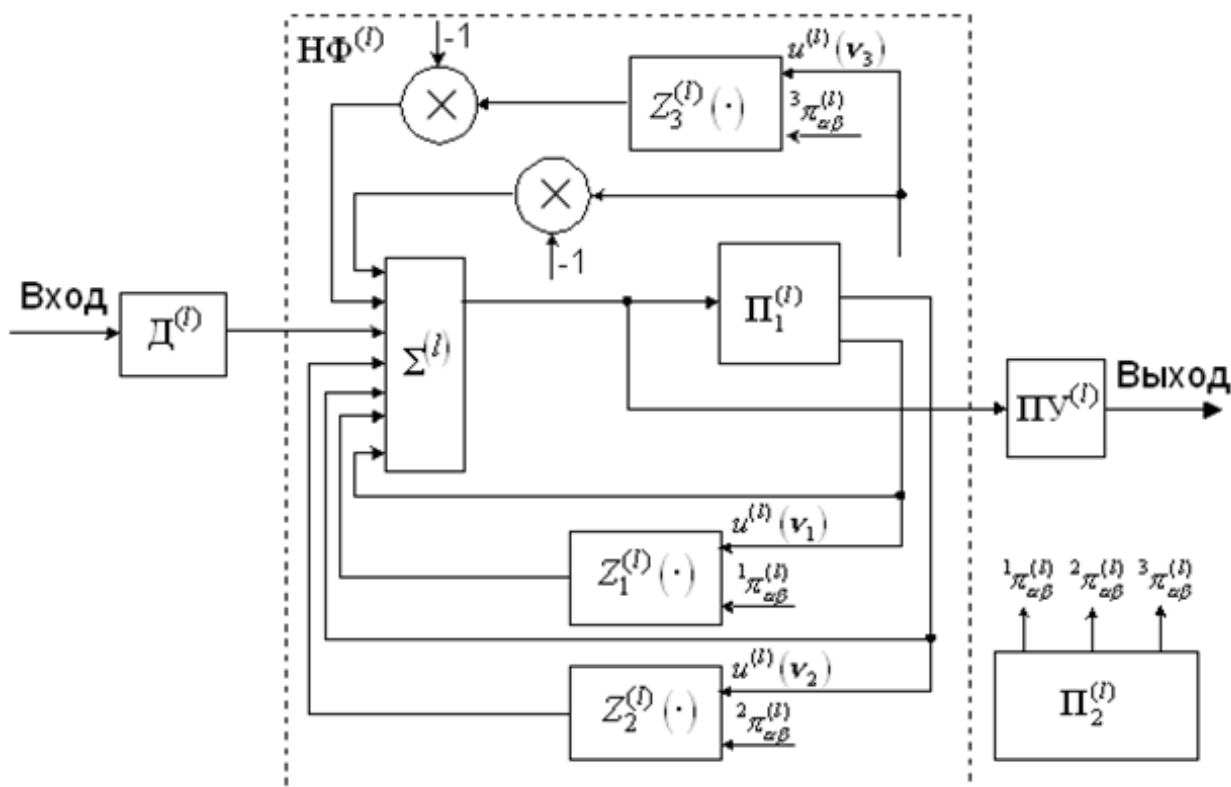


Рис. 5.5. Устройство нелинейной фильтрации

При $r\pi_{ij}^{(l)} = 0,5$ обратные связи в устройстве разрываются, и решение о наличии в принятом сигнале того или иного состояния дискретного параметра сигнала $\mu_{ijk}^{(l)}$ принимается в каждом такте на основе единичного замера.

Если $r\pi_{ij}^{(l)} = 1$, когда передаются импульсы одного знака $M_1^{(l)}$ или $M_2^{(l)}$, то функция $z_r(\cdot) = 0$ и в сумматоре $\Sigma^{(l)}$ осуществляется «чистое» накопление поступающих на вход приемного устройства данных.

Аналогично, используя, трехмерную математическую модель, можно получить уравнение фильтрации последовательности изображений (рис. 5.4) [35,36]:

$$\begin{aligned}
 u(v_4^{(l)}) = & \left[f(M_1(v_4^{(l)})) - f(M_2(v_4^{(l)})) \right] + u(v_1^{(l)}) + z_1 \left[u(v_1^{(l)}), {}^1\pi_{ij}^{(l)} \right] + u(v_2^{(l)}) + \\
 & + z_2 \left[u(v_2^{(l)}), {}^2\pi_{ij}^{(l)} \right] + u(v_4'^{(l)}) + z_4 \left[u(v_4'^{(l)}), {}^4\pi_{ij}^{(l)} \right] + u(v_3'^{(l)}) + z_7 \left[u(v_3'^{(l)}), {}^7\pi_{ij}^{(l)} \right] - \\
 & - u(v_3^{(l)}) - z_3 \left[u(v_3^{(l)}), {}^3\pi_{ij}^{(l)} \right] - u(v_1'^{(l)}) - z_5 \left[u(v_1'^{(l)}), {}^5\pi_{ij}^{(l)} \right] - u(v_2'^{(l)}) - z_6 \left[u(v_2'^{(l)}), {}^6\pi_{ij}^{(l)} \right] \geq H,
 \end{aligned} \tag{5.16}$$

где $u(v_4^{(l)}) = \ln \left(p(M_1(v_4^{(l)})) / p(M_2(v_4^{(l)})) \right)$ – логарифм отношения апостериорных вероятностей состояния фильтруемого элемента $v_4^{(l)}$ l -го РДИ; $\left[f(M_1(v_4^{(l)})) - f(M_2(v_4^{(l)})) \right]$ – разность логарифмов функций правдоподобия состояний элемента v_4 ;

$$\begin{aligned}
 z_r(\cdot) = \ln \frac{{}^r\pi_{ii}^{(l)} + {}^r\pi_{jj}^{(l)} \exp\{-u(v_q^{(l)})\}}{{}^r\pi_{jj}^{(l)} + {}^r\pi_{ii}^{(l)} \exp\{u(v_q^{(l)})\}}; \quad q = \overline{1,3}; r = \overline{1,3}; \\
 z_r(\cdot) = \ln \frac{{}^r\pi_{ii}^{(l)} + {}^r\pi_{jj}^{(l)} \exp\{-u(v_q'^{(l)})\}}{{}^r\pi_{jj}^{(l)} + {}^r\pi_{ii}^{(l)} \exp\{u(v_q'^{(l)})\}}; \quad q = \overline{1,4}; r = \overline{4,7}.
 \end{aligned} \tag{5.17}$$

На рис. 5.6 показан пример нелинейной фильтрации видеопоследовательности «Самолет» из 5 кадров размером 1489×997 . На рис. 5.6а приведен незашумленный первый кадр видеопоследовательности, на рис. 5.6б – тот же кадр, зашумленный БГШ при $\rho_{\text{вх}}^2 = -9$ дБ. На рис. 5.6в,г представлены отфильтрованные 1-й и 5-й кадры видеопоследовательности.



а)



б)



в)



г)

Рис. 5.6. Нелинейная фильтрация видеопоследовательности «Самолет»

Выигрыш в помехоустойчивости при фильтрации видеопоследовательности по сравнению с фильтрацией статических ЦПИ в диапазоне отношений сигнал-шум по мощности $\rho_{\text{вх}}^2 = -12 \dots 0$ дБ составил 6-8 дБ.

5.3. Сегментация изображений на основе цепей Маркова

5.3.1. Сегментация изображений на основе выделения контуров

Пусть ЦПИ представлено набором из g РДИ (рис. 5.7). Поскольку старшему разряду ЦПИ соответствует половина уровней яркости, то основные детальные области можно выделить по старшему разряду (8-му). Для выделения менее контрастных областей необходимо выделить контуры на следующих РДИ (7-м, 6-м или 5-м). Контурное изображение в этом случае будет представлять сумму контурных изображений нескольких разрядов.

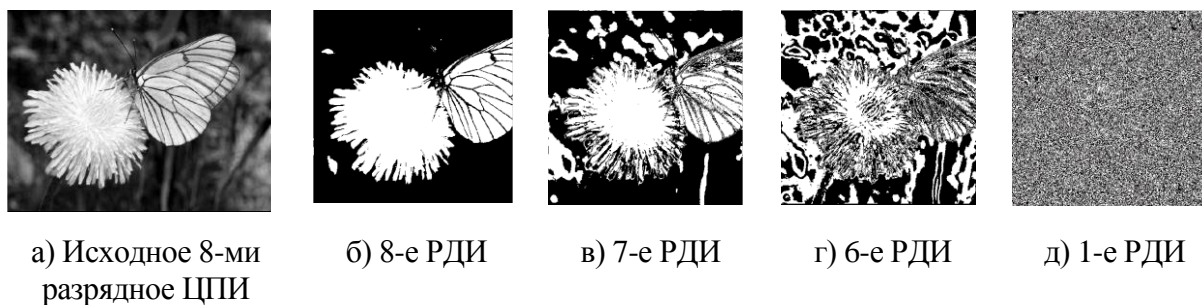


Рис. 5.7. Представление ЦПИ набором РДИ

Представление РДИ двумерной ММ (5.5) позволило решить задачу выделения контуров объектов интереса на изображениях [37].

Для выделения контуров вычисляется величина количество информации в каждом элементе РДИ в соответствии с различными сочетаниями элементов окрестности (5.5–5.7).

Количество информации в элементе РДИ будет минимально, если окрестные элементы $\nu_1^{(l)}$, $\nu_2^{(l)}$ имеют состояния одинаковые с $\nu_3^{(l)}$ (рис.5.3).

В случае появления на РДИ областей другой яркости, на границе области один или два окрестных элемента будут иметь разные с $\nu_3^{(l)}$ состояния, и количество информации в элементе $\nu_3^{(l)}$ увеличивается. Сравнивая значение вычисленной величины количества информации в элементе изображения с порогом, определяем принадлежать данной точки контуру.

Значение порога h вычисляют для каждого РДИ с учетом вычисленного минимального количества информации и количества информации, когда хотя бы один из элементов окрестности будет иметь другое состояние:

$$h = \frac{I(v_3^{(l)} = M_i^{(l)} | v_1^{(l)} = M_i^{(l)}, v_2^{(l)} = M_i^{(l)}) + I(v_3^{(l)} = M_i^{(l)} | v_1^{(l)} = M_i^{(l)}, v_2^{(l)} = M_j^{(l)})}{2}. \quad (5.18)$$

Например, для старшего (восьмого) РДИ при ${}^1\pi_{ii}^{(l)} = {}^2\pi_{ii}^{(l)} = 0,95$, если окрестные элементы $v_1^{(l)}$ и $v_2^{(l)}$ имеют состояния одинаковые с $v_3^{(l)}$, то количество информации в элементе РДИ будет минимально:

$$I(v_3^{(l)} | v_1^{(l)}, v_2^{(l)}) = -\log\left(\frac{{}^1\pi_{ii}^{(l)} {}^2\pi_{ii}^{(l)}}{{}^3\pi_{ii}^{(l)}}\right) = -\log\left(\frac{0,95^2}{0,905}\right) = 0,004 \text{ бит.}$$

При появлении границы на РДИ, в случае, если только один из окрестных элементов совпадает по знаку с элементом $v_3^{(l)}$, то количество информации в элементе РДИ равно:

$$I(v_3^{(l)} | v_1^{(l)}, v_2^{(l)}) = -\log\left(\frac{{}^1\pi_{ii}^{(l)} {}^2\pi_{ij}^{(l)}}{{}^3\pi_{ij}^{(l)}}\right) = -\log\left(\frac{0,95 \cdot 0,05}{0,095}\right) = 1 \text{ бит,}$$

а значение порога – $h = \frac{0,004 + 1}{2} \approx 0,5$.

После выделения контуров задается диапазон значений яркости, которые может принимать объект интереса. Сравнивая контурное изображение и исходное полутоновое, определяем принадлежность i -го пиксела области объекта. Если значение яркости пиксела на исходном полутоновом изображении попадает в заданный диапазон яркости, то i -й пиксел на контурном изображении является элементом объекта интереса и ему присваивается среднее значение яркости. Иначе он является элементом фона и ему присваивается значение, соответствующее белому цвету.

Задав несколько диапазонов яркости, можно выделить несколько различных сегментов на контурном изображении, которые закрашиваются разными цветами. Для типовых изображений, на которых объекты интере-

са имеют близкую по значению яркость, число РДИ с выделенными контурами и диапазон значений яркости будут принимать постоянные значения.

На рис.5.8 в приведен пример выделения нефтяных разливов по контурному изображению (рис.5.8 б), полученному по трем старшим РДИ ЦПИ (рис.5.8 а), по которому можно определить наличие нефтяных разливов, а также их площадь и форму. На изображении, приведенном на рис.5.8 г дополнительно выделены контуры островов, относительно которых можно отследить движение нефтяных пятен на водной поверхности.

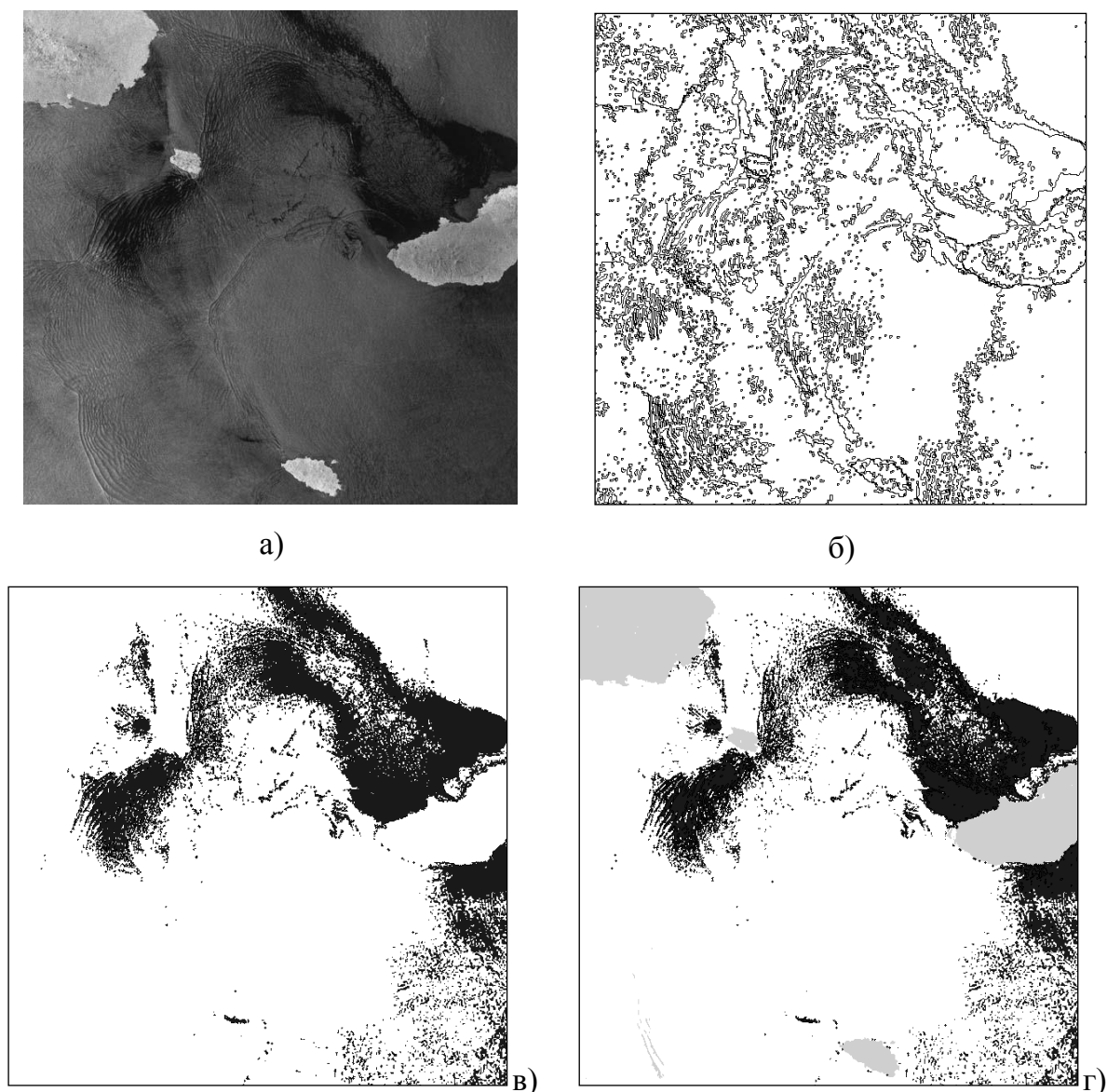


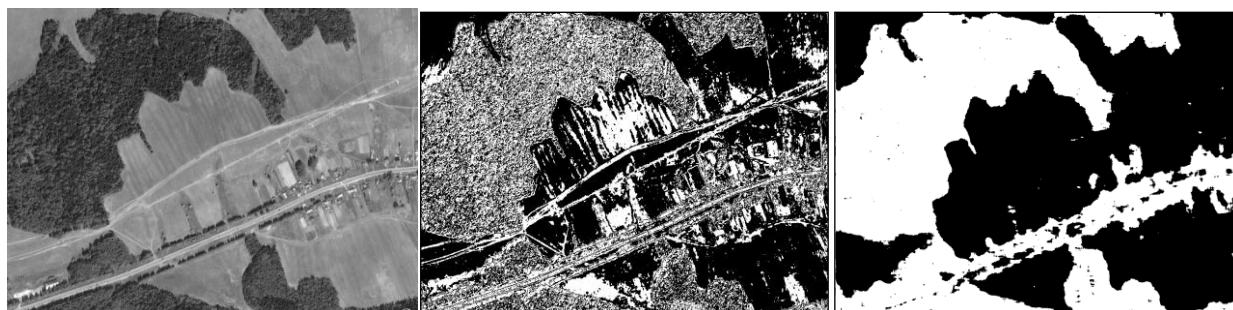
Рис. 5.8. Пример выделения нефтяных разливов по контурному изображению

5.3.2. Текстурная сегментация изображений

Часто естественные сцены, такие как леса, поля, лишены существенных деталей и не имеют четко выраженных границ. В этом случае необходимо использовать идентификацию областей изображений на основе анализа их текстуры. В качестве текстуры могут быть использованы марковские случайные поля. Так как однородная двумерная цепь Маркова полностью определяется матрицами вероятностей переходов, то в качестве текстурного признака предлагается использовать оценку вероятностей перехода в двумерной цепи Маркова [38].

В пределах сканирующего окна, используя элементы множества $\psi = \{v_1^{(l)}, v_2^{(l)}, v_3^{(l)}, v_4^{(l)}\}$, вычисляется оценка $\hat{\pi}_{iii}^{(l)}$ вероятности перехода для двумерной цепи Маркова и усредненная оценка $\tilde{\pi}_{iii}^{(l)}$, соответствующая центральному элементу окна. На каждом последующем шаге окно перемещается на один элемент и процесс повторяется. Для выделения областей с разной текстурой выполнялось сравнение вычисленной оценки $\tilde{\pi}_{iii}^{(l)}$ с пороговым значением.

На рис. 5.9 приведен пример сегментации спутникового изображения (размер 1024×1024), на котором эффективно выделена область леса. Сегментация осуществлялась по РДИ шестого разряда ЦПИ.



а) ЦПИ

б) РДИ шестого разряда
ЦПИ

в) сегментированное
изображение

Рис.5.9. Пример текстурной сегментации спутникового изображения

5.4. Сжатие изображений

На основе многомерной математической модели (5.1) разработан метод сжатия видеоизображений [39], позволяющий снизить вычислительные ресурсы на реализацию при соотношении качество/степень сжатия практически не уступающем известным алгоритмам (JPEG).

Представление разрядных двоичных изображений двумерной цепью Маркова с двумя равновероятными состояниями позволяет часть элементов не передавать по каналу связи, т. к. они могут быть восстановлены с учетом корреляционных связей по принятым окрестным элементам. Изображения разбивают на блоки. На рис. 5.10 представлен фрагмент двоичного изображения из девяти элементов, в котором пунктирные линии указывают на наличие статистических связей между элементами РДИ, \circ , \square – элементы изображения, передаваемые и не передаваемые по каналу связи, соответственно.

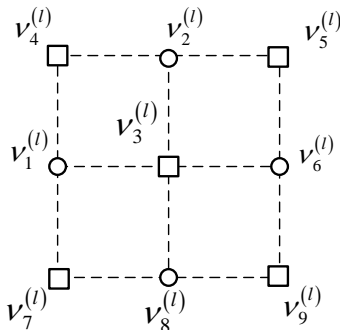


Рис. 5.10. Фрагмент изображения

На фрагменте изображения (рис 5.10) элементы $v_3^{(l)}$, $v_4^{(l)}$, $v_5^{(l)}$, $v_7^{(l)}$, $v_9^{(l)}$, с учетом корреляционных связей между элементами РДИ, можно восстановить по окрестным элементам $v_1^{(l)}$, $v_2^{(l)}$, $v_6^{(l)}$, $v_8^{(l)}$, поэтому они в канал связи не передаются.

Другой характерной особенностью ЦПИ является то, что в соседних РДИ имеются большие области, в которых элементы повторяются или переходят в противоположные (рис. 5.7). Сжатие, в этом случае, можно обеспечить за счет замены повторяющихся или инверсных блоков в соседних РДИ более короткими кодовыми комбинациями. Учитывая, указанные особенности изображения разработан алгоритм сжатия статических и динамических изображений.

Восстанавливают элементы статических изображений по выражениям для двумерной модели, а динамических – по выражениям для трехмерной модели с учетом вероятностей переходов и известных окрестных элементов.

На рис. 5.11 показаны восстановленные ЦПИ с помощью предложенного метода и JPEG и вычисленные характеристики побитовой СКО, ПОСШ, K_c .



а) исходное ЦПИ

б) восстановленное разработанным методом

$K_c = 2,53$; СКО=0,113;

ПОСШ=36,0, дБ

в) восстановленное JPEG

$K_c = 2,54$; СКО=0,184;

ПОСШ=37,6, дБ

Рис. 5.11. Примеры восстановленных изображений

При близких по значению коэффициентах сжатия разработанный метод превышает метод JPEG по критерию СКО в 1,6...2,6 раза и незначительно уступает по критерию ПОСШ при значительно меньших вычислительных затратах. Увеличение сжатия ЦПИ, без существенного снижения визуального качества, возможно за счет потерь в младших РДИ.

6. ЗАДАНИЯ К ЛАБОРАТОРНЫМ И ПРАКТИЧЕСКИМ РАБОТАМ

6.1. Практическое занятие 1.

Представление цифрового изображения

ЦЕЛЬ РАБОТЫ: Знакомство с представлением изображений в цветовых пространствах RGB и YCbCr в среде Matlab.

ЗАДАНИЕ

1. Ознакомиться с представлением изображений в цветовых пространствах RGB и YCbCr (глава 1).

2. Изображения в системе RGB (в среде Matlab) разбить на три полутоновых изображения, соответствующих цветовым компонентам R, G, B.

3. Выполнить конвертирование изображения из системы RGB в YCbCr.

4. Изображения в системе YCbCr разбить на три полутоновых изображения, соответствующих яркостной компоненте Y и двум цветоразностным – Cb и Cr.

5. Представить полученные полутоновые изображения набором g разрядных двоичных изображений (битовых плоскостей).

6. Вычислить вероятности переходов между элементами изображений по горизонтали и вертикали в каждом РДИ одной цветовой компоненты (R, G или B) и одной цветоразностной компоненты (Cb или Cr). Построить графики вероятностей переходов от номера разряда цифрового полутонового изображения.

7. В отчет включить:

- изображения цветовых компонент в системах RGB и YCbCr;

- битовые плоскости соответствующие одной цветовой компоненте и одной цветоразностной;
- графики вероятностей переходов по горизонтали и вертикали изображения;
- ВЫВОДЫ.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какие цвета называются основными?
2. Что представляет собой цветовая модель RGB?
3. Что представляет собой цветовая модель YCbCr?
4. Изображение, представленное в какой цветовой системе, может иметь больший коэффициент сжатия. Почему?
5. Как определить статистическую избыточность в изображении?
6. Битовым плоскостям каких разрядов полутонового изображения будет присуща большая статистическая избыточность?
7. Яркостной или цветоразностным компонентам будет присуща большая статистическая избыточность? Как можно учесть это свойство при обработке изображения?
8. Сколько бит информации потребуется при передаче группы из 4 пикселей для форматов сэмплирования: 4:4:4; 4:2:2; 4:2:0.

6.2. Практическое занятие 2. Статистическое сжатие изображений

ЦЕЛЬ РАБОТЫ: Изучить методы статистического сжатия изображений и получить практические навыки кодирования изображений

ЗАДАНИЕ

1. Постройте код Хаффмана для источника символов:
 - a) a b a b c d d a a e;

б) a b b b c d d a a e;

в) a b c d e f g h i a b c d e f g h a b c.

Вычислить среднюю длину для каждого кода и сравнить ее с энтропией сообщения.

2. Даны вектора движения (-2,-1,0,+1,+2) с вероятностями (0,15; 0,15; 0,35; 0,25; 0,15). Закодировать последовательность векторов (0, -1, 2) кодом Хаффмана.

3. Закодировать последовательности элементов восьмиразрядного изображения кодом LZW:

а) 9,9,9,5,5,5,0,0,0,0,0,5,0,5,0,0;

б) 7, 7, 7, 7, 0, 0, 0, 0, 0, 0, 3, 3, 0, 3, 0, 0, 3.

в) 1, 1, 3, 3, 3, 1, 3, 1, 1, 3, 3, 3, 1, 3, 1, 3, 3.

Вычислить коэффициент сжатия.

4. Последовательность на выходе кодера Лемпеля-Зива-Велча:

а) 8, 258, 10, 10, 9, 262, 1, 264, 265, 1;

б) 0, 258, 1, 260, 1, 2, 263, 263, 3, 3;

в) 3, 258, 258, 2, 260, 2, 261, 3.

Определить последовательность символов на входе кодера.

5. Закодировать последовательности элементов восьмиразрядного изображения арифметическим кодом:

а) 9,9,9,5;

б) 7, 7, 7, 0, 0;

в) 1, 1, 3, 3, 3.

6. Арифметическое кодирование использовано для кодирования последовательности трех букв с вероятностями $p(a) = 1/3$ и $p(b) = 2/3$.

Кодовое слово на выходе арифметического декодера имеет вид:

а) 101;

б) 011;

в) 11.

Определить последовательность на входе кодера.

6.3. Практическое занятие 3. Обработка изображений с помощью дискретного косинусного преобразования

ЦЕЛЬ РАБОТЫ: Получение практических навыков по обработке изображений с помощью дискретно-косинусного преобразования в среде Matlab.

ЗАДАНИЕ

Моделирование выполняется в среде Matlab.

1. Ознакомиться с принципами дискретно-косинусного преобразования (раздел 2.3.1).
2. Задать матрицу значений пикселей X размером 4×4 ($x_i = \overline{0, 255}$) (пример в разделе 2.3.1).
3. Выполнить прямое ДКП и вычислить матрицу Y .
4. Выполнить обратное ДКП и сравнить полученную матрицу с исходной.
5. Применить прямое ДКП к полутоновому изображению.
6. Выполнить квантование коэффициентов. Для этого в матрице Y (размером 4×4 или 8×8) оставить наиболее значимые коэффициенты (например, пять, пятнадцать, тридцать коэффициентов), а остальные заменить нулями.
7. Выполнить обратное ДКП и восстановить изображение с разной степенью качества.
8. Сравнить восстановленные изображения с исходными. Оценить искажения визуально и по критериям: СКО, PSNR, SSIM. (Для вычисления критериев можно воспользоваться программой, разработанной на каф. РЭС «Imqs»).
9. В отчет включить:
 - матрицу исходных значений пикселей X размером 4×4 , матрицу коэффициентов Y , матрицу, полученную после обратного DCT.

- три примера исходного и восстановленных изображений с разной степенью качества и вычисленными критериями СКО, PSNR, SSIM.
- выводы (оценить искажения).

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. В каких случаях целесообразно применять дискретно-косинусное преобразование изображений?
2. Как вычисляются коэффициенты прямого ДКП?
3. Каковы особенности распределения коэффициентов преобразования ДКП блока изображения?
4. Какие искажения возникают в изображениях после ДКП и квантования?
5. Чем вызвано использование в алгоритме JPEG зигзаг-сканирования?
6. За счет чего можно добиться большего коэффициента сжатия изображения при неизменных искажениях?
7. Какие из вычисленных критериев позволяют наиболее точно оценить визуальное качество восстановленных изображений?

6.4. Практическое занятие 4. Обработка изображений с помощью дискретного вейвлет-преобразования

ЦЕЛЬ РАБОТЫ: Получение практических навыков по обработке изображений с помощью дискретного вейвлет-преобразования в среде Matlab.

ЗАДАНИЕ

Моделирование выполняется в среде Matlab.

1. Ознакомиться с принципами дискретного вейвлет-преобразования (раздел 2.3.2).
2. Загрузить тестовое изображение в программу Matlab.

3. Выбрать вейвлет-функцию по согласованию с преподавателем.
4. К исходному изображению применить одноуровневое двумерное ДВП и разбить изображение на 4 подполосы (LL, LH, HL, HH). Визуализировать изображения полученных разложений.
5. Выполнить операцию квантования коэффициентов ДВП, в соответствии с которой оставить девять (8–5) старших двоичных разрядов коэффициентов ДВП, а остальные, небольшие по значению коэффициенты ДВП, обнулить.
6. Выполнить обратное ДВП. Визуализировать изображение.
7. Оценить качество восстановленного изображения визуально и по критериям: СКО, PSNR, SSIM с помощью программы «Imqс».
8. К LL-области применить повторно двумерное вейвлет-разложение и сформировать четыре новых субполосных изображения. Выполнить квантование коэффициентов, обратное ДВП и оценить качество восстановленного изображения.
9. Добавить к исходному изображению БГШ.
10. Выполнить прямое ДВП, квантование и обратное ДВП. Оценить качество полученного изображения.
10. В отчет включить:
 - примеры исходного и восстановленных изображений;
 - вычисленные критерии СКО, PSNR, SSIM;
 - выводы.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Как вычисляются вейвлет-коэффициенты?
2. Какова особенность распределения двумерных вейвлет-коэффициентов?
3. Какие этапы включает в себя ДВП?
4. Какие вейвлеты гарантируют точную реконструкцию сигналов?
5. Перечислите критерии выбора типа вейвлета.
6. Какие искажения возникают в изображениях после ДВП и квантования?

7. Какие из вычисленных критериев позволяют наиболее точно оценить визуальное качество восстановленных изображений?

6.5. Практическое занятие 5. Формирование искусственных изображений на основе двумерной математической модели

ЦЕЛЬ РАБОТЫ: Сформировать искусственное двоичное изображение на основе математической модели двумерной цепи Маркова и оценить адекватность сформированного изображения реальному.

Моделирование выполняется в среде Matlab.

1. Ознакомиться с математическими моделями на основе многомерных цепей Маркова (раздел 5.1).
2. Задать размер изображения.
3. По заданным матрицам вероятностей переходов между элементами изображений по горизонтали ${}^1\Pi^{(l)}$ и вертикали ${}^2\Pi^{(l)}$ (табл.6.1) вычислить матрицу ${}^3\Pi^{(l)} = {}^1\Pi^{(l)} \times {}^2\Pi^{(l)}$ и Π (5.7).

Таблица 6.1

В-нт	1	2	3	4	5	6	7	8	9	10
${}^1\Pi^{(l)}$	0,9	0,85	0,7	0,75	0,95	0,5	1	0,8	0,95	0,7
${}^2\Pi^{(l)}$	0,9	0,85	0,7	0,75	0,95	1	0,5	0,8	0,85	0,9

Элементы $\alpha_q^{(l)}$ и $\alpha'_q{}^{(l)}$ ($q = \overline{1,4}$) в матрице Π соответствуют следующим состояниям элементов окрестности (рис. 5.3):

$\alpha_q^{(l)}$	$v_1^{(l)} v_2^{(l)} \rightarrow v_3^{(l)}$	$\alpha'_q{}^{(l)}$	$v_1^{(l)} v_2^{(l)} \rightarrow v_3^{(l)}$
$\alpha_1^{(l)}$	00 → 0	$\alpha'_1{}^{(l)}$	00 → 1
$\alpha_2^{(l)}$	01 → 0	$\alpha'_2{}^{(l)}$	01 → 1
$\alpha_3^{(l)}$	10 → 0	$\alpha'_3{}^{(l)}$	10 → 1
$\alpha_4^{(l)}$	11 → 0	$\alpha'_4{}^{(l)}$	11 → 1

т. е. $\alpha_1^{(l)} = \alpha'_4{}^{(l)}$, $\alpha_2^{(l)} = \alpha'_3{}^{(l)}$, $\alpha_3^{(l)} = \alpha'_2{}^{(l)}$ и $\alpha_4^{(l)} = \alpha'_1{}^{(l)}$.

Элементы $\alpha_q^{(l)}$ и $\alpha_q'^{(l)}$ вычисляются по формулам (5.8).

4. Берется случайное число ξ , равномерно распределённом на интервале $[0, 1]$.

5. Из первого столбца матрицы Π выбирается элемент $\alpha_q^{(l)}$, соответствующий значениям элементов окрестности $\Lambda_{i,j}^{(l)}$.

6. Если число $\xi \leq \alpha_i^{(l)}$, то элемент изображения – v_3 принимает значение ноль, иначе – $v_3=1$.

7. По сформированным изображениям вычислить оценки вероятности переходов по горизонтали и вертикали и сравнить с заданными.

8. В отчет включить:

- сформированное искусственное изображение;
- вычисленные оценки вероятности переходов по горизонтали и вертикали;
- выводы.

КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ

1. Как оценить адекватность искусственного изображения реальному?

2. Используя выражение (5.1), напишите формулу условной энтропии для четырехмерной математической модели ($Q = 4$).

3. Определите количество основных и вспомогательных одномерных матриц вероятностей переходов в четырехмерной математической модели.

4. Напишите выражения для вычисления элементов матрицы Π (5.11) трехмерной модели для случаев:

а) $v_1 = 0, v_2 = 1, v_3 = 0, v_4 = 0,$

б) $v_1 = 0, v_2 = 1, v_3 = 0, v_4 = 1,$

в) $v_1 = 0, v_2 = 1, v_3 = 1, v_4 = 1,$

г) $v_1 = 1, v_2 = 1, v_3 = 0, v_4 = 1.$

6.6. Практическое занятие 6. Восстановление изображений по отдельным отсчетам с использованием двух- и трехмерной математической модели

ЦЕЛЬ РАБОТЫ: Исследовать способы предсказания элементов в изображении, используя многомерные цепи Маркова, на качество восстановленного изображения.

Моделирование выполняется в среде Matlab.

1. Ознакомиться с математическими моделями на основе многомерных цепей Маркова (раздел 5.1).

2. Сформировать искусственное изображение, используя математическую модель на основе двух- или трехмерных цепей Маркова (раздел 6.5).

3. Полагая, что элементы, соответствующие закрашенным окружностям $x_{i,j,k}$ (рис. 6.1), в искусственном изображении неизвестны, восстановить их по ближайшим окрестным элементам с использованием многомерной математической модели. Возможные варианты предсказания элементов представлены в таблице 6.2.

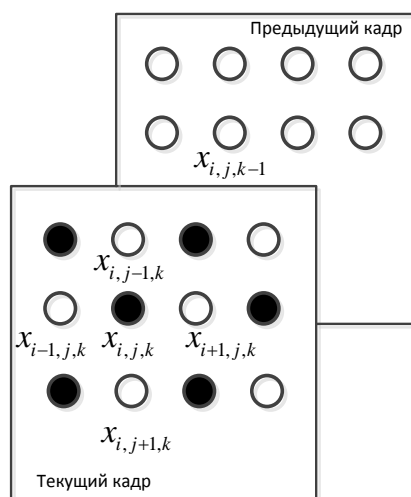


Рис. 6.1. Фрагмент видеопоследовательности

Таблица 6.2

	Элементы, по которым осуществляется предсказание
1	$x_{i,j,k-1}$
2	$x_{i-1,j,k}$
3	$x_{i,j-1,k}$
4	$x_{i-1,j,k}, x_{i,j-1,k}$
5	$x_{i-1,j,k}, x_{i+1,j,k}$
6	$x_{i,j-1,k}, x_{i,j+1,k}$
7	$x_{i-1,j,k}, x_{i,j,k-1}$
8	$x_{i,j,k-1}, x_{i,j-1,k}$
9	$x_{i-1,j,k}, x_{i,j-1,k}, x_{i,j,k-1}$
10	$x_{i-1,j,k}, x_{i,j-1,k}, x_{i+1,j,k}$

4. Для всех восстановленных изображений вычислить СКО.

5. В отчет включить:

- сформированное искусственное изображение;
- изображение с предсказанными элементами;
- СКО;
- ВЫВОДЫ.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какая модель изображения используется в приведенном алгоритме восстановления?
2. Как вычисляются оценки значений неизвестных элементов?
3. Как оценивалось качество восстановленного изображения?

6.7. Лабораторная работа №1. Исследование методов фильтрации изображений

ЦЕЛЬ РАБОТЫ: Исследовать методы пространственной фильтрации на качество восстановленных изображений, пораженных различными типами помех.

Исследование методов фильтрации изображений выполняется в визуальной инструментальной системе «Image Filtration», разработанной на кафедре радиоэлектронных средств.

Описание программы «Image Filtration»

Загрузка изображения

Загрузка изображений в программу для последующей обработки выполняется выбором пункта меню Файл -> Открыть. Обработка изображений осуществляется в формате BMP, GIF, JPEG, PNG, TIFF.

Добавление шума на изображении

Пункт меню – Добавить шум – позволяет вызвать окно для установки параметров шума (рис. 6.2).

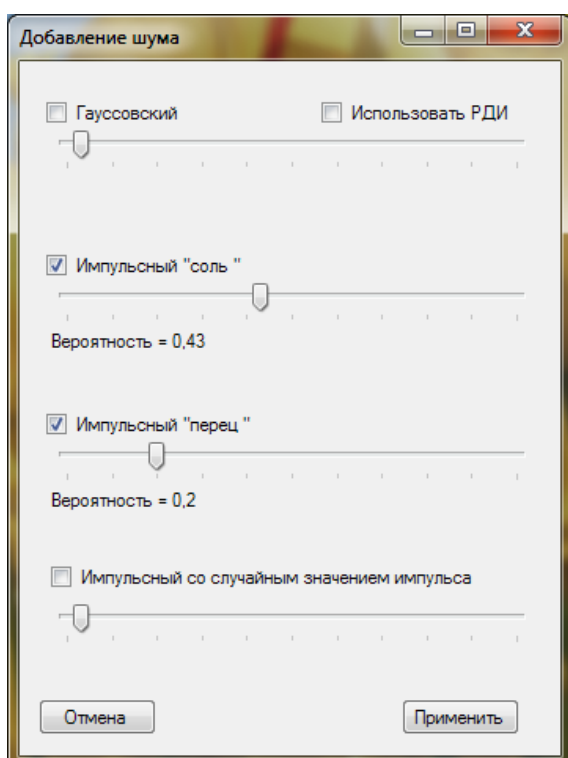


Рис. 6.2. Окно задания параметров накладываемого на изображение шума

Для применения одного из четырех видов шума нужно установить галочку у соответствующего пункта, после чего активизируется бегунок, перемещением которого с помощью мыши или стрелок клавиатуры можно задать нужное значение параметра, название и величина которого озвучиваются в подсказке под элементом управления. После нажатия кнопки «Применить» отмеченные виды шума будут последовательно применены к изображению.

Фильтрация изображений

Для фильтрации вызвать пункт меню «Применить фильтр», в котором содержится список классов фильтров (рис. 6.3а-б).

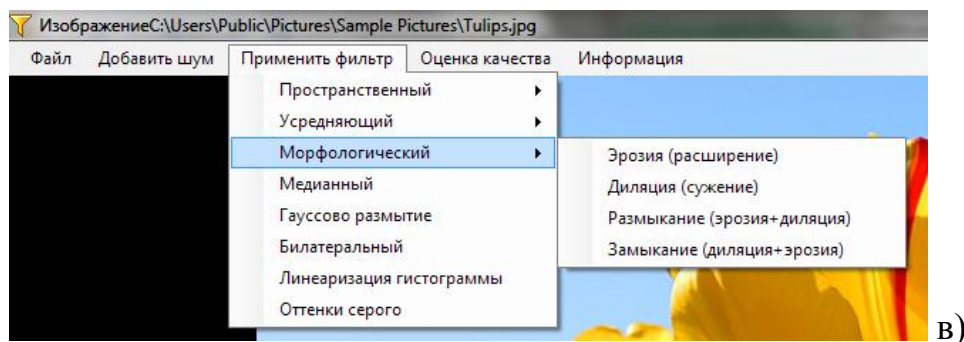
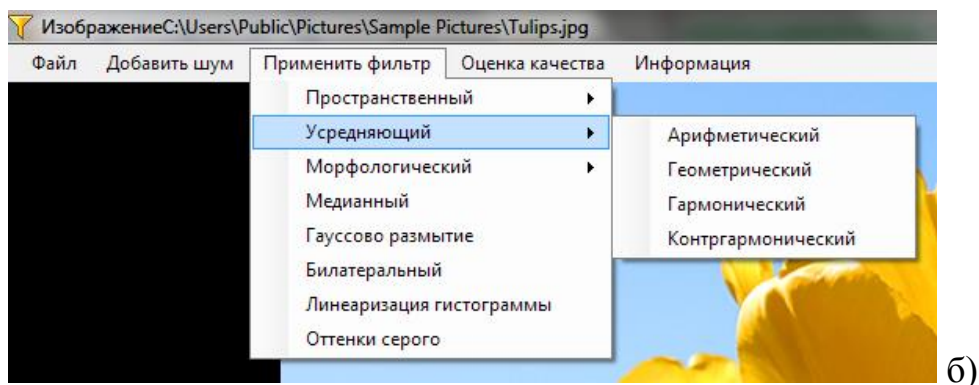
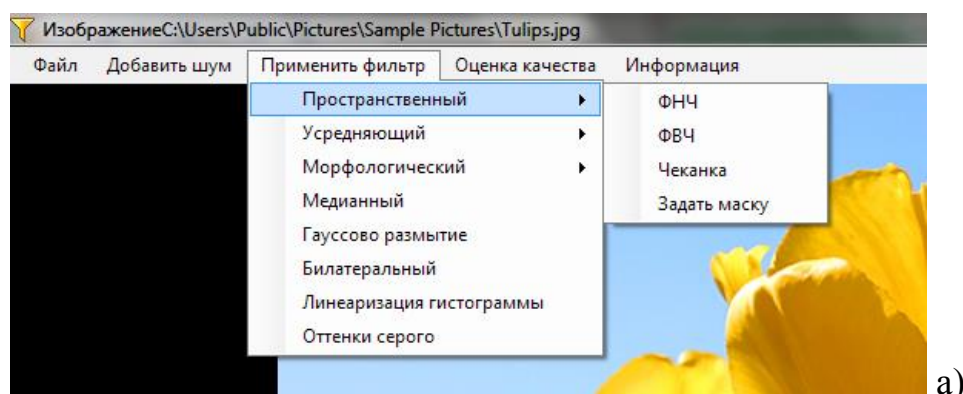


Рис. 6.3. Подпункты меню «Применить фильтр» с развернутыми списками фильтров

Для обработки изображений низкочастотным фильтром (ФНЧ) или высокочастотным фильтром (ФВЧ) задается размер маски и выбирается одна из стандартных масок. Вариант стандартной маски выбирается нажатием кнопки следующая (рис. 6.4 а,б).

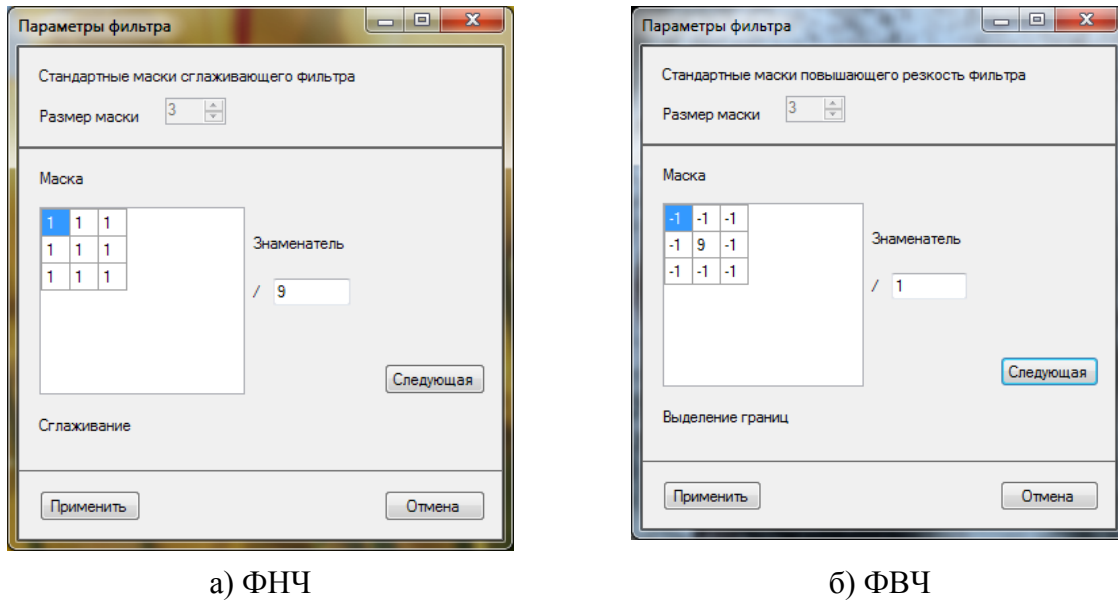


Рис. 6.4. Окна для выбора маски фильтра

При выборе усредняющего фильтра задаются размеры области усреднения по горизонтали и вертикали (рис. 6.5). Элемент, позволяющий указать порядок фильтра, активен только в случае выбора контргармонического фильтра.

Окно для ввода параметров медианного фильтра показано на рис. 6.6. Размер области усреднения по вертикали и по горизонтали может отличаться, доступные значения лежат в пределах от 3 до 9 пикселей.

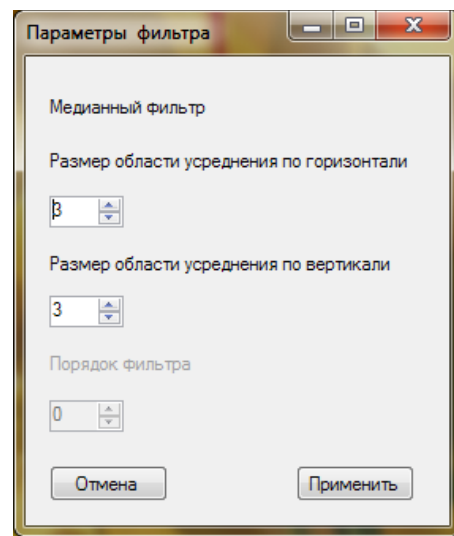
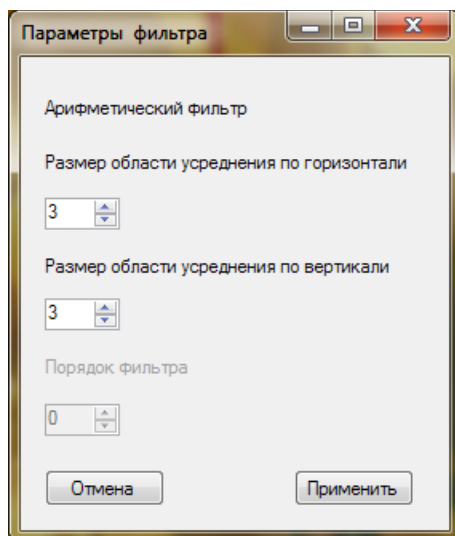


Рис. 6.5. Окно для ввода параметров усредняющего фильтра

Рис. 6.6. Окно для ввода параметров медианного фильтра

Оценка качества изображения может быть определена двумя критериями: пиковое отношение сигнал-шум (ПОСШ) и максимальное правдоподобие (КМП) (рис. 6.7).

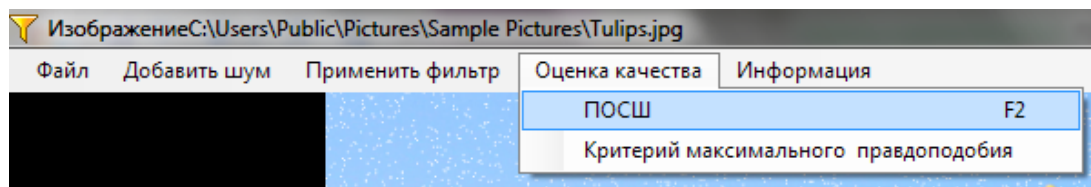


Рис. 6.7. Критерии оценки качества, используемые в программе Image Filtration

Задание к лабораторной работе

«Исследование методов фильтрации изображений»

1. Ознакомиться с методами пространственной фильтрации изображений (глава 3).
2. С помощью программы «Image Filtration» откройте тестовое изображение.
3. Обработайте это изображение фильтрами с разными размерами (3×3 , 5×5 , 7×7). Тип фильтра выберите согласно номеру варианта по таблице 6.3.
4. Исследуйте работу фильтров для разных видов шумов:
 - гауссовом ($\sigma = 0,1; 0,3; 0,5$);
 - импульсном шуме типа «соль» ($p = 0,2; 0,4; 0,6$);
 - импульсном шуме типа «перец» ($p = 0,2; 0,4; 0,6$);
 - импульсном шуме со случайным значением импульсов ($p = 0,2; 0,4; 0,6$).

Таблица 6.3

№ в-нт	Фильтры		
1.	Пространственный ФНЧ	Арифметический	Медианный
2.	Пространственный ФВЧ	Геометрический	Медианный
3.	Пространственный ФНЧ	Гармонический	Медианный
4.	Пространственный ФВЧ	Котргармонический	Медианный
5.	Пространственный ФНЧ	Геометрический	Медианный
6.	Пространственный ФВЧ	Гармонический	Морфологический
7.	Пространственный ФНЧ	Котргармонический	Морфологический
8.	Пространственный ФВЧ	Арифметический	Морфологический
9.	Пространственный ФНЧ	Геометрический	Морфологический
10.	Пространственный ФВЧ	Гармонический	Морфологический

5. Оцените качество обработанного изображения визуально, с помощью критериев ПОСШ и максимального правдоподобия.

6. Составьте последовательность действий, позволяющих восстановить зашумленное тестовое изображение.

7. В отчете приведите изображения до и после обработки и вычисленный критерий (ПОСШ) и максимальное правдоподобие (КМП). Сделайте выводы.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Поясните принцип работы масочных фильтров. Каковы их достоинства и недостатки?

2. Как качество фильтрации зависит от размера маски?

3. Почему при ограниченном размере окрестности, применяемой при масочной фильтрации, нельзя достичь предельного подавления шума?

4. Поясните принцип работы усредняющих фильтров. Каковы их достоинства и недостатки?

5. При каких значениях параметра Q контргармонический фильтр будет эффективно устранять импульсный шум типа «соль» и при каких значениях – шум типа «перец»? Объясните причину.

6. Поясните принцип работы медианного фильтра. Каковы его достоинства и недостатки?

7. Поясните принцип работы морфологического фильтра. Каковы его достоинства и недостатки?

8. Перечислите возможные области применения морфологического фильтра.

9. Какие методы фильтрации изображений эффективны при наличии импульсных помех, БГШ?

10. При решении каких задач выполняется предварительное сглаживание изображений?

11. Какие фильтры позволяют подчеркнуть границы на изображениях?

12. Перечислите последовательность действий, позволяющих уменьшить размытость изображений.

6.8. Лабораторная работа №2

«Исследование методов выделения контуров на изображениях»

ЦЕЛЬ РАБОТЫ: Исследовать методы выделения контуров на изображениях на точность выделения объекта интереса.

Исследование методов фильтрации изображений выполняется в визуальной инструментальной системе «EdgeDetector», разработанной на кафедре радиоэлектронных средств.

Описание программы «EdgeDetector»

Загрузка изображения

Загрузка полутонового или цветного изображения в формате bmp осуществляется через меню «Файл» программы EdgeDetector.

Преобразование цветного изображения в полутоновое

Преобразование загруженного изображения в 8-битное полутоновое осуществляется путём вызова процедуры «Преобразовать в Полутоновое» через главное меню «Операции» (рис. 6.8).

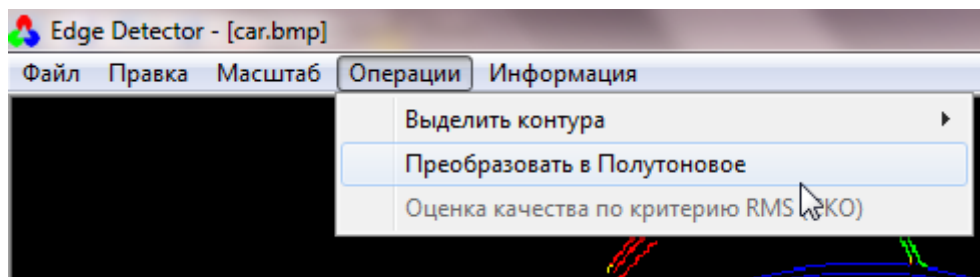


Рис. 6.8. Преобразование в полутоновое

Отмена преобразований

Для отмены всех преобразований, выполненных над исходным изображением, выполняется команда «Отмена» через главное меню «Правка» (рис. 6.9).

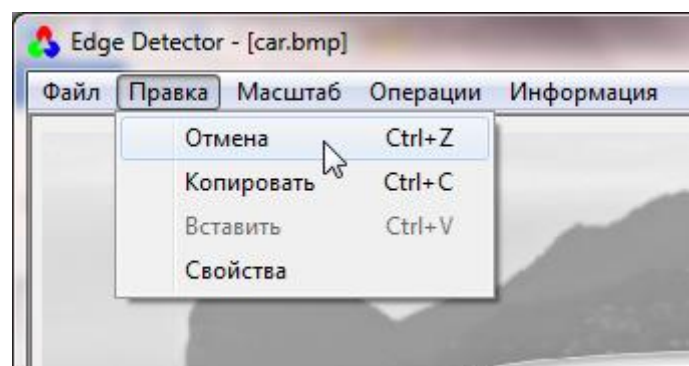


Рис. 6.9. Отмена операций

Операции выделения контуров на изображении

Выделение контуров на изображении осуществляется через главное меню «Операции».

Окно методов выделения контуров Собела, Превитта, Робертса показано на рис. 6.10. При выборе метода из выпадающего списка маски отображаются ниже в таблицах.

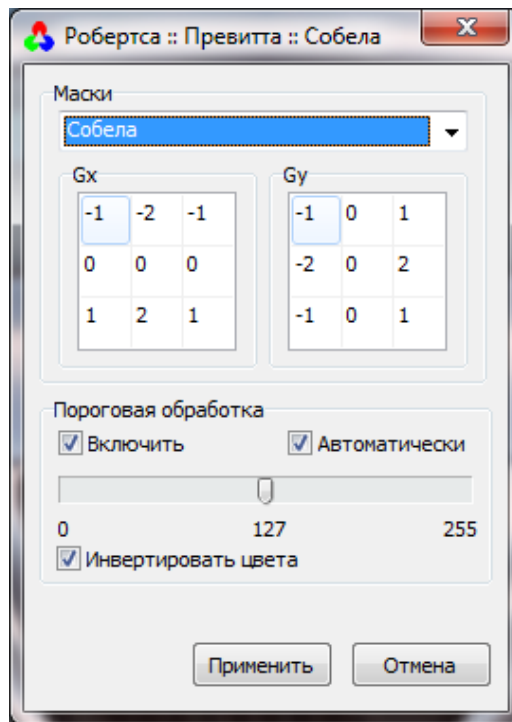


Рис. 6.10. Окно методов Собела, Превитта, Робертса

В окне доступно включение пороговой обработки, автоматического подбора порога, цветовой инверсии. Так же пользователь может с помощью движка выбрать порог, предварительно отключив автоматический режим.

По нажатию «Применить» будет сформировано контурное изображение (рис. 6.11).

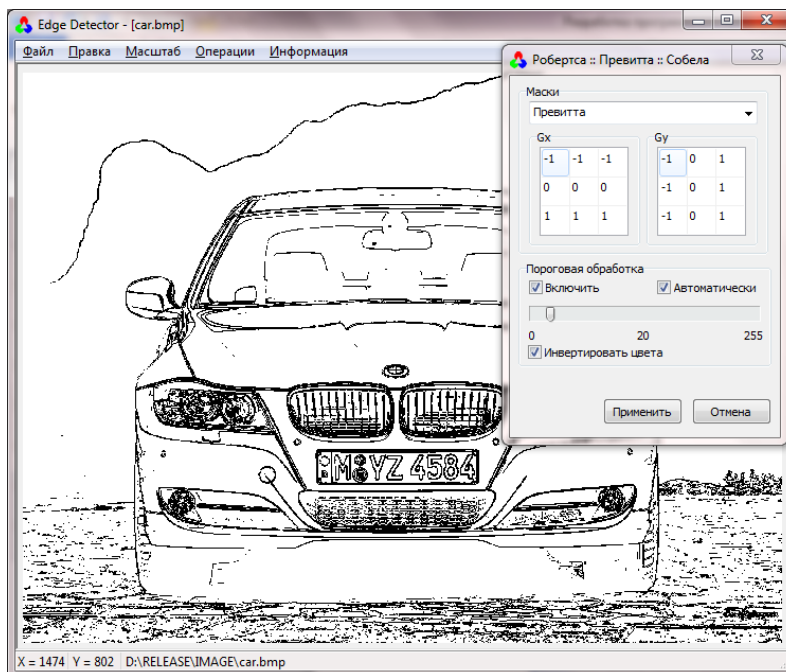


Рис. 6.11. Контурное инверсное изображение

Окно метода Превитта-Кирша (рис. 6.12). Из выпадающего списка выбираются маски Превитта или Кирша. Первая маска из серии отображается в таблице, в остальном интерфейс окна аналогичен интерфейсу остальных методов выделения контуров.

Окно для задания масок градиентных методов пользователем (рис. 6.13) предназначено для задания размеров и коэффициентов маски пользователем.

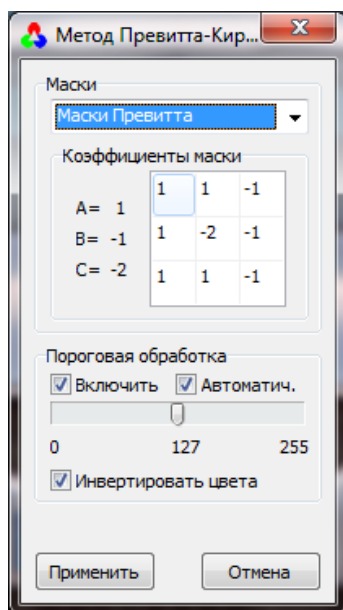


Рис.6.12. Окно метода Превитта-Кирша

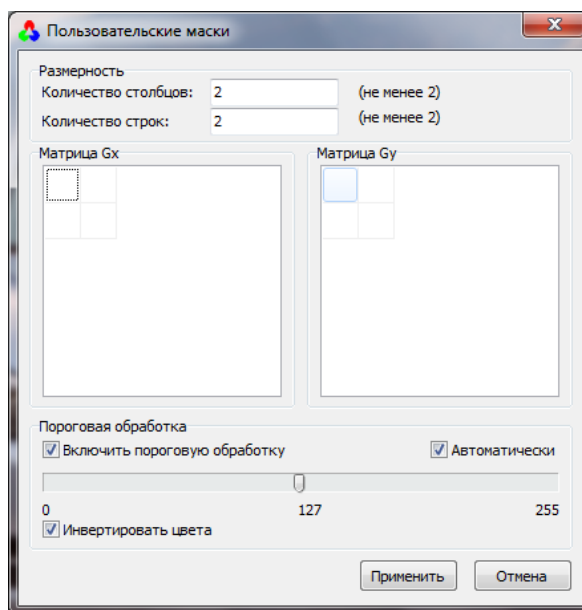


Рис.6.13. Окно для задания масок градиентных методов пользователем

Окно Лапласиана (рис. 6.14). Из выпадающего списка пользователю предлагается выбрать одну из предустановленных масок или задать маску 3x3 вручную.

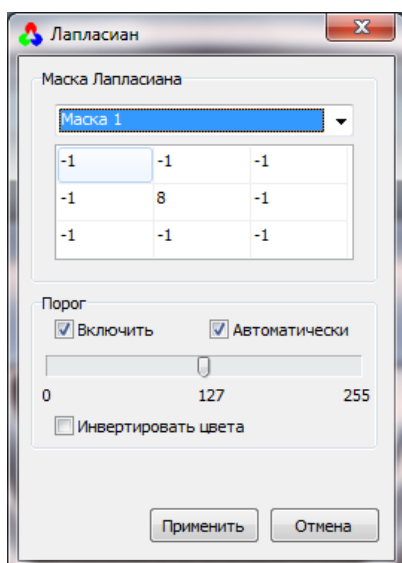


Рис.6.14. Окно Лапласиана

Окно Лапласиана Гауссиана. Из выпадающего списка пользователю предлагается выбрать способ задания маски: ручную (рис. 6.15) или автоматически (рис. 6.16).

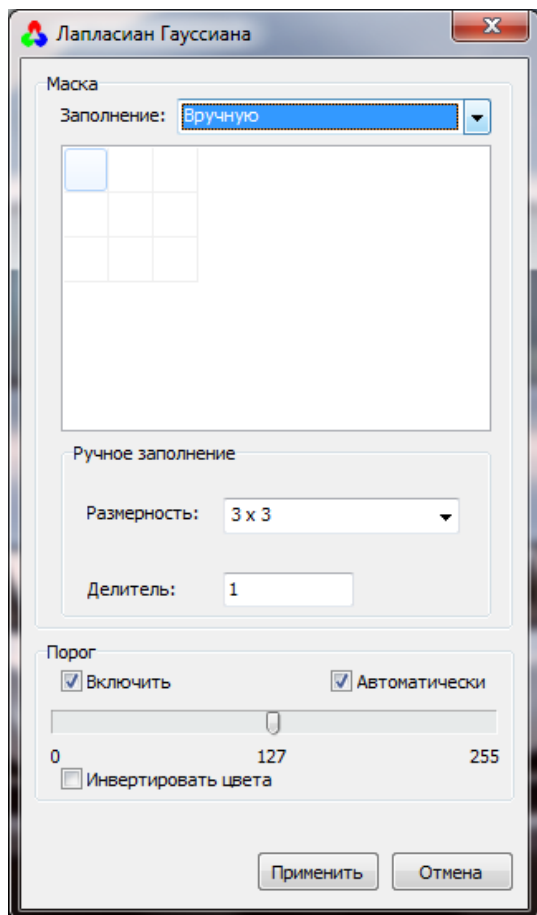


Рис.6.15. Пользовательский способ задания маски

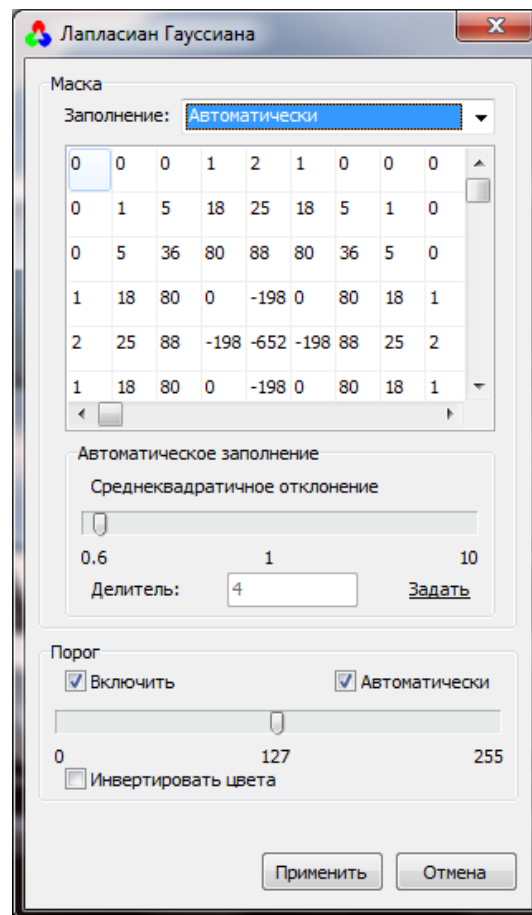


Рис.6.16. Автоматический способ задания маски

В ручном режиме все параметры вводятся пользователем, размерность маски выбирается из предустановленных вариантов, делитель обычно равен сумме коэффициентов маски. В автоматическом режиме пользователь задаёт только среднеквадратичное отклонение. По одиночному клику мышкой по надписи «Задать» производится генерация маски с заданным параметром.

Окно ручного ввода маски метода второй производной (рис. 6.17) имеет стандартный интерфейс задания порога, а так же полностью ручной ввод маски: задание её размеров и коэффициентов.

Окно метода Канни (рис. 6.18) отличается интерфейсом от всех остальных окон методов выделения контуров. Главное отличие – применение двух порогов в пороговой обработке и использование цвета при отображении преобразованного изображения в главном окне программы. Здесь цветом обозначены направления градиента (рис. 6.18).

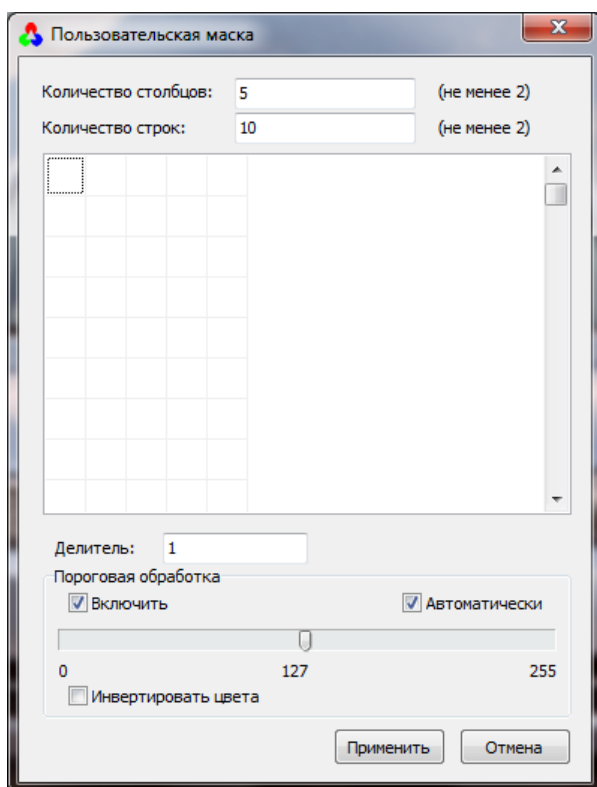


Рис. 6.17. Окно ручного ввода маски метода второй производной

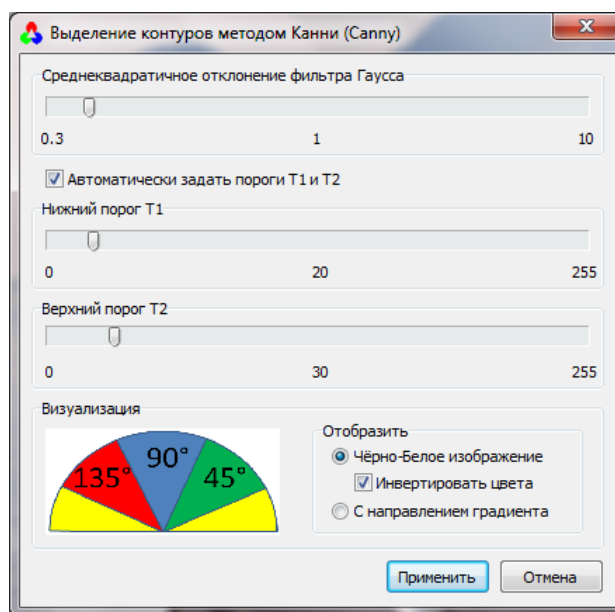


Рис. 6.18. Окно метода Канни

Расчет среднеквадратичной ошибки

Для расчёта критерия СКО выполняется сравнение полученного контурного изображения с идеальным. Для этого в главном окне программы загружается контурное изображение, полученное программой EdgeDetector. А через главное меню «Операции» | «Оценка качества по критерию СКО» загружается идеальное контурное изображение. При загрузке файла нужного формата, будет доступна клавиша «Расчёт», пример расчёта СКО приведён на рис. 6.19.

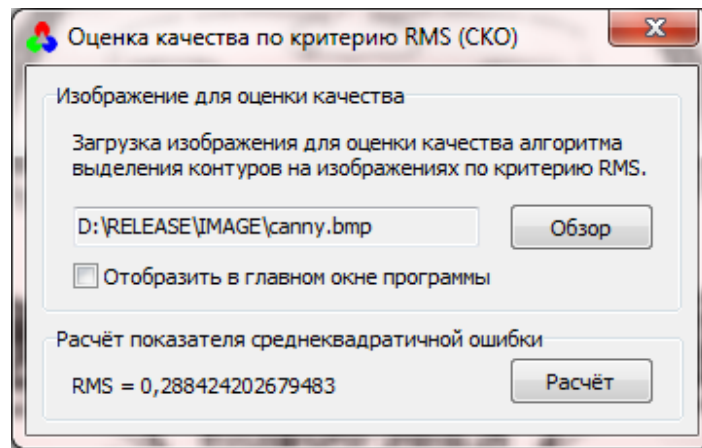


Рис.6.19. Пример расчёта СКО

Наложение шума на изображение

Наложение шума на изображение выполняется через главное меню «Операции» | «Добавить шум». В окне (рис. 6.20) с помощью движка задается дисперсия шума и добавляется шум.

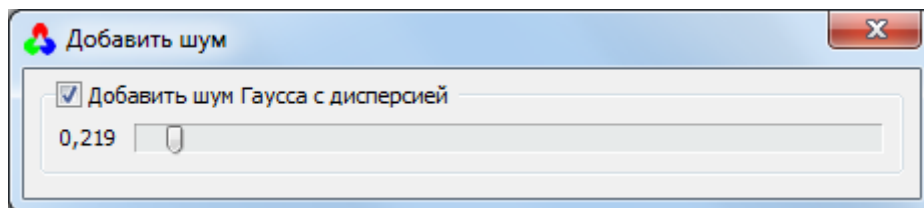


Рис. 6.20. Добавление шума

Задание к лабораторной работе

«Исследование методов выделения контуров на изображениях»

1. Ознакомиться с методами выделения контуров на изображениях (глава 4).
2. С помощью программы «EdgeDetector» откройте тестовое изображение (таблица 6.4).

Таблица 6.4

№ в-нт	Методы			Тестовые изображения к п.2-6	Тестовые изображения к п.7
1.	Робертса	Канни	Лапласиан	3096	а)
2.	Собела	Канни	Лапласиана-Гауссиана	37073	б)
3.	Превитта	Канни	Лапласиан	42049	в)
4.	Превитта-Кирша	Канни	Лапласиана-Гауссиана	69020	г)
5.	Собела	Канни	Лапласиан	119082	д)
6.	Робертса	Канни	Лапласиана-Гауссиана	14037	а)
7.	Собела	Канни	Лапласиан	41033	б)
8.	Превитта	Канни	Лапласиана-Гауссиана	66053	в)
9.	Превитта-Кирша	Канни	Лапласиан	101087	г)
10.	Собела	Канни	Лапласиана-Гауссиана	167062	д)

3. Получить контурные изображения методами, перечисленными в таблице 6.4.

4. Исследовать качество выделения контуров в зависимости от выбора порога. Определить, при каком значении порога качество контурного изображения будет наилучшее. Качество контурного изображения оценить визуально и с помощью критерия *СКО*. Для вычисления *СКО* необходимо сравнить сформированное контурное изображение с идеальным. В отчете привести по три примера изображений для каждого метода контурной сегментации: с наилучшим качеством, недосегментированное и пересегментированное изображения и их вычисленные *СКО*.

5. Исследовать качество выделения контуров от уровня шума. Для наложения шума выберите пункт ОПЕРАЦИИ /ДОБАВИТЬ ШУМ. Определить порог, при котором качество контурного изображения будет наилучшее. Получите контурное изображение всеми методами. Вычислите *СКО*. В отчете приведите по три примера выделения контуров разными методами при разных уровнях шума (например, $\sigma = 0, 2; 0, 4; 0, 6$).

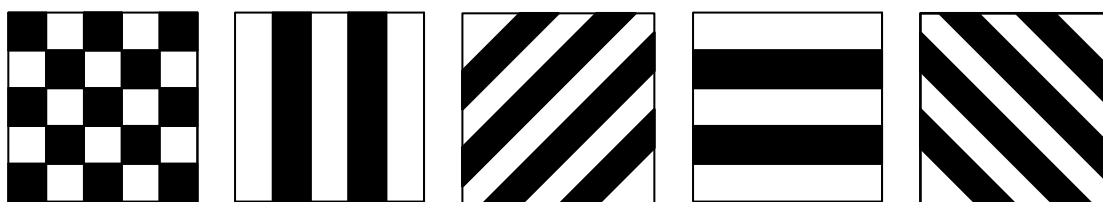
6. Создать новую маску из пункта ОПЕРАЦИИ / ВЫДЕЛИТЬ КОНТУРЫ/.../ЗАДАТЬ МАСКИ ВРУЧНУЮ:

$$\text{а) } \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad \text{б) } \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}, \quad \text{в) } \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}, \quad \text{г) } \begin{bmatrix} 0 & -1 & 0 \\ -1 & n & -1 \\ 0 & -1 & 0 \end{bmatrix},$$

$$5 \leq n \leq 10.$$

Примените созданные маски к изображению. Посмотрите, что получится.

7. Создайте маски, позволяющие выделить контуры на следующих изображениях (рис. 6.21):



а) б) в) г) д)

Рис. 6.21

8. В отчете приведите изображения до и после обработки всеми методами и вычисленный критерий *СКО*. Сделайте выводы.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Поясните принцип работы градиентных методов выделения контуров. Каковы их достоинства и недостатки?

2. Поясните принцип методов второй производной выделения контуров. Каковы их достоинства и недостатки?

3. Поясните принцип метода Канни выделения контуров. Каковы достоинства и недостатки метода Канни?

4. Нарисуйте вид градиента и лапласиана для изображения на рис. 6.22.



рис. 6.22

5. Какие из методов выделения контуров, рассмотренные в главе 4, наиболее устойчивы к шуму?

6. Какие из методов выделения контуров, рассмотренные в главе 4, требуют меньших вычислений?

7. Двоичное изображение содержит прямые линии, ориентированные горизонтально. Приведите пример маски 3×3 для обнаружения в этих линиях разрывов шириной в один пиксел.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Дворкович, В.П. Цифровые видеоинформационные системы (теория и практика) [Текст] / В.П. Дворкович, А.В. Дворкович. – М.: Техносфера, 2012 – 1008 с.
2. Гонсалес, Р. Цифровая обработка изображений [Текст] / Р. Гонсалес, Р. Вудс. – М.: Техносфера, 2012 – 1104 с.
3. Ричардсон, Я. Видеокодирование. H.264 и MPEG-4 – стандарты нового поколения. [Текст]. – М.: Техносфера, 2005. – 368 с.
4. Приоров, А.Л. Цифровая обработка изображений [Текст]: уч. пособие / А.Л. Приоров, И.В. Апальков, В.В. Хрящев; Яросл. гос. ун-т. – Ярославль: ЯрГУ, 2007.
5. Хафизов, Р.Г. Обработка цветных медицинских изображений [Текст]: уч. пособие / Р.Г. Хафизов. – Йошкар-Ола: Поволжский государственный технологический университет, 2012 – 100 с.
6. Ватолин, Д. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео [Текст] / Ватолин Д., Ратушняк А., Смирнов М., Юкин В. – М.: ДИАЛОГ-МИФИ, 2002. – 384 с.
7. Методы компьютерной обработки изображений [Текст] / под ред. В. А. Сойфера. – М.: ФИЗМАТЛИТ, 2006. – 784 с.
8. Зубарев, Ю.Б. Видеоинформационные технологии систем связи: Монография [Текст] / Ю.Б. Зубарев, Ю.С. Сагдуллаев, Т.Ю. Сагдуллаев. – М.: Изд-во «Спутник+», 2011. – 296 с.
9. Чобану, М. Многомерные многоскоростные системы обработки сигналов [Текст]. – М.: Техносфера, 2009. – 480 с.
10. Штарк, Г.-Г. Применение вейвлетов для ЦОС [Текст]. – М.: Техносфера, 2007. – 192 с.
11. Пескин, А.Е. Мировое вещательное телевидение. Стандарты и системы: Справочник [Текст] / А.Е. Пескин, В.Ф. Труфанов. – М.: Горячая линия – Телеком, 2008. – 308 с.

12. Дворкович, В. П. Проблемы и перспективы развития систем кодирования динамических изображений [Текст] / В. П. Дворкович, М. К. Чобану // MediaVision. – 2011, №№2–5.

13. ISO/IEC 14496-2:2004 - Information technology - Coding of audio-visual objects (MPEG-4 video). - ITU-T. 2005.

14. ITU-T Recommendation H.264. Advanced video coding for generic audiovisual services. - ITU-T. 2009.

15. ITU-T Recommendation H.265. High efficiency video coding, 04/2013.

16. Грузман, И. С. Цифровая обработка изображений в информационных системах [Текст]: учебник НГТУ: учебное пособие / И. С. Грузман, В. С. Киричук, В. П. Косых, Г. И. Перетягин, А. А. Спектор. – Новосибирск: Изд-во НГТУ, 2002. – 352 с.

17. Тихонов, В. И. Статистическая радиотехника [Текст]. – М.: Советское радио, 1966. – 678 с.

18. Богословский, А. В. Обработка многомерных сигналов. Линейная многомерная дискретная обработка сигналов. Методы анализа и синтеза [Текст]. / А. В. Богословский, Е. А. Богословский, И. В. Жигулина, В. А. Яковлев. – М.: Радиотехника, 2013. – 168 с.

19. Потапов, А. А. Новейшие методы обработки изображений [Текст] / А. А. Потапов, Ю. В. Гуляев, С. А. Никитов. М.: ФИЗМАТЛИТ, 2008. – 496 с.

20. Бакут, П. А. Сегментация изображений: Методы пороговой обработки [Текст] / П. А. Бакут, Г. С. Колмогоров, И. Е. Ворновицкий // Зарубежная радиоэлектроника, 1987 – №10 – С. 6–24.

21. Бакут, П. А. Сегментация изображений: Методы выделения границ областей [Текст] / П. А. Бакут, Г. С. Колмогоров // Зарубежная радиоэлектроника 1987 – №10. – С. 25–47.

22. Zhang, Y. Advances in Image And Video Segmentation / Y. Zhang. – USA: IRM Press, 2006. – 473 p.

23. Яне, Б. Цифровая обработка изображений [Текст]. – М.: Техносфера, 2007. – 584 с.
24. Введение в контурный анализ и его приложения к обработке изображений и сигналов [Текст] / Я.А. Фурман, А.В. Кревицкий, А.К. Передреев и др.; под.ред. Я.А. Фурмана. – М.: ФИЗМАТЛИТ, 2002. – 592 с.
25. Лукьяница, А.А. Цифровая обработка видеоизображений [Текст] / А.А. Лукьяница, А.Г. Шишкин. – М.: Техносфера, 2009. – 518 с.
26. Методы автоматического обнаружения и сопровождения объектов. Обработка изображений и управление [Текст]/ Б.А. Алпатов, П.В. Бабаян, О.Е. Балашов, А.И. Степашкин. – М.: Радиотехника, 2008. – 176 с.
27. Медицинские информационные технологии и системы [Текст] / С.В. Абламейко и др. – Минск: ОИПИ НАН Беларуси, 2007. – 176 с.
28. Роджерс, Д. Алгоритмические основы машинной графики [Текст]. – М.: Мир, 1989. – 512с.
29. Шапиро, Л. Компьютерное зрение [Текст] / Л. Шапиро, Дж. Стокман. Изд-во: Бином. Лаборатория знаний, 2006 г. – 752 с.
30. Васильев, К.К. Статистический анализ многомерных изображений [Текст]/ К.К. Васильев, В.Р. Крашенинников. – Ульяновск: УлГТУ, 2007. – 170 с.
31. Петров, Е.П. Метод синтеза математических моделей видеоизображений на основе многомерных цепей Маркова [Текст]/ Е.П. Петров, Е. В. Медведева, А.П. Метелев // Нелинейный мир, № 4, 2011. – С.213–231.
32. Петров, Е.П. Математические модели видеопоследовательностей цифровых полутоновых изображений [Текст] / Е.П. Петров, И.С. Трубин // Успехи современной радиоэлектроники. – 2007, №6. – С. 3–31.
33. Petrov, E. P., Trubin, I. S., Medvedeva, E. V. , Smolskiy, S. M. Mathematical Models of Video-Sequences of Digital Half-Tone Images // Integrated models for information communication systems and net-works : design and development / Aderemi A. Atayero and Oleg I. Sheluhin, editors, 2013 by IGI Global. – PP.207–241.

34. Petrov, E. P, Trubin, I. S., Medvedeva, E. V., Smolskiy, S. M. Development of Nonlinear Filtering Algorithms of Digital Half-Tone Images // Integrated models for information communication systems and networks : design and development / Aderemi A. Atayero and Oleg I. Sheluhin, editors, 2013 by IGI Global. – P.278–304.

35. Петров, Е.П. Нелинейная фильтрация статистически связанных видеопоследовательностей цифровых полутоновых изображений [Текст] / Е.П. Петров, Е. В. Медведева // Радиотехника и электроника, том 55, № 3, 2010. – С. 330–339.

36. Medvedeva, E.V. A Two-stage image preprocessing algorithm / E. V. Medvedeva, E. E. Kurbatova // Pattern Recognition and Image Analysis, Vol. 21, No. 2, 2011. – P.297–301.

37. Медведева, Е.В. Метод текстурной сегментации изображений на основе марковских случайных полей [Текст] / Е.В. Медведева, Е.Е. Курбатова // Цифровая обработка сигналов. – 2012. – №3. – с. 76–80.

38. Медведева, Е. В. Метод компрессии видеоизображений на основе двумерных цепей Маркова [Текст] / Е. В. Медведева, Б.О.Тимофеев // Т-Comm. Телекоммуникации и транспорт. № 1, 2011. – С.24–28.