

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение высшего
профессионального образования
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ
И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)

**Институт инноватики
Факультет инновационных технологий**

Отделение кафедры ЮНЕСКО «Новые материалы и технологии»

Методические указания

для проведения практических занятий
по дисциплине **«Методы искусственного интеллекта в мехатронике и робототехнике»**
по направлению 221000.68 «Мехатроника и робототехника»
магистерская программа «Проектирование и исследование мультикоординатных
электромехатронных систем движения»

Методические указания по практическим занятиям по направлению 221000.68 «Мехатроника и робототехника», магистерская программа «Методы искусственного интеллекта в мехатронике и робототехнике» рассмотрены и утверждены на заседании Отделения кафедры ЮНЕСКО «Новые материалы и технологии» 27.03. 2012 г., протокол № 8.

Разработчики:

Преподаватель ОКЮ

С.В. Комзолов

СОГЛАСОВАНО:

Зав. профилирующей каф. УИ

_____ А.Ф. Уваров

Зав. выпускающим

Отделением кафедры ЮНЕСКО

_____ Ю.М. Осипов

Цель проведения занятий

Практические занятия направлены на закрепление и расширение знаний, полученных на лекциях; объем занятий - 18 часов.

Практические занятия по курсу, направлены на укрепление знаний в области мехатроники и робототехники, а также в области управления и автоматизации с использованием искусственного интеллекта.

Предусмотрен тестовый контроль полученных знаний в объеме, предусмотренном рейтинговой раскладкой для данной дисциплины (см. приложение А). Тестовый контроль проводится в виде контрольных работ по изучаемым темам.

Практическая работа № 1

СЛУЧАЙНЫЕ ХАРАКТЕРИСТИКИ ОБЪЕКТОВ В ПОТОКАХ ДАННЫХ

Цель работы: освоение методов работы с файлами; получение навыков проведения анализа информационных массивов с использованием методов математической статистики.

Общие сведения

Математическая статистика в курсе «Теория принятия решений и распознавание образов» рассматривается как наука о статистических решениях, принимаемых при оценке характеристик исследуемых объектов и помех: случайных величин, событий, процессов, сигналов, потоков и т. п.

Достоверность оценок, доверительные интервалы их существования, виды законов распределения последних – важнейшие предпосылки рационального принятия решений, в том числе и при распознавании образов.

В общей схеме распознавания образов и принятия решений преобладающей по трудовым затратам является операция преобразования пространства наблюдений с целью получения компактного описания объектов в пространстве признаков (рис. 1.1). Для формирования зон существования признаков, характеристики и параметры которых носят случайный характер, используют методы математической статистики. Наиболее часто – это определение законов распределения и их параметров по имеющимся выборкам.

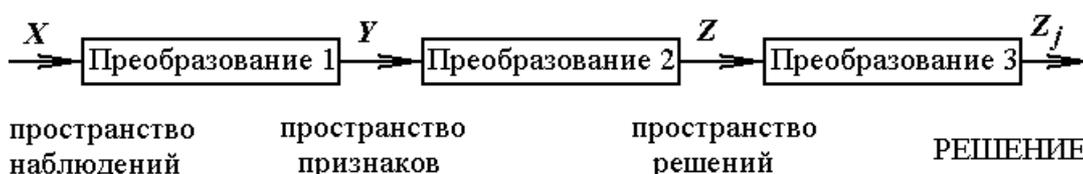


Рис. 1.1. Обобщенная схема принятия решений при распознавании образов

По результатам наблюдений проявлений признаков в различных реализациях распознаваемых объектов, ситуаций формируется статистическая гипотеза о виде и параметрах закона распределения. Непрерывные распределения удобны для аналитической оценки параметров исследуемых объектов. В практике же эксперимента работают с дискретными пространствами, и наблюдаемые случайные величины характеризуются дискретными распределениями. Для анализа структуры и наполнения файлов необходимы программные средства доступа файлам. Их разработка мало целесообразна в связи с наличием достаточно большого задела в данной области.

Доступ к файлам

Различные среды программирования обработки, пакеты анализа данных предоставляют для ввода данных, различных форматов, отработанные процедуры, классы, СОМ – объекты. Среди них выделяются универсальные, не привязанные к внутренней структуре файла функции. Примером могут служить функции среды Mathcad READBIN() и обратная функция WRITEBIN(). Их параметры описаны ниже:

READBIN("file", "type", [[endian], [cols], [skip], [maxrows]]) Returns an containing the binary data in file.

WRITEBIN("file", "type", endian, [M]) writes an of scalars to the binary data file named file. This function can be used either on the right or the left side of the . When used on the right, you must supply the argument M, the name of the matrix to write to the file. In this case, the function returns

the contents of the matrix. When used on the left, do not supply the argument M, but instead place it on the right-hand side of the definition. See the QuickSheet Example.

Arguments:

- file is a containing the filename or the full and filename. Pathnames are relative to the .
- type is a that specifies the data format used in the file. READBIN supports the following data types:
 - byte - 8-bit unsigned integer.
 - double - 64-bit floating point number.
 - float - 32-bit floating point number.
 - int16 - signed 16-bit integer.
 - int32 - signed 32-bit integer.
 - uint16 - unsigned 16-bit integer.
 - uint32 - unsigned 32-bit integer.
- endian (optional for READBIN) indicates whether the data in the file is big-endian (high byte first) or little-endian (low byte first). Big-endian is represented by a 1, while little-endian is represented by a 0. If no value is specified, endian is assumed to be 0.
- cols (optional) is the positive integer number of columns per row in the input file. Default is 1.
- skip (optional) is the non-negative integer number of bytes at the beginning of the file to ignore before reading in data. Default is 0.
- maxrows (optional) is the non-negative integer maximum number of rows of data to be read from the data file. Default is 0 (no limit).

Note:

· If you are not sure whether your data is written in big-endian (the Macintosh standard) or little-endian (the IBM-PC standard) format, please consult the system documentation for your data source before relying upon the accuracy of any result from READBIN or WRITEBIN.

Функции, ориентированные на конкретную предметную область разнообразны и сами производят дешифрацию данных, предоставляя их в наиболее удобном для анализа виде. Следующие функции преобразуют прочитанные файлы изображений представленных в различных форматах в матрицу.

READ_IMAGE("file") Creates an array containing a grayscale representation of the image file.

READBMP("file") Creates an array containing a grayscale representation of a bitmap format image file. (deprecated)

READRGB("file") Creates an array containing a red-green-blue packed matrix representation of the image file.

READ_RED("file"), READ_GREEN("file"), READ_BLUE("file") Create an array containing only the red, green, or blue component from a color image file.

WRITEBMP("file", [M]) Creates an grayscale bitmap image file. This function can be used either on the right or the left side of the [definition operator](#). When used on the right, you must supply the argument M, the name of the matrix to write to the file. In this case, the function returns 0. When used on the left, do not supply the argument M, but instead place it on the right hand side of the definition.

WRITERGB("file", [M]) Creates an 16-million-color bitmap image file. This function can be used either on the right or the left side of the [definition operator](#). Notes for WRITEBMP apply.

READ_HLS("file") Creates an array containing a hue, lightness, and saturation representation of the image file.

READ_HLS_HUE("file"), READ_HLS_LIGHT("file"), READ_HLS_SAT("file") Extract only the hue, light, or saturation component from a color image.

READ_HSV("file") Creates an array containing a hue, saturation, and value representation of the image file.

READ_HSV_HUE("file"), READ_HSV_SAT("file"), READ_HSV_VALUE("file") Extract only the hue, saturation, or value component from a color image.

WRITE_HSV("file", [M]) Creates an 16-million-color bitmap image file. This function can be used either on the right or the left side of the [definition operator](#). Notes for WRITEBMP apply.

WRITE_HLS("file", [M]) Creates an 16-million-color bitmap image file. This function can be used either on the right or the left side of the [definition operator](#). Notes for WRITEBMP apply.

Arguments:

- "file" is a string containing the file name or the full pathname and the file name. Recognized image formats are BMP, GIF, JPG, PCX, or TGA.
- M is a matrix.

Notes:

- Function names are all UPPERCASE.
- Regardless of original image file format, images in Mathcad are matrices with values between 0 (black) and 255 (white), which can be displayed with the Picture operator. Each matrix element represents the intensity of one pixel. RGB Color images are stored as three packed m x n matrices, augmented as follows:

Аналогично и в других средах, например, в C++ совокупности с SDK DirectX чтение различных форматов не вызывает затруднений. Это - данные, изображения и т. п.:

IDirectXFileBinary::Read

Reads the binary data. Deprecated.

```
HRESULT Read(  
    LPVOID pvData,  
    DWORD cbSize,  
    LPDWORD pcbRead  
);
```

Parameters

pvData

[out] Pointer to the buffer that receives the data that has been read.

cbSize

[in] Size of the buffer pointed to by pvData, in bytes.

pcbRead

[out] Pointer to the number of bytes actually read.

Return Values

If the method succeeds, the return value is DXFILE_OK. If the method fails, the return value can be one of the following values: DXFILEERR_BADVALUE, DXFILEERR_NOMOREDATA.

Requirements

Header: Declared in Dxfile.h.

Файлы изображений

D3DXIFF_BMP

Windows bitmap (BMP) file format.

D3DXIFF_JPG

Joint Photographics Experts Group (JPEG) compressed file format.

D3DXIFF_TGA

Truevision (Targa, or TGA) image file format.

D3DXIFF_PNG

Portable Network Graphics (PNG) file format.

D3DXIFF_DDS

DirectDraw surface (DDS) file format.

D3DXIFF_PPM

Portable pixmap (PPM) file format.

D3DXIFF_DIB

Windows device-independent bitmap (DIB) file format.

D3DXIFF_HDR

High dynamic range (HDR) file format.

D3DXIFF_PFM

Portable float map file format.

Статистическая обработка полученных изображений

Определим центральные моменты, назначим число каналов анализа (например 100), положение канала на оси th и найдем функцию гистограммы rh (например, для составляющей R выбранного сегмента):

$$th_i = Rmean + 0.1 \cdot Rstdev \cdot (i - 50),$$

$$rh = \text{hist}(th, R),$$

где $Rmean$, $Rstdev$ – оценки математического ожидания и среднеквадратичного отклонения составляющей R изображения в сегменте.

Ниже на рис. 1.2 приведены параметры гистограмм для сегментов и на рис. 1.3 – их вид.

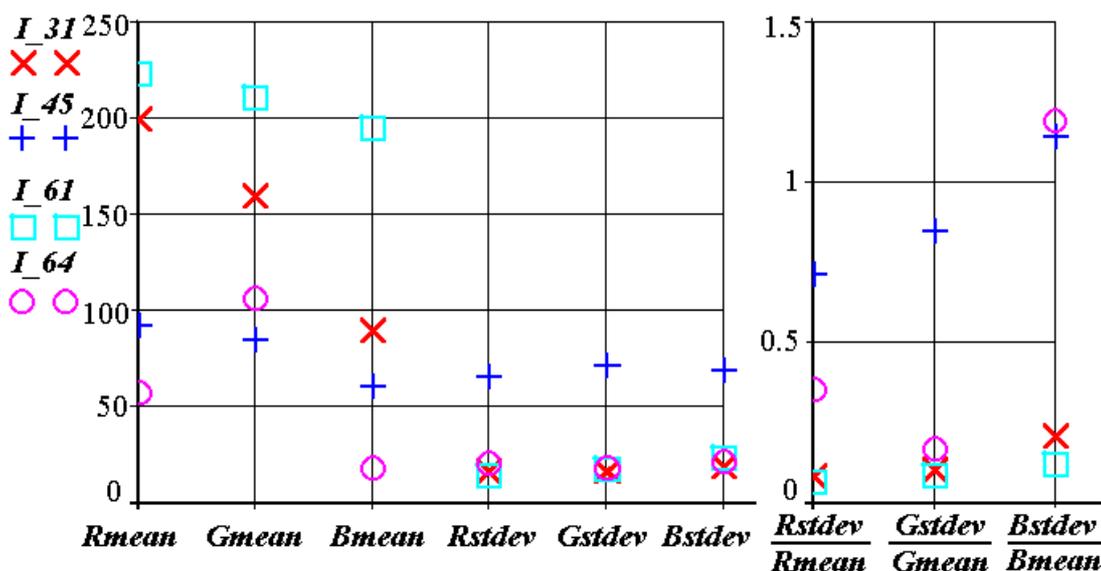


Рис. 1.2. Параметры гистограмм r , g , b :

$Rmean$, $Gmean$, $Bmean$ оценки величин математических ожиданий составляющих R , G , B ;

Rstdev, Gstdev, Bstdev оценки величин среднеквадратичных отклонений составляющих; R, G, B; I_31, I_61, I_45, I_64 значения оценок указанных центральных моментов для шляпки, ножки гриба, для неоднородного и квазиоднородного фона

В приведенном примере не обосновывается выбор количества каналов, метрика оси параметра th для простоты принята линейной. В ряде практических случаев такие упрощения не допустимы. Доверительный интервал оценки показаний канала гистограммы зависит от его показаний, общего числа опытов.

Визуально ошибки в выборе положения и размера канала видны по изрезанности полученных кривых, в явном превышении показаний отдельных каналов над показаниями остальных и т. п.

По полученным гистограммам принимается решение о виде закона распределения и о его параметрах. Наиболее часто гистограммы анализируются по их виду на близость к известным функциям плотности вероятности.

Рассмотрим эти процедуры, ориентируясь на терминологию теории распознавания образов. Описание классов, к одному из которых следует отнести исследуемую гистограмму, задано аналитическими выражениями. На рис. 1.5 приведены примеры функций плотностей вероятности, имеющих в перечне стандартных функций. Необходимо определить потери (или выигрыш) от отнесения полученных гистограмм к заданным классам и принять решение о принадлежности конкретной гистограммы к одному из них.

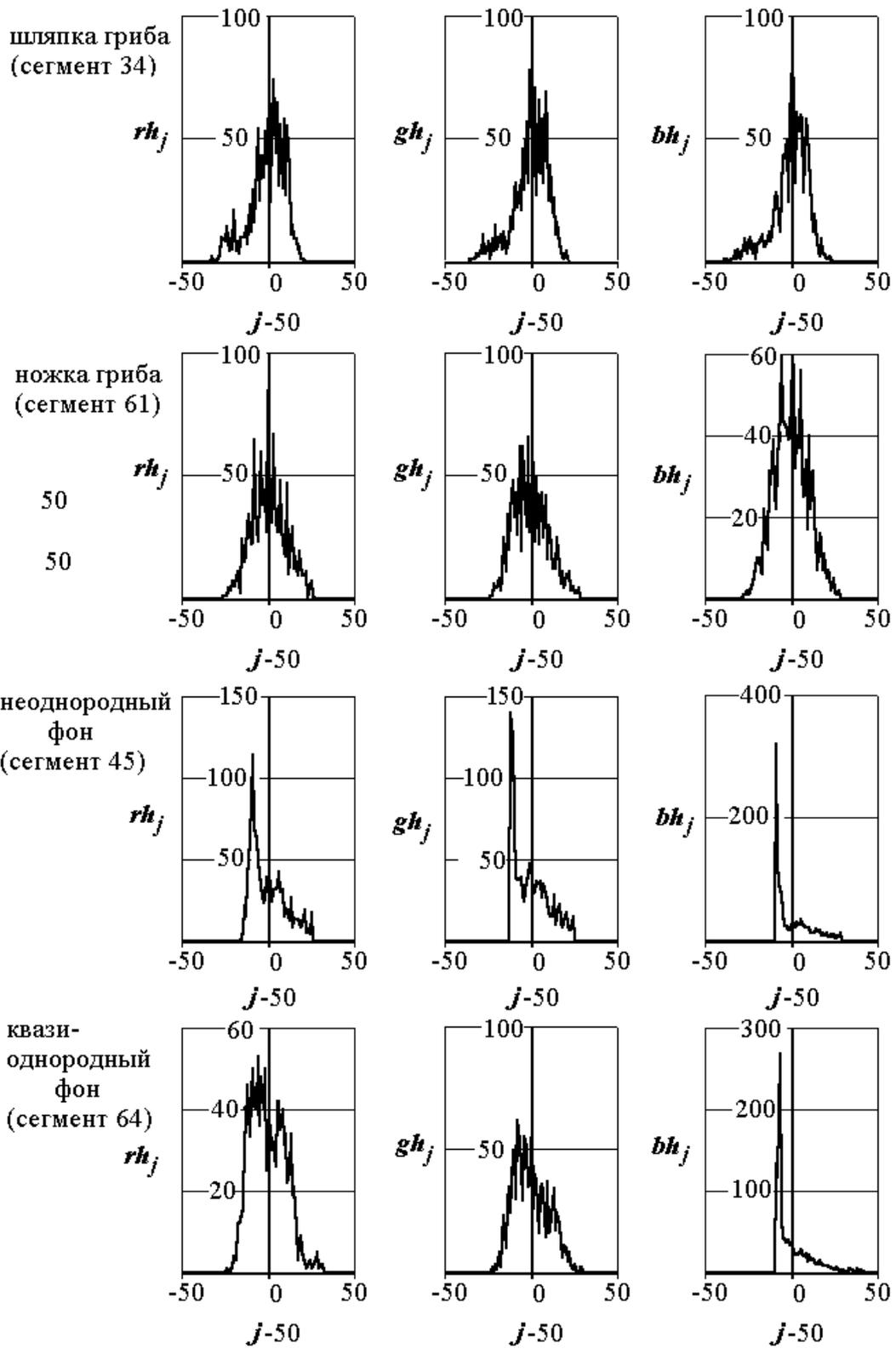


Рис. 1.3. Гистограммы rh , gh , bh для выделенных объектов

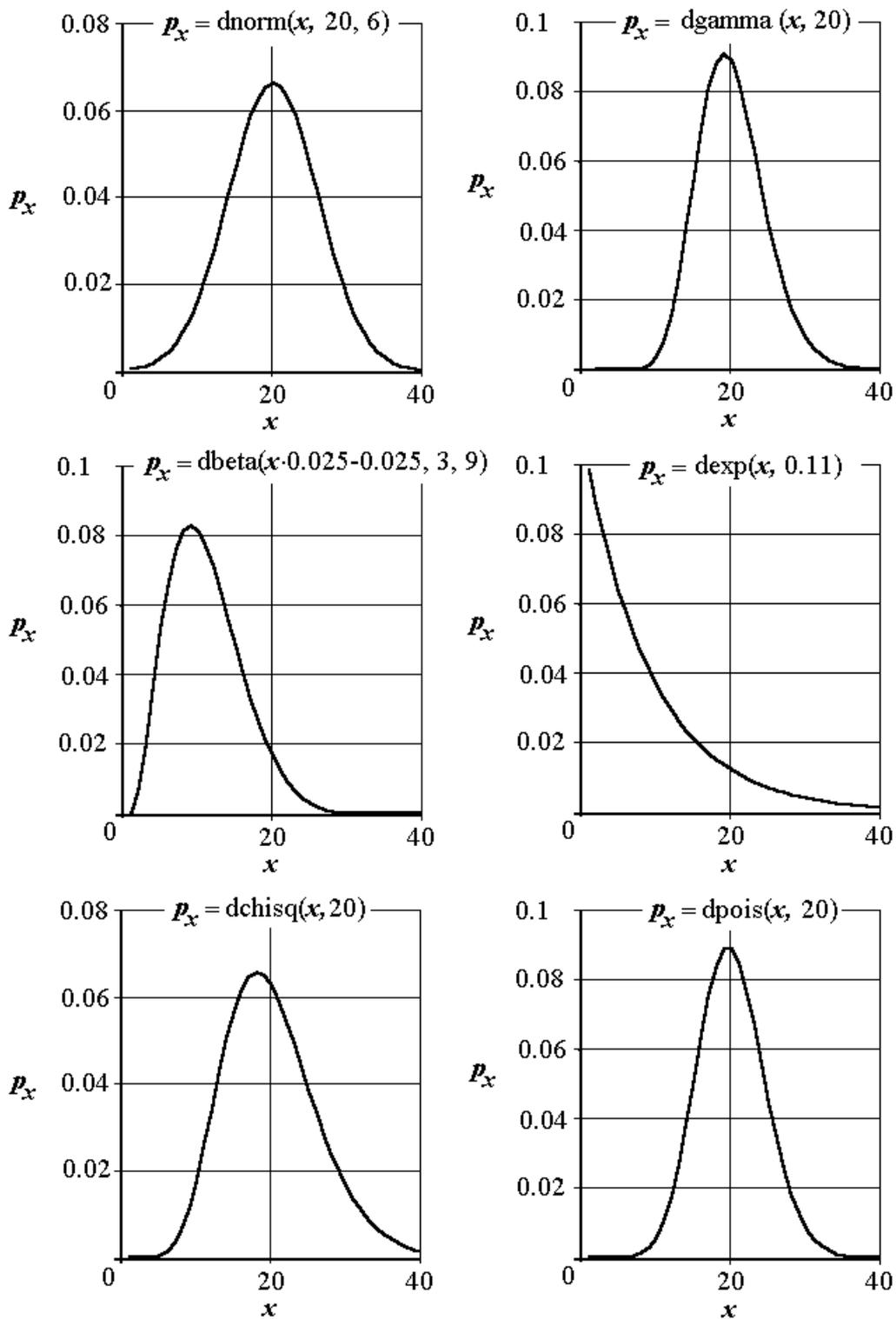


Рис 1.4. Генерация функций плотности распределения

Для наглядности покажем результаты сравнения на искусственном примере. Сравним результат генерации случайных чисел в среде математического пакета с теоретической кривой плотности вероятности (в примере – плотность нормального распределения).

Сформируем вектор случайных чисел s , распределенных по нормальному закону с параметрами: математическое ожидание $Mean$, среднеквадратичное отклонение $Stdev$ и числом элементов вектора im (в примере $im = 1000$).

$$s_i = \text{rnorm}(im, Mean, Stdev)_i.$$

Построим гистограмму p_x при числе каналов, равном 40, и вызываем функцию нормального распределения p с теми же параметрами (рис. 1.5):

$$j = 1..40;$$

$$x_j = Mean + 0.15 \cdot Stdev \cdot (j - 20);$$

$$p_x = \frac{\text{hist}(x, s)}{im};$$

$$p_j = \text{dnorm}(j, Mean, Stdev).$$

Разность теоретических и практических результатов приведена на рис. 1.6. По этим данным можно оценить потери от использования выбранного закона распределения.

Величины потерь очень часто описываются симметричными относительно нуля функциями, например:

$$dp = k \sqrt{\sum_j (p_j - p_{x_j})^k},$$

где k – показатель степени, определяющий метрику оси отклонений.

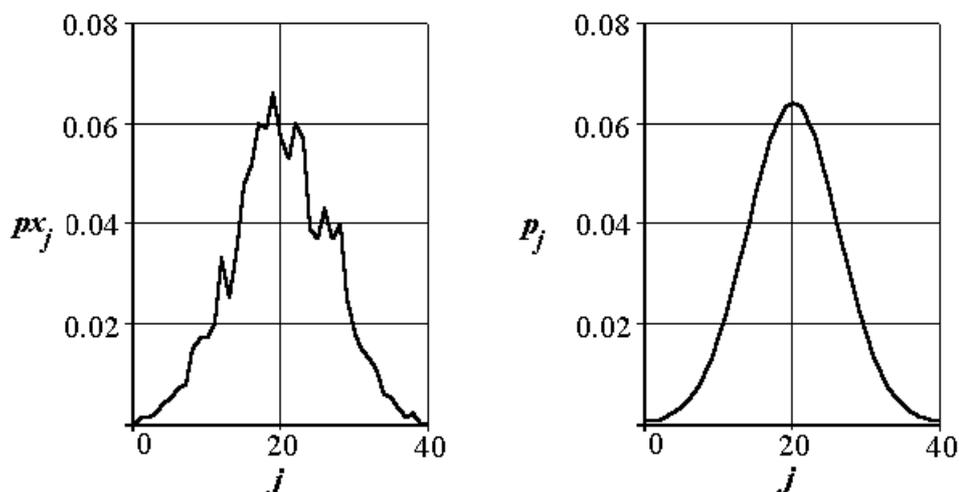


Рис 1.5. Гистограмма p_x и кривая плотности нормального распределения

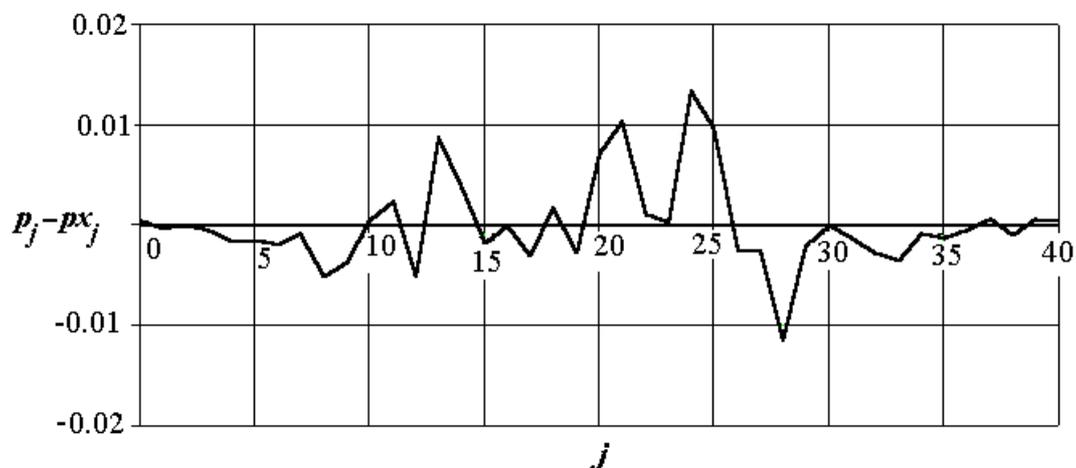


Рис 1.6. Разность экспериментально полученной гистограммы и теоретической кривой

При $k = 2$ (принятие решения по \min среднеквадратичного отклонения), для нашего случая $dp = 0.05$.

В практике распознавания образов широко используется представление описаний классов в производных пространствах. В данной практической работе рассматривается поиск отличительных признаков объектов в области пространственных частот. На рис. 1.7 представлены - модуль частотного спектра неоднородного сегмента (45) и его сечение.

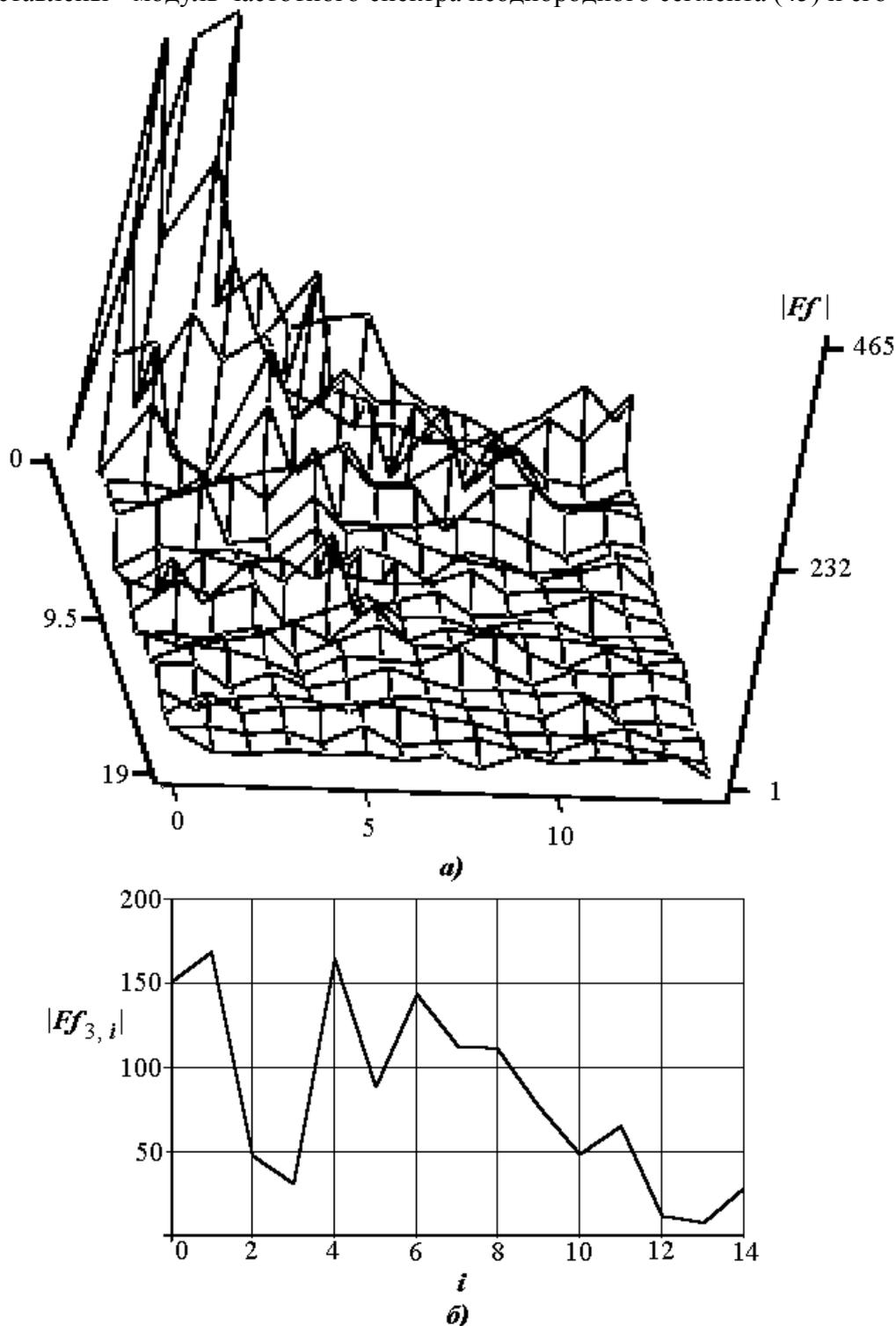


Рис. 1.7. Фурье-образ R – составляющей сегмента неоднородного фона

Вызов функции преобразования Фурье, как правило, затруднений не вызывает. Однако переход к пространственным частотам не всегда очевиден. Для приобретения навыков

работы с Фурье-образом рекомендуется сформировать изображение с гармонически изменяющейся яркостью и детально проанализировать метрику осей пространственных частот.

Например: задав размеры изображения в координатах j и i , вызовите функцию ImI периодом T и амплитудой $A < 127.5$. Выполните преобразование Фурье и определите масштабы осей пространственных частот.

$$ImI_{j,i} = A + A \cdot \sin\left(\frac{i}{T} \cdot 2 \cdot \pi\right).$$

Практическое задание

1. Сформировать рабочие файлы изображений объектов и помех (индивидуальное задание получить у преподавателя).
2. Сформировать рабочие сегменты, включающие в себя 2–3 объекта и фон.
3. Определить законы распределения сигналов в сегментах и их параметры для объектов и фона.
4. Определить доверительные интервалы оценок параметров выбранных сегментов.
5. Получить спектр пространственных частот для сегментов фона и объектов.

Содержание отчета

1. Описание содержимого рабочих сегментов.
2. Математические формулировки полученных законов распределения и числовые значения их параметров.
3. Полученные значения доверительных интервалов для оценок параметров.
4. Фурье-образ исследованных сегментов.

Практическая работа № 2

АЛФАВИТ ПРИЗНАКОВ, КОМПОНОВКА И МИНИМИЗАЦИЯ

Цель работы: ознакомление с методами анализа информационного наполнения массивов данных и команд и усвоение методики формирования компактного алфавита признаков.

Общие сведения

Назначение признака элементом алфавита признаков сопровождается присвоением ему коэффициента значимости, обычно за 1 принимается значимость наиболее мощного признака (при вычислениях в целых числах для уменьшения времени расчетов верхнее значение будет равно 255).

Признаки выбираются итерационной процедурой, исходя из априорных сведений об объектах, фоне и результатах сравнения перекрытий описаний классов. Сравнение проводится при отсутствии корреляции между признаками на числовой оси признака или в гиперпространствах коррелированных признаков.

Понижение размерности исследуемого пространства признаков – классическая задача в теории распознавания образов. В технических приложениях ее сводят к задаче построения преобразующего фильтра, выходной вектор сигнала которого имеет заметно меньшую размерность, чем входной вектор. Классификация в пространстве меньшей размерности выполняется быстрее и проще.

В технике наиболее распространенным принципом понижения размерности является преобразование исследуемого пространства в пространство базисных функций (тригонометрических, экспоненциальных, дельта функций). Выбор типа базисных функций основывается на понятии расстояния между различаемыми классами в новых пространствах. Важную роль играют априорные сведения об исследуемых объектах и их образах. При достаточном объеме данных можно решать задачу так, чтобы, сохранив вероятность правильного принятия решения о принадлежности объекта к собственному классу, получить новое описание в базисных функциях минимального размера.

Практически правило сжатия гиперпространства базисных функций, содержащего образ исходного вектора, может быть сформулировано для случая непересекающихся классов следующим образом:

- сжатие допустимо до тех пор, пока не пересекутся крайние точки доверительных интервалов классов, дополненные доверительными зонами крайних объектов в соседних классах (рис. 2.1).

В исходном описании изображений первой процедурой обычно является процедура укрупнения пикселя. Если отсутствует отработанный алгоритм описания классов в пространстве признаков, то целесообразно ее проводить почти до уровня сохранения различия объектов на изображении при решении принимаемом человеком. Данная процедура проводится и в пространствах базисных функций, например, в Фурье-отображениях сегментов. В простейшем случае укрупнение пикселя – это выборка с массива отсчетов в массив меньшей размерности. Ниже приведена процедура формирования массива RQ из R . Размерность нового массива в $my \times mx$ меньше, чем исходного. На рис. 2.2 показаны изображения до и после сжатия (левый верхний угол). Возможность распознавания объектов не потеряна. Это подтверждает и вид сигналов снятых с одних и тех же участков изображений (рис. 2.3).

$$RQ_{\text{floor}(\frac{y}{my}), \text{floor}(\frac{x}{mx})} = R_{y,x}$$

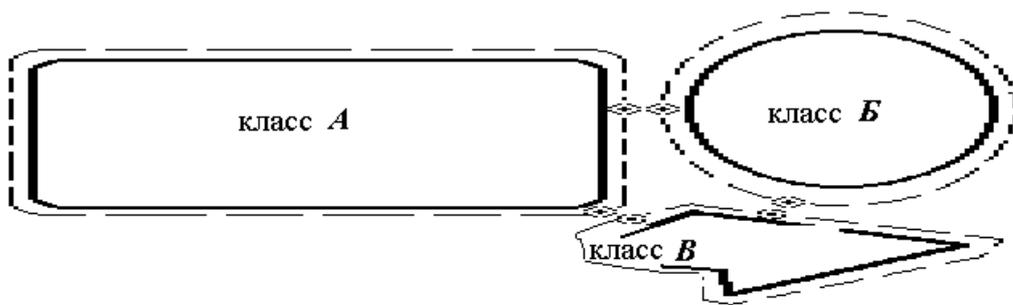


Рис. 2.1. Сжатие описаний пространства признаков до соприкосновения доверительных интервалов границ классов (A и B)

Более корректно операция укрупнения пикселя проводится через сегментацию и идентификацию миниобъектов внутри сегмента (например, при распознавании чертежей не уничтожаются последние пиксели линий).



$$m_x = 4, \quad m_y = 7.9$$

Рис. 2.2. Укрупнение пикселя в 4 раза по x и примерно в 8 раз по y

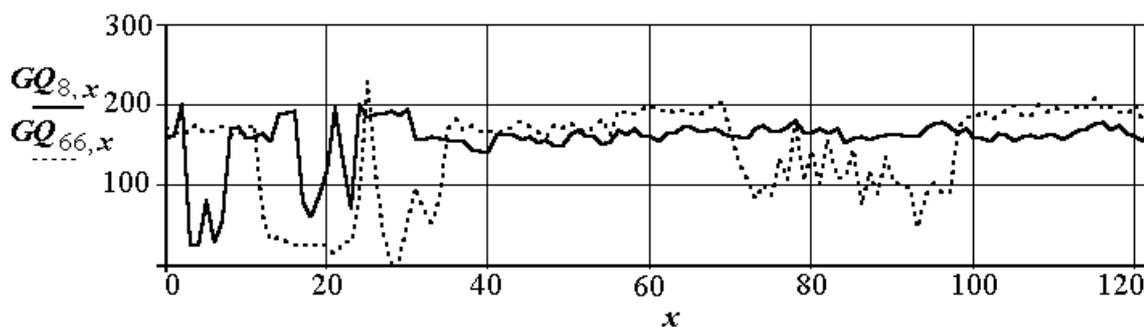


Рис. 2.3. Сигналы: после сжатия – строка 8 и до сжатия – строка 66

Следующая процедура – уменьшение размерности описания пикселя. Обычно исходное описание в 3-цветной модели имеет размер 24, 30, 48 бит в зависимости от разрядности систем оцифровки аналоговых сигналов.

Уменьшение разрядности линейным, нелинейным масштабированием сигналов цвета или адаптивным выбором цветов, сохраняющих межклассовую специфику объектов, позволяет снизить объем описания пикселя в несколько раз. На рис. 2.4-2.8 показаны результаты выполнения этапов уменьшения разрядности описания пикселя.

На первом этапе находятся гистограммы R , G , B составляющих (рис. 2.4).

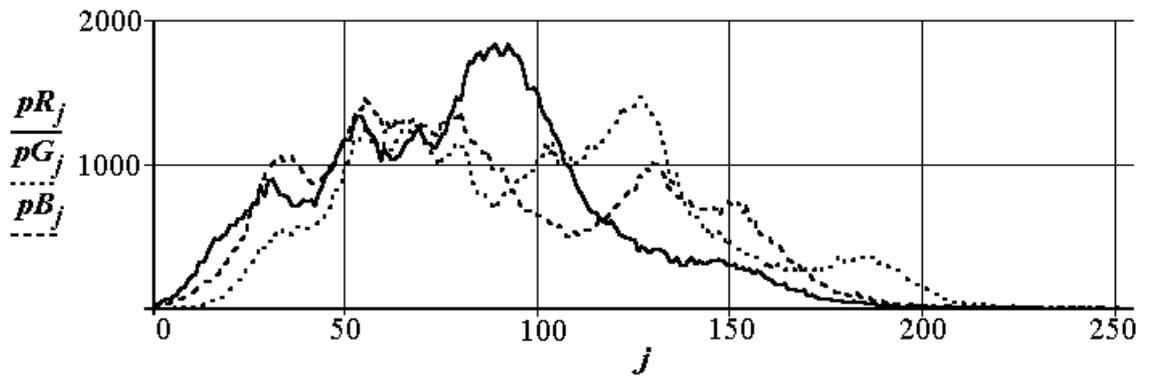


Рис. 2.4. Гистограммы R , G , B составляющих изображения кисти винограда

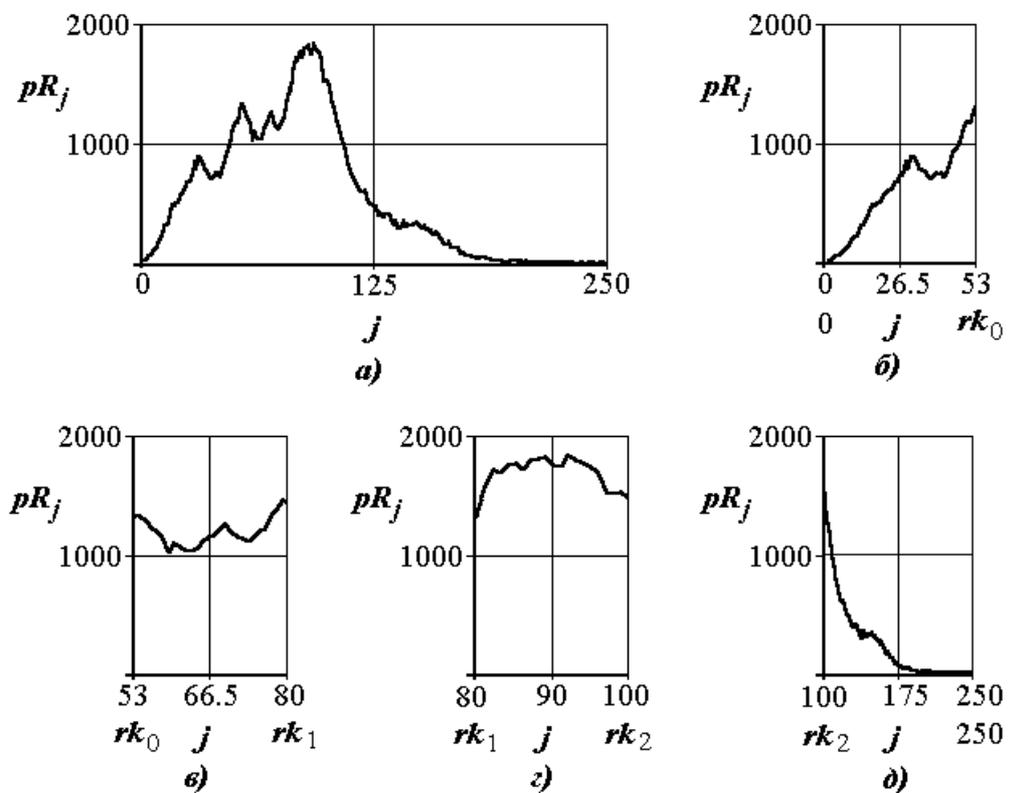


Рис. 2.5. Выделенные участки гистограммы R составляющей:
 а) вся зона, б), в), г), д) зоны содержащие одинаковое число пикселей

Далее задается правило разбиения осей на участки (исходное описание 8 бит – на составляющую).

В нашем случае отработано 2 алгоритма с преобразованием 2-3-2 т. е. 7-битном и 1-2-1 т. е. 4-битном представлении выходного описания пикселя. Результат разбиения оси R на четыре участка с равным количеством пикселей отображен на рис. 2.5. Границы участков служат уровнями логических компараторов, перекодирующих описание пикселей.

На рис. 2.6 и 2.7 приведены графики формирования выходного кода k по уровням R , G , B составляющих.

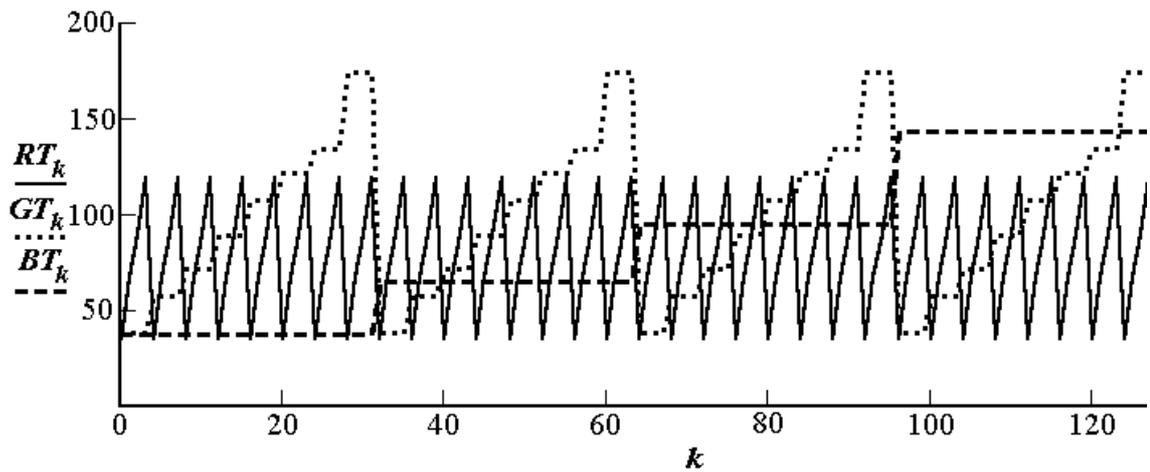


Рис. 2.6. Графики перекодировки RGB описания пикселя (возвращается k) для выходного описания 2-3-2 бита

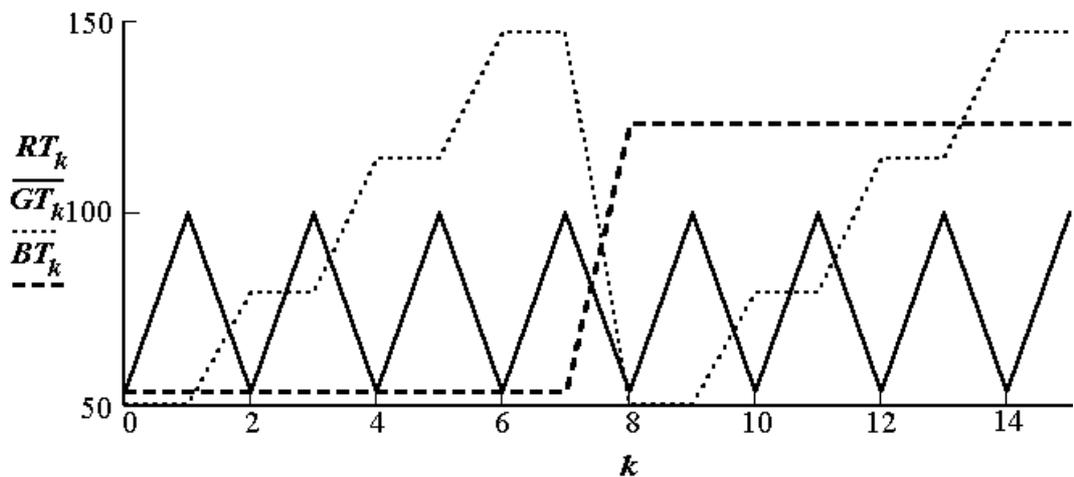


Рис. 2.7. Графики перекодировки RGB описания пикселя (возвращается k) для выходного описания 1-2-1 бит

На рис. 2.8 показан результат перекодировки 24 бита в 7 бит без потери различия объектов. В варианте 24 в 4 бита часть объектов стали неразличимы.

Предельным вариантом сжатия является бинаризация описания, т. е. представление яркости и цвета пикселя нулем или единицей. Выбор порога в бинаризации достаточно сложная процедура. В простейшем случае величина порога задается фиксированной по всему полю изображения. На рис. 2.9 показаны сигнал по строке и результат бинаризации изображения текста низкого качества при величине порога $r_s = 117$. Отдельные элементы букв утеряны (буква "н" в 6 строке не отличается от буквы "п") или стали не различимы по направлению элементов (ряд букв "и" в 4, 5 строке не различимы от буквы "н") и т. п.

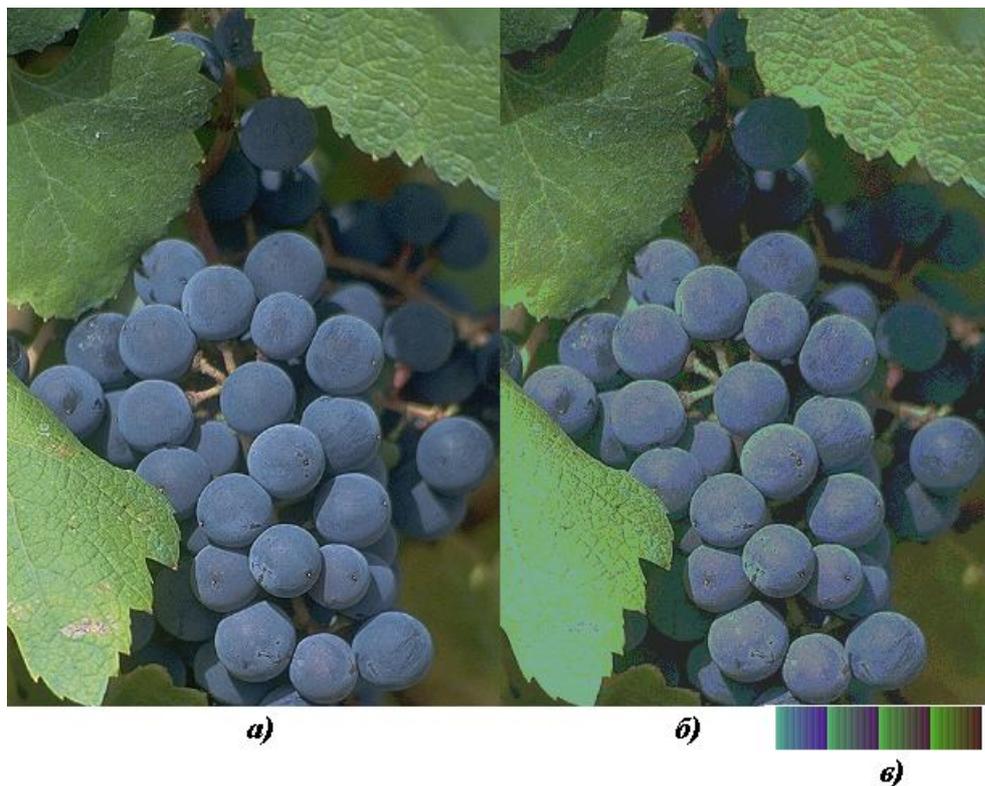
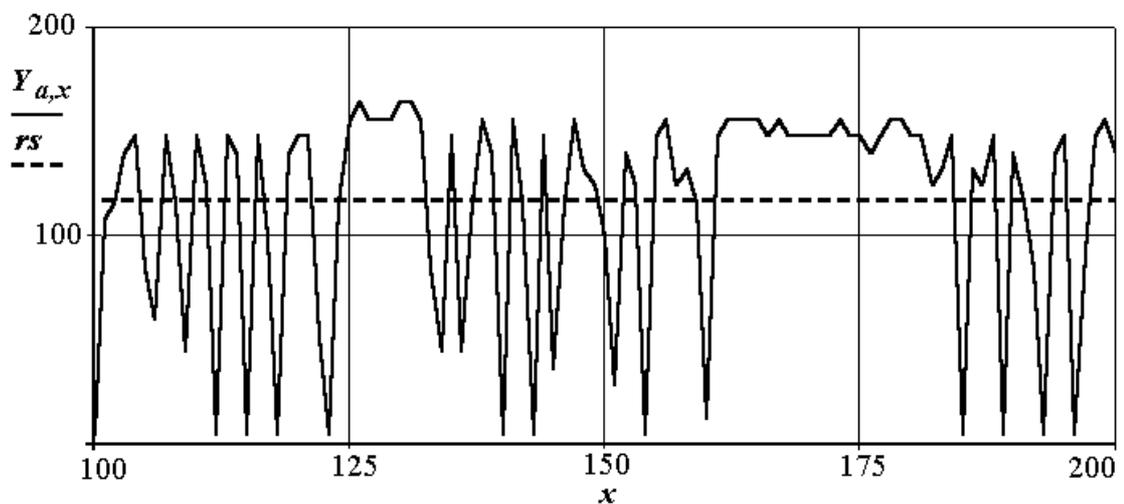


Рис. 2.8. Изображение кисти темного винограда на фоне листвы:
 а) до перекодировки; б) после перекодировки; в) новая палитра (2-3-2)



$a = 33$
 $rs = 117$

со средними характеристиками, отличать один случайный процесс от другого, предсказывать поведение случайного процесса. В узком смысле закон больших чисел — это теоремы, устанавливающие количественные соотношения для приближения средних характеристик большого числа опытов к математически точным характеристикам. Закон больших чисел — это теоремы, основанные на теории множеств и меры, предназначенные для определения числовых характеристик случайных величин.

Рис. 2.9. Бинаризация изображения низкого качества с фиксированным порогом

Несколько лучшие результаты получаются при плавающей по полю величине порога. Она может выбираться в процентах от усредненного сигнала в сегменте, средним элементом которого является бинаризуемый пиксель. Величина процента также может быть адаптивно изменяемой, в зависимости от типа сегмента, места пикселя в сегменте и т. п. Например, если W — исходное изображение, w — бинаризованное изображение, y, x — координаты

бинаризуемой точки, x_s , y_s – размеры сегмента, h_x , h_y – шаг смещения сегмента, n_x – число сегментов по x , то правило бинаризации можно записать следующим выражением:

$$w_{y,x} = \text{if} \left(W_{y,x} \leq m \left\lfloor \frac{x-x_s}{h_x} \right\rfloor + \left\lfloor \frac{y-y_s}{h_y} \right\rfloor \cdot n_x \right), 0, 255$$

где m_j – среднее, медиана или мода j -го сегмента. На рис. 2.10 показан результат бинаризации изображения текста с $y_s=30$, $x_s=20$, $h_y=20$, $h_x=20$, $n_x=18$ и $m_j = \text{mean}(w(j))$.

со средними характеристиками, отличать один случайный процесс от другого, предсказывать поведение случайного процесса. В узком смысле закон больших чисел — это теоремы, устанавливающие количественные соотношения для приближения средних характеристик большого числа опытов к математически точным характеристикам. Закон больших чисел — это теоремы, основанные на теории множеств и меры, предназначенные для определения числовых характеристик случайных величин.

Рис. 2.10. Бинаризация изображения низкого качества с плавающим порогом

Ряд символов стали более близки по форме к исходным, чем в случае бинаризации с фиксированным порогом, однако 100 % качества распознавания только по форме символа добиться не удалось.

Практическое задание

1. Выбрать рабочие изображения и перечень распознаваемых объектов.
2. Минимизировать размеры изображений и их цветовое представление.
3. Определить Фурье-образы 2 изображений, минимизировать их описание.
4. Сформировать алфавит признаков.
5. Определить расстояние между классами в пространстве двух признаков.

Содержание отчета

1. Перечень объектов и признаков.
2. Результаты минимизации по размерам пикселей и цветовой кодировке.
3. Полученный Фурье-образ одного из объектов.
4. Алфавит признаков.

Практическая работа № 3

РАСПОЗНАВАНИЕ ОБЪЕКТОВ В КОДОВЫХ ПОСЛЕДОВАТЕЛЬНОСТЯХ

Цель работы: Освоение методики работы с описаниями естественных и виртуальных объектов, освоение корреляционных алгоритмов распознавания образов приобретение навыков работы с описанием ситуаций, графическим отображением функций предпочтения.

Общие сведения

Идентификация естественных объектов по их геометрическим параметрам (размерам, форме, положению), цвету широко используется во многих отраслях техники, в медицине, криминалистике и т. п. Многообразие объектов и специфика их применения порождают и множество алгоритмов их распознавания.

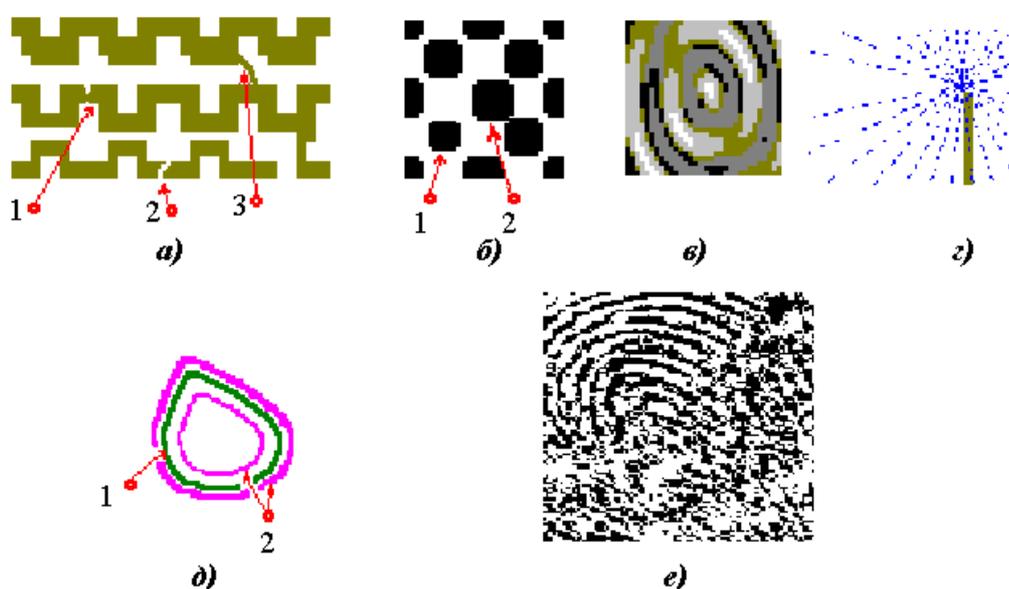


Рис. 3.1. Примеры объектов

В микроэлектронике это дефекты топологии технологических компонентов (фотошаблонов, масок и т. п.), а также самих изделий (металлические проводники, слои СБИС и т. п.). На рис. 3.1 *а*) и 3.1 *б*) приведены участки топологий с дефектными зонами: *а*) 1 – дефект проводника, не нарушающий электрической цепи, но понижающий надежность изделия, 2 – дефект проводника, нарушающий работу изделия, 3 – ошибочная перемычка (ус), замыкающая проводники; *б*) 1– уход размеров, который может изменить параметры изделия (например, частотные), 2 – смещение элемента топологии с нарушением изоляции. В реальных задачах необходимо анализировать комбинации дефектов.

Трехмерные объекты также часто распознаются по двумерным изображениям. Определение качества паянных соединений, например, контактных площадок, пропайка выводов микросхем проводится по девиации яркости, зависящей от формы фрагментов застывших волн припоя (рис. 3.1 *в*).

В физическом эксперименте следы частиц (подобно рис. 3.1 *г*) позволяют определить параметры источника. В многих случаях контур объекта 1 должен принадлежать некоторой заданной зоне 2 (рис. 3.1 *д*), это характерно для биологических, медицинских и отдельных технических задач. В криминалистике поиск объектов проводится нередко по сходству геометрии линий (рис. 3.1 *е*).

Исходной операцией процедуры распознавания во всех указанных случаях является определение геометрических параметров объектов. В практической работе рассматриваются две группы алгоритмов:

- упрощенные с определением искомых параметров при погрешностях сравнимых или больших, чем эквивалентный размер пикселя;
- точные с интегрирующими цифровыми фильтрами и аппроксимацией результата с разрешением в десятые и сотые доли эквивалентного размера пикселя.

Упрощенное определение геометрических параметров

Данная операция широко используется в системах обнаружения, как первая процедура при локализации объектов. Полное решение задачи распознавания, как правило, содержит много процедур, в дальнейшем часть из них будем опускать, останавливаясь только на ключевых. Рассмотрим простейший пример. Пусть необходимо распознать объекты (лодки) по площади их бортов. На рис. 3.2 приведено изображение борта лодки и сигнал по столбцу с номером 212, взятый из анимационного фильма, показывающего изменение видеосигнала при смещении анализируемого столбца по кадру. На нем отчетливо виден всплеск сигнала при переходе от светового сигнала с водной поверхности на борт лодки и спад сигнала в зоне перехода с борта лодки на водную поверхность. Анализ сигналов показывает возможность определения площади борта как сумму расстояний по столбцам от первого яркостного всплеска до последнего провала сигнала (по границам контуров борта лодки).

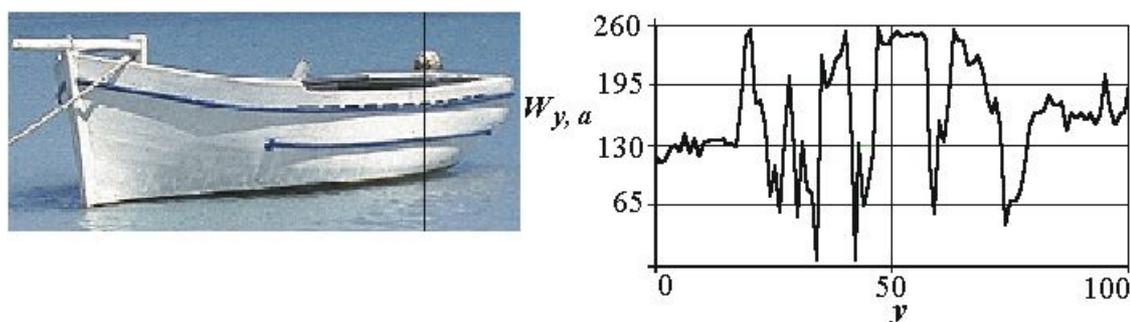


Рис. 3.2. Изменения сигнала по выделенному черной линией столбцу

Зададим пороги яркости – 220 и 70 (встреча с бортом и с тенью от борта). Зададим координату по y – 50. Эта величина будет использована для ограничения зоны встречи с бортом. В столбцах, в которых процедуры не обнаружили наличие борта вставим 255. Из исходного массива W сформируем (с учетом порогов) рабочие массивы WH , WL . Найдем минимум и максимум по столбцам и скорректируем полученные данные, устранив недостоверные результаты. На рис. 3.3 приведены выходные функции.

$$\begin{aligned}
 WL_{y,x} &= \text{if}(W_{y,x} \leq 220, 255, y); \\
 WH_{y,x} &= \text{if}(W_{y,x} \geq 70, 0, y); \\
 wl(x) &= \text{submatrix}(WL, 0, ym - 1, x, x); \\
 yl_x &= \min(wl(x)); \\
 wh(x) &= \text{submatrix}(WH, 0, ym - 1, x, x); \\
 yh_x &= \max(wh(x)); \\
 yl_x &= \text{if}(yl_x \geq 50, 255, yl_x); \\
 yh_x &= \text{if}(yh_x \leq 50, yh_{x-1}, yh_x); \\
 yl_x &= \text{if}(yh_x = 0, 255, yl_x);
 \end{aligned}$$

$$yh_x = \text{if}(yl_x = 255, yl_x, yh_x).$$

Разностная функция (рис. 3.3 в) показывает высоту борта лодки. Найдем интеграл от нее по x :

$$s = \sum_x (yh_x - yl_x).$$

Расчет дает $s = 11470$ квадратных пикселей, это и есть грубая оценка площади борта лодки.

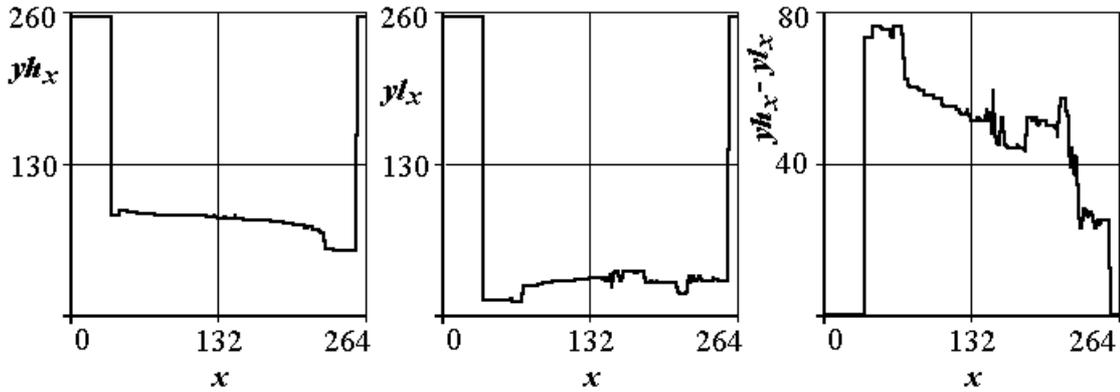


Рис. 3.3. Определение площади по границам контуров

Второй пример: определение местоположения объектов, выделяющихся из общего фона по яркости (рис. 3.4). Зададимся минимальными размерами объектов по $x - xs=15$, по $y - ys=5$. Сформируем сегмент с указанным размером и переместим его с шагом по x и y равным единице от начала до конца изображения, вычисляя каждый раз среднее значение сигнала в сегменте d (рис. 3.5).

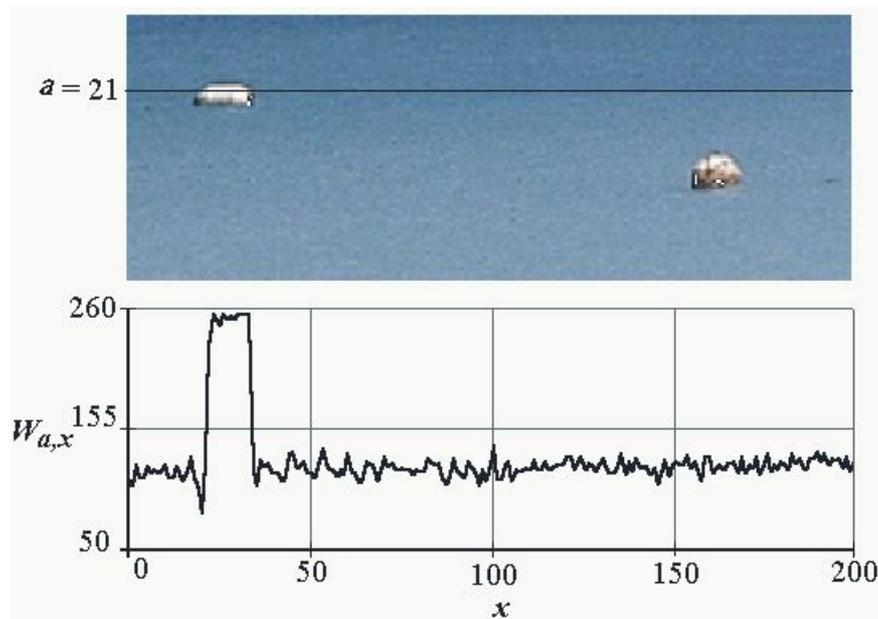


Рис. 3.4. Локация объектов

Такой прием позволяет увеличить соотношение сигнал/шум и надежнее распознать объекты по энергии сигнала. Координата максимума j_m (номер сегмента) показывает

положение наиболее освещенного объекта. Переведем ее в физические координаты объекта на изображении y_0, x_0 :

$$y_0 = \text{floor}\left(\frac{jm}{nx}\right) + 0.5 \cdot ys ;$$

$$x_0 = \text{mod}(jm, nx) + 0.5 \cdot xs ;$$

где nx – число сегментов в строке. Расчеты в приведенном примере показали оценки координат первого объекта $y_0 = 22.5$ и $x_0 = 26.5$. Продолжая поиск максимума вне зоны обнаруженного объекта по описанной методике получаем оценки координат второго объекта $y_0 = 42.5$ и $x_0 = 168.5$. Более корректное выполнение операций поиска координат требует проведения низкочастотной фильтрации сигнала в сегменте для устранения тренда, обусловленного конкретным положением источника света.

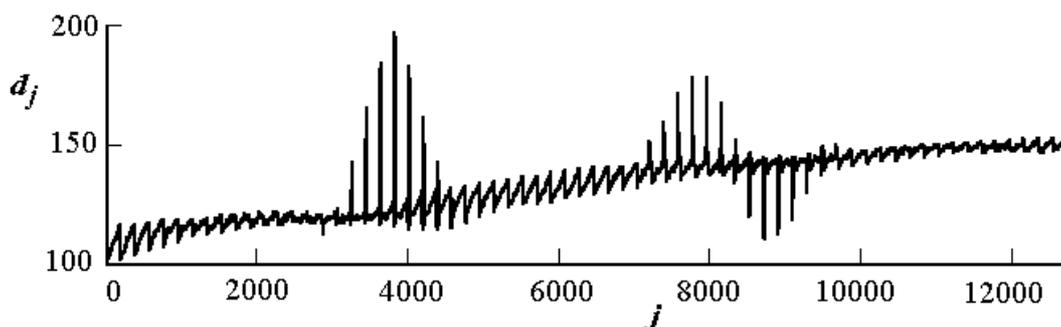


Рис. 3.5. Бегущее среднее значение сигнала в сегменте

Точное определение геометрических параметров

Поясним последовательность и специфику операций на примере. Пусть требуется определить координату по строке и размер объекта, приведенного на рис. 3.6. Там же прорисован вид сигнала по строке, выделенной на изображении черной линией.

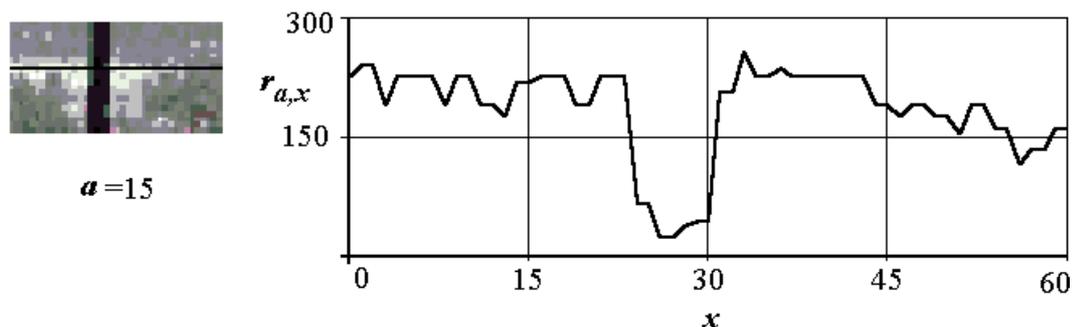


Рис. 3.6. Объект на естественном фоне

Вначале выделим объект (по априорным данным его ширина ≥ 7 пикселям). Введем фильтр $F1$ (рис. 3.7) и профильтруем данные массива r по строке a . Выходной сигнал $f1$ (рис. 3.8) найдем по формуле

$$f1_x = \sum_{i=-5}^5 r_{a,|\text{mod}(x+i, xm)|} \cdot F1_{i+5} ;$$

где $F1_i = \text{if}(|i-4| \leq 3, 1, 0)$, xm – размер строки, а операция взятия адреса точки на строке по модулю размера строки исключает выход за ее пределы.

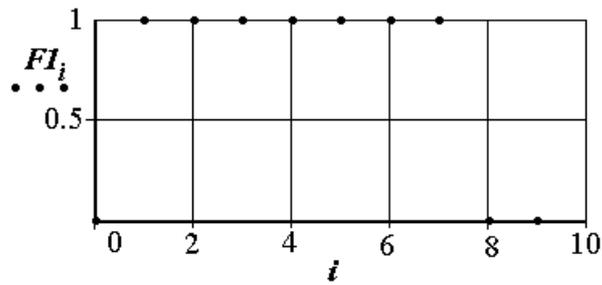


Рис. 3.7. Прямоугольный фильтр, селектирующий объекты по энергии

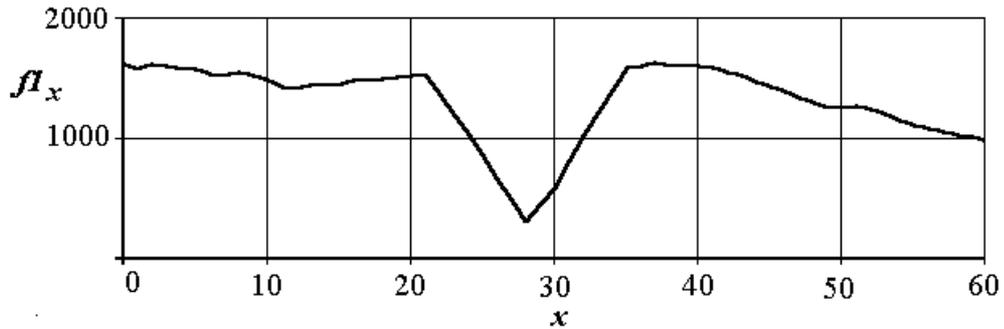


Рис. 3.8. Сигнал по строке после фильтрации

Как видно из графиков (рис. 3.6 и 3.8), улучшилось соотношение сигнал /шум. Оценить улучшение количественно можно оценив шум вне зоны сигнала и размах сигнала. Найдем si – соотношение сигнал/шум в децибелах для исходного и отфильтрованного сигналов по формуле

$$si = 20 \cdot \log \left(\left| \frac{r_{a,b} - r_mean}{r_stdev} \right|, 10 \right);$$

где точка b – положение максимума сигнала, r_mean , r_stdev – оценки среднего и среднеквадратичного отклонения на участке, предшествующем сигналу. Расчеты показали увеличение si после фильтрации практически на 7 децибел.

По положению минимума не сложно оценить приблизительно координату объекта x_grubo . Для более точного расчета изменим форму фильтра (рис. 3.9), формулу свертки оставим прежней.

$$F2_i = \text{if}(i \leq 3, 1, \text{if}(i \leq 7, -1, 0)).$$

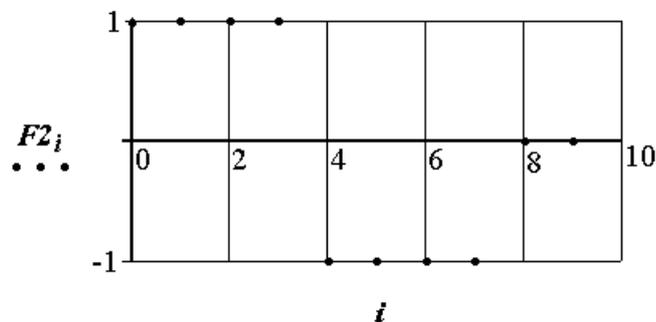


Рис. 3.9. Прямоугольный фильтр для поиска координаты объекта

Выходной сигнал $f2$ представлен на рис. 3.10. На рисунке изображен сигнал по полной строке и на ее участке в зоне пересечения нуля. Точка пересечения сигналом нуля является точкой, в которой сумма участка исходного сглаженного сигнала в положительном крыле

фильтра равна сумме в отрицательном крыле фильтра, т. е. фильтр делит сигнал на две равные по сумме составляющие и данная точка является искомой координатой.

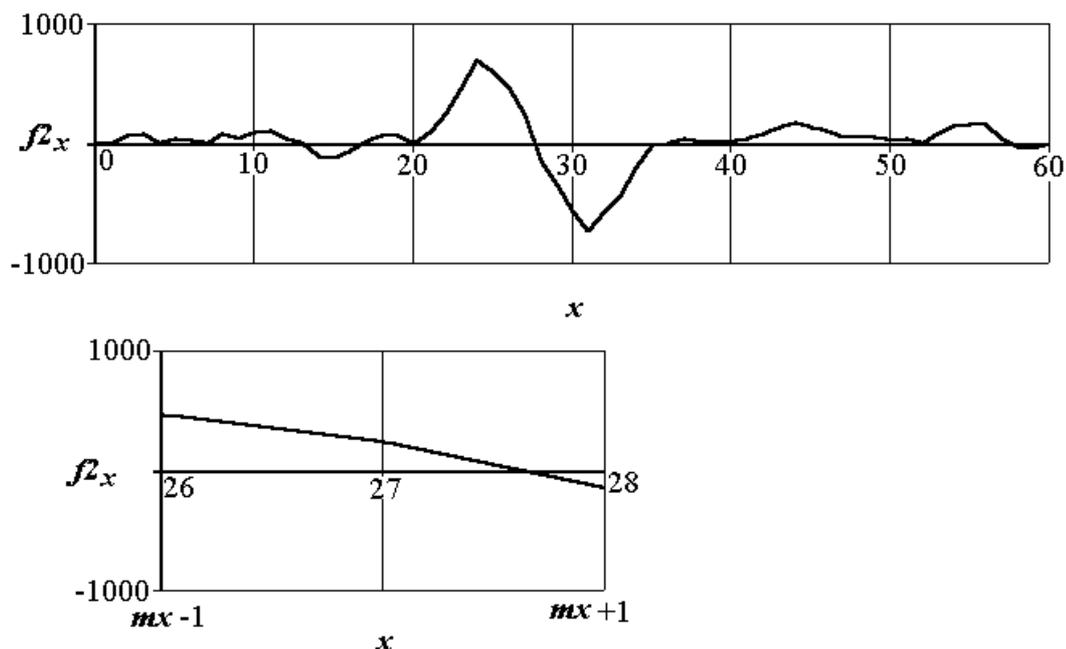


Рис. 3.10. Выходной сигнал при поиске координаты объекта

Уточним оценку координаты в зоне x_grubo , используя линейную интерполяцию

$$i = x_grubo - 4..x_grubo + 4;$$

$$x_i = \text{if}(f_{2i} > 0, i, 0);$$

$$mx = \max(x);$$

$$x0 = mx + \frac{f_{2mx}}{f_{2mx} - f_{2mx+1}}.$$

Новая оценка координаты – $x0 = 27.6$.

Для определения размера приблизим форму фильтра к форме второй производной от фронта сигнала. Математические пакеты позволяют таблично задать вектор, введем

$$F3 = \{0, 0, 0, 1, -1, 0, 0, 0, 0, 0, 0\}$$

(форма фильтра изображена на рис. 3.11), и вычислим свертку (выходной сигнал - на рис. 3.12). Расстояние между положениями максимума и минимума по оси x в выходном сигнале равно 7, это и есть искомый размер.

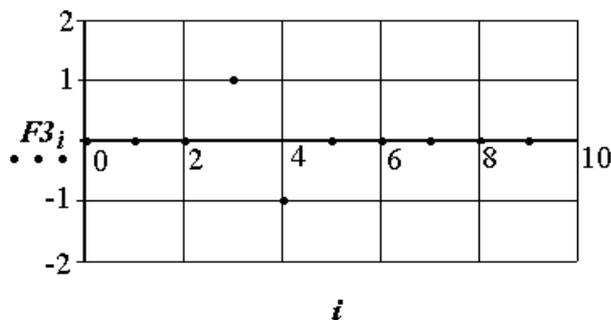


Рис. 3.11. Цифровой фильтр для определения размера доминирующего объекта

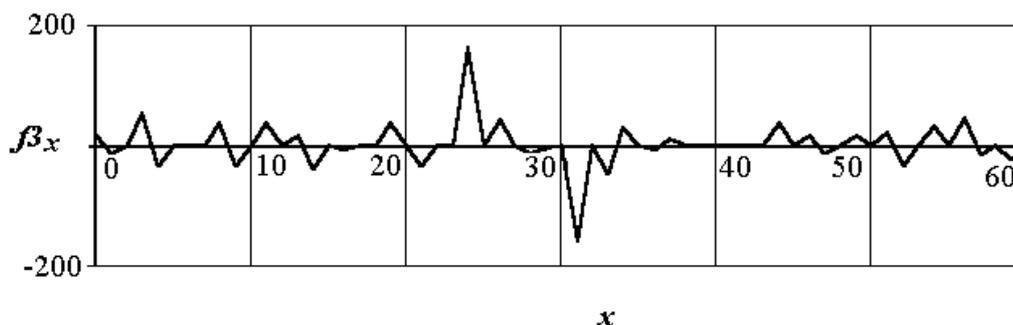


Рис. 3.12. Выходной сигнал при оценке ширины объекта

Приведенные выше фильтры являются квазиоптимальными. Для линейных алгоритмов определения координат, размеров объектов, они дают достаточно хорошие результаты.

Более корректное решение предполагает распознавание формы сигнала, оценку его параметров, генерацию эталонного сигнала полученной формы, получение разностного сигнала при смещении эталонного вдоль распознаваемого и определение положения эталонного, при котором минимизируются штрафы за ошибки в оценке координат, размеров.

По сути, процедура решения задач распознавания объекта часто подобна раскрытию матрицы. Определившись с одним объектом для распознавания, Вы обнаруживаете задачу распознавания его компонентов и т. д. Описав геометрию объекта, необходимо распознать форму границы объекта, составить описания классов помех во всех случаях и т. п., распознать границу, определить грубо ее положение и уточнить координаты.

Наряду с анализом признаков одиночных объектов, часто требуется определить геометрические признаки и для групповых объектов. К ним относятся средние величины расстояний, средние размеры объектов, входящих в группу.

Геометрические параметры групповых объектов

Рассмотрим пример расчета геометрических характеристик изображения текста. Определим величину межстрочного интервала и координаты центров строк. На рис. 3.13 представлено изображение текста, сигнал по столбцу с номером a - Y . Однако на приведенном сигнале даже человек вряд ли сможет распознать строки. Как уже указывалось, возможность распознавания можно улучшить, интегрируя участки изображения, в данном случае в направлении, перпендикулярном направлению анализа. Вычислим интегральный сигнал $Y_2 = \sum_x Y_{y,x}$, он также приведен на рис. 3.13. На данном сигнале отчетливо видно хорошее различие заполненных черными точками текста участков.

со средними характеристиками, отличать один случайный процесс от другого, предсказывать поведение случайного процесса. В узком смысле закон больших чисел — это теоремы, устанавливающие количественные соотношения для приближения средних характеристик большого числа опытов к математически точным характеристикам. Закон больших чисел — это теоремы, основанные на теории множеств и меры, предназначенные для определения числовых характеристик случайных величин.

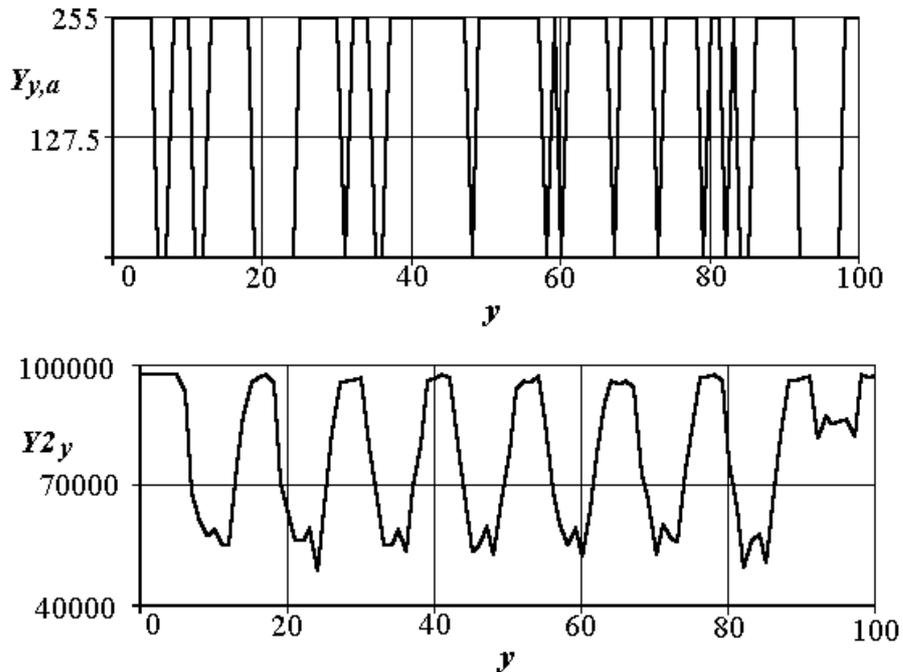


Рис. 3.13. Бинаризованный текст, сечение по столбцу и интегральный сигнал

Переходя в частотное пространство, определим величину интервала между строками. Уменьшим размер вектора до 64 (кратно 2 в степени n), и вычислим модуль $F2$ частотного спектра сигнала

$$Y2f_{j1} = Y2s_{j1} - \text{mean}(Y2s),$$

где $Y2s_{j1} = Y2_{j1}$.

$$F2_k = |F1_k|,$$

где $F1 = \text{fft}(Y2f)$ — Фурье-образ $Y2f$, а $k=0..c$ и $c = \text{last}(F1)$ — индекс последнего элемента вектора.

Вид спектра пространственных частот приведен на рис. 3.14.

Интервал между строками в пикселях h равен:

$$h = \text{floor} \left[\frac{2 \cdot c}{\max(f2)} + 0.5 \right],$$

где $f2_k = \text{if}(F2_k \neq \max(F2), 0, k)$ — массив нулей с координатой максимума частотного спектра. В нашем случае $h=13$.

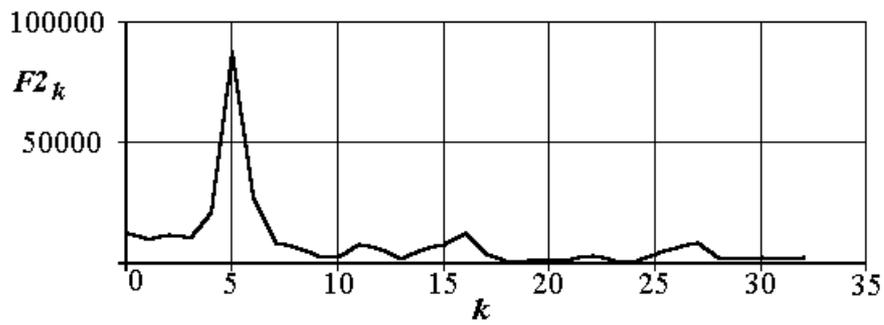


Рис. 3.14. Модуль частотного спектра интегрального сигнала (постоянная составляющая отфильтрована)

Для определения координат центров строк введем фильтр $F1$ (рис. 3.15):

$$F1_i = \text{if} \left[\left(i - \frac{h}{4} \right) \leq \frac{h}{4}, 1, 0 \right], \text{ где } i = 0..h + 2.$$

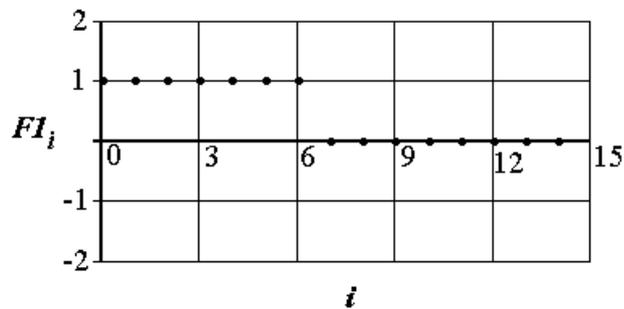


Рис. 3.15. Цифровой фильтр для определения координат центров строк

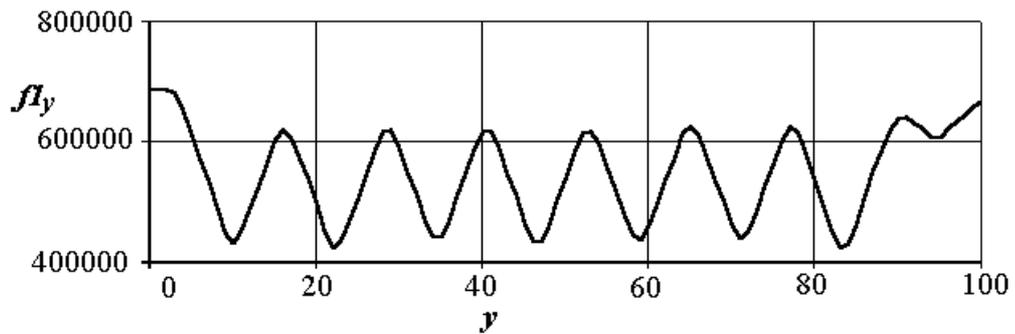


Рис. 3.16. Выходной сигнал после свертки $Y2$ с $F1$

Вид выходного сигнала свертки сигнала $Y2$ с $F1$ показан на рис. 3.16.

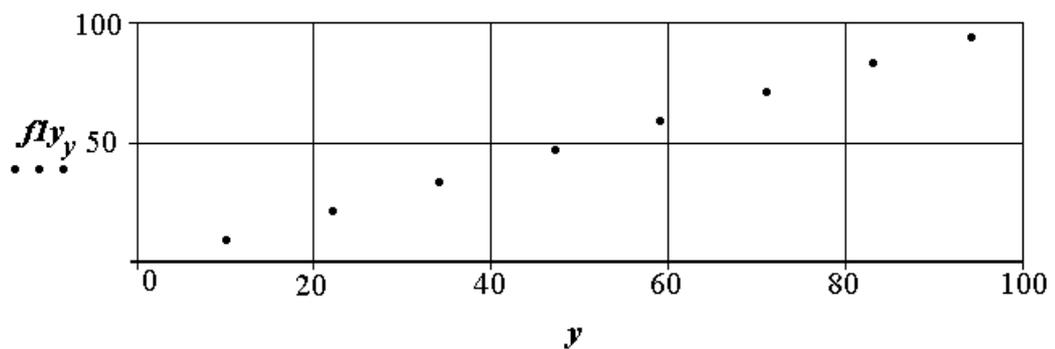


Рис. 3.17. Координаты центров строк

Составим вектор центров строк текста fly , найдя координаты вершин через упрощенное выражение:

$$fly_y = \text{if}[fI_{|y-1|} > fI_y, \text{if}(fI_{\text{mod}(y+1,ym)} > fI_{y,y,0}), 0];$$

где ym – число строк текста. Вид вектора представлен на рис. 3.17. Числа отличные от нуля являются координатами строк. Полученные числовые данные по положению строк текста приведены ниже.

$$\{10, 22, 34, 47, 59, 71, 83, 94\}$$

Возможности цветового различия тесно связаны с понятием спектра длин волн оптического излучения, как информационного параметра сигнала с элемента изображения. В целом ряде случаев этот параметр избыточен. Более приемлемо характеризовать объект совокупностью меньшего количества компонент

$$V_i = \int_{\lambda_1}^{\lambda_2} G_i(\lambda) \cdot W_j(\lambda) \cdot d\lambda,$$

где i – номер компонента (от 1 до N),

$G_i(\lambda)$ – спектральная чувствительность i – канала,

$W_j(\lambda)$ – спектральная плотность потока от j – го объекта попавшая в i – канал,

λ – длина волны оптического излучения,

$\lambda_1 \dots \lambda_2$ – рабочий диапазон длин волн.

Часто при распознавании можно отказаться от использования информации об общем уровне освещенности, перейдя в пространство цветов. Это понижает размерность исходного пространства на единицу и облегчает процедуру распознавания. Сформируем нормированные, не зависящие от общего уровня сигнала цветовые составляющие:

$$v_i = V_i / \sum_{i=1}^N V_i.$$

Величина N определяет размерность цветового пространства R^N до нормировки или R^{N-1} после нормировки ($\sum_{i=1}^N v_i = 1$). Часто говорят о N – компонентном цветовом

пространстве. Очень распространена трехцветная модель RGB в видимом диапазоне длин волн оптического излучения. Она отображает чувствительность рецепторов глаза человека. Для данного диапазона создана недорогая аппаратура считывания и генерации изображений, разработаны стандарты описаний файлов и имеется множество графических программных продуктов для работы с ними. Как правило, информационное наполнение изображений, полученных не в RGB пространстве, переносят в него для более естественного анализа.

Процедура отображения изображений в пространство RGB содержит однозначную операцию отображения пикселя, которая состоит в последовательности следующих шагов:

- формируются матрицы R , G , и B в двухмерном геометрическом пространстве;
- формируется новая матрица с числом столбцов равным числу уровней квантования первой цветовой составляющей и числом строк равным числу уровней квантования второй цветовой составляющей пикселя;
- матрица заполняется тегом отсутствия запроса ячейки, например, - 256, данное число выбирается вне зоны существования уровней квантования третьей цветовой составляющей пикселя;

- в матрицу по координатам описания первой и второй цветовой составляющих записывается значение третьей цветовой составляющей пикселя.

Эта процедура может быть распространена и на все изображение, однако пиксели, запрашивающие одинаковые координаты по значениям первой и второй цветовой составляющих, будут конкурировать друг с другом по записи. Преимущество отдается последнему запросу и информация о предыдущих запросах будет утеряна. Искусственно можно устранить этот недостаток проверкой тега запроса ячейки, и при наличии факта предыдущего запроса кодировать информацию в ячейке, например, формируя число в позиционной системе счисления с основанием, равным числу уровней квантования третьей цветовой составляющей пикселя +1 (для тега отсутствия запроса позиции). В дальнейшем мы будем допускать потерю части информации о предыдущих запросах.

На рис. 3.18 приведен вид зон существования объектов в *RGB* пространстве. В качестве исследуемого изображения принята комбинация поверхности горы и неба, т. е. двух классов.

Как видно, они образуют соприкасающиеся, достаточно компактные группы точек. Для сравнения на рис. 3.19 показано отображение цветового хаоса.

Матрица Z на рисунках заполнена по формуле

$$Z_{G_{y,x}, B_{y,x}} = R_{y,x}.$$

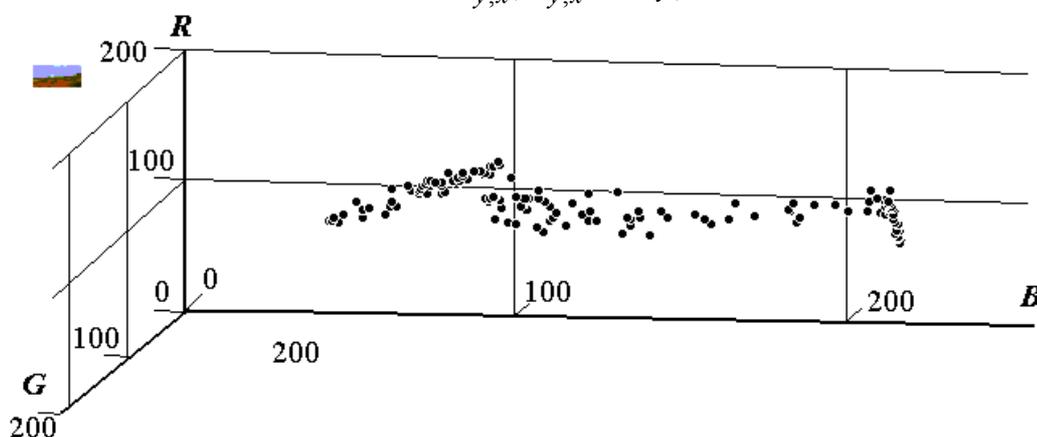


Рис. 3.18. Кластеры объектов в *RGB* пространстве

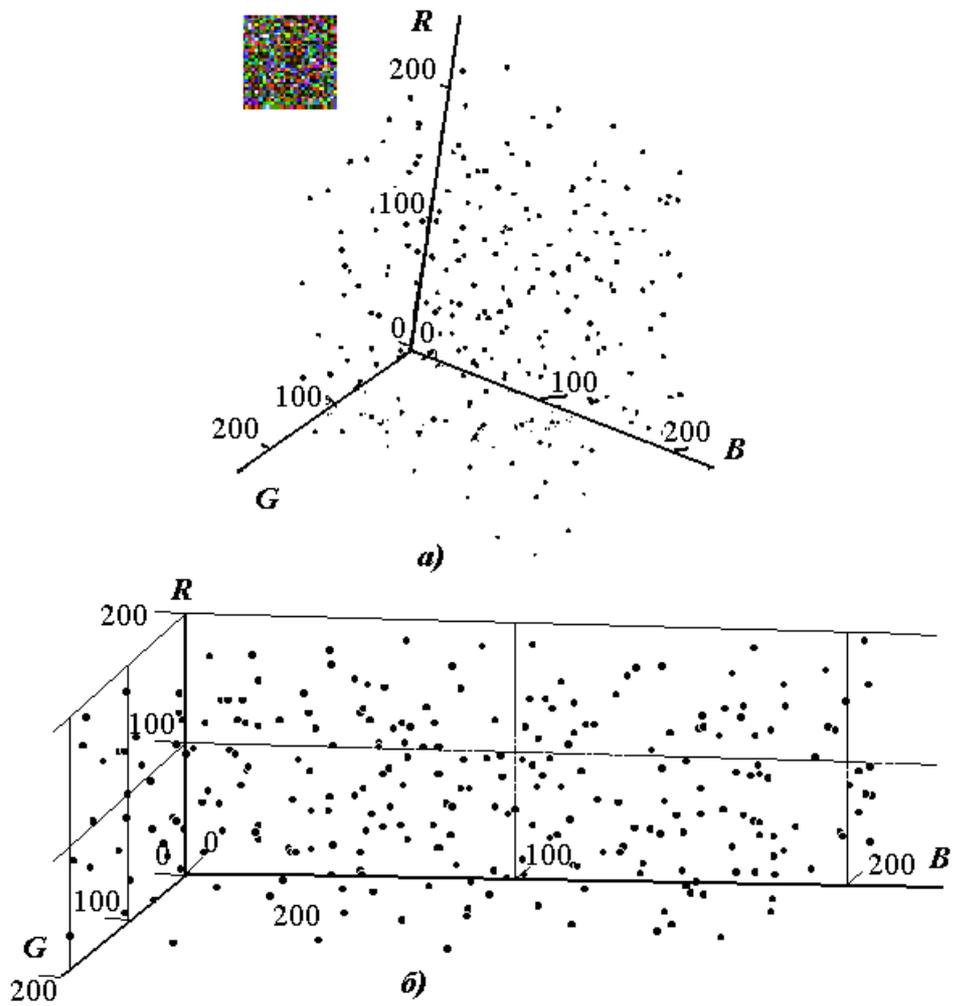


Рис. 3.19. Варианты отображение в RGB – пространстве цветового “хаоса”

Для упрощения в качестве тега отсутствия запроса ячейки принято число “0”, а информация о точках с $R_{y,x} = 0$ утеряна. Имеет место и маскирование пикселями с $R_{y,x} = 0$ предыдущих записей.

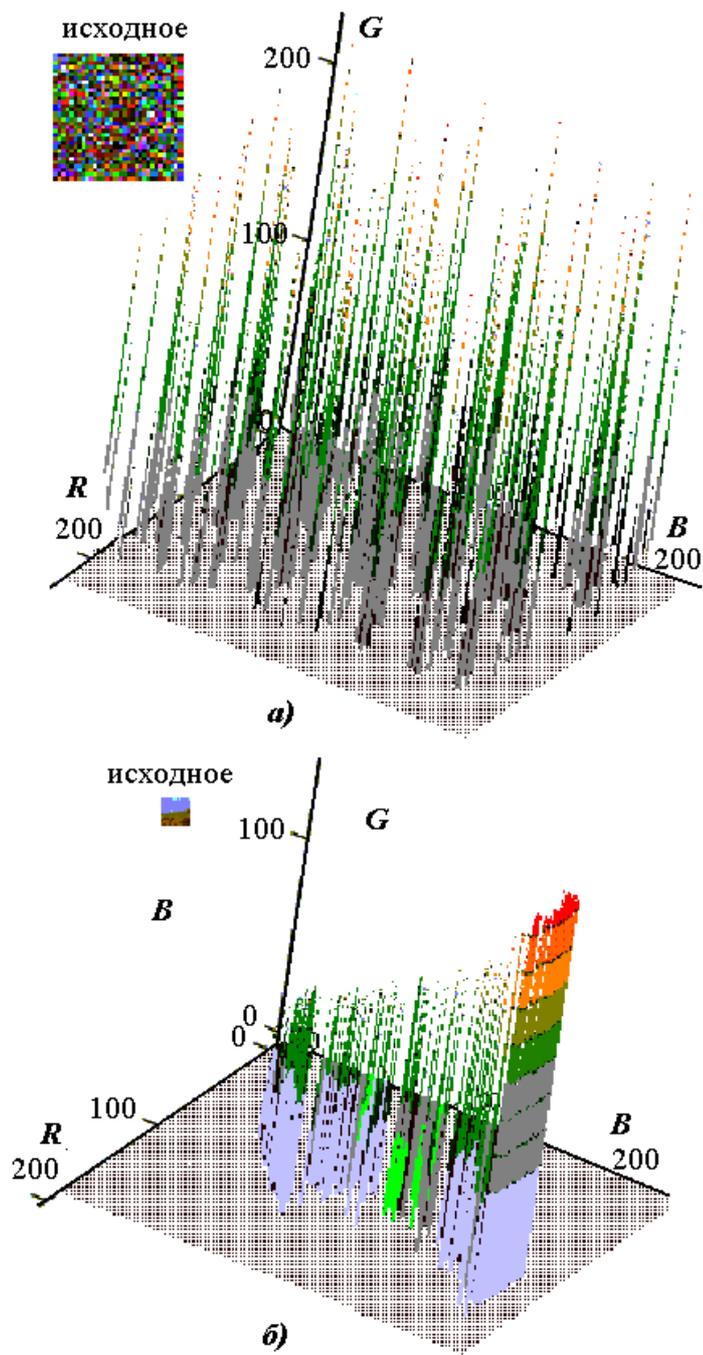


Рис. 3.20. Следы отображений пикселей на плоскости RB

В качестве решающих границ в трехмерном пространстве описания пикселей используются различные поверхности. Более просто, но с определенной потерей общности анализа, можно учитывать только следы точек на координатной плоскости (рис. 3.20). В этом случае решающие границы вырождаются в линии, которые разделяют следы точек на плоскости.

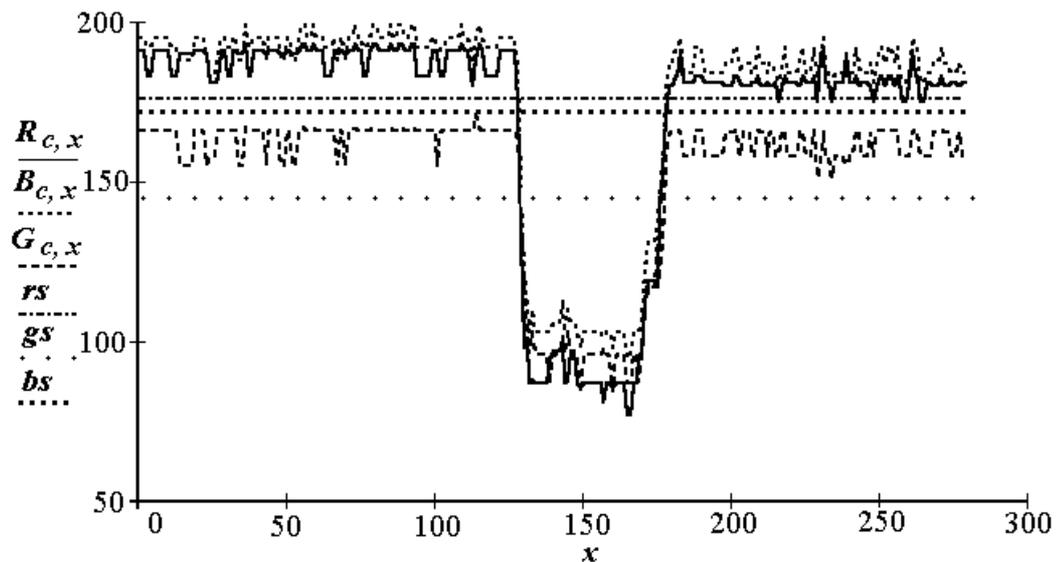


Рис. 3.21. Сечения решающих плоскостей и сигналов от цветových составляющих

Простейшие задачи распознавания решаются сечением пространства признаков плоскостями, которые разделяют зоны описаний классов. Например, через задание порогов по R , G , B компонентам (рис. 3.21 пороги rs , gs , bs). В этом случае говорят о гиперконусе принятия решений. Назначая нижние и верхние границы существования классов, получим прямоугольную область существования объектов класса. Однако многие ситуации значительно сложнее, и решающие функции приобретают вид сложных гиперповерхностей.

Сформируем прозрачную маску $mask1$ на распознаваемое здание (код 255 назначим прозрачным цветом, такое решение часто используется в графических редакторах)

$$mask1_{y,x} = \text{if}(R_{y,x} \leq rs, \text{if}(G_{y,x} \leq gs, \text{if}(B_{y,x} \leq bs, 255, 0), 0), 0).$$

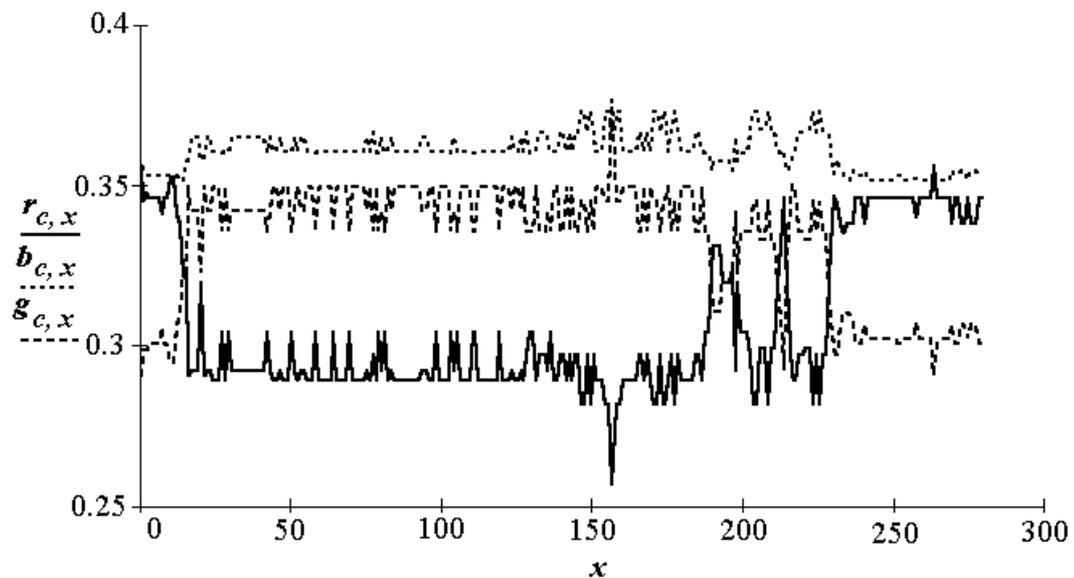


Рис. 3.22. Нормированные цветových сигналы

Автоматическое задание фиксированных порогов по цветovým составляющим предполагает наличие алгоритмов определения уровня освещенности объектов. Но освещенность отдельных фрагментов объекта обычно значительно изменяется. Выходом может стать переход к распознаванию объектов в нормированном цветovém пространстве.

На рис. 3.22 приведены нормированные сигналы r, g, b полученные из исходных сигналов R, G, B путем их деления на сумму $R+G+B$, вычисляемой для каждого пикселя.

Как и в первом случае сформируем прозрачную маску

$$mask2_{y,x} = \text{if}(g_{y,x} \geq r_{y,x}, 255, 0).$$

Результаты вычислений представлены на рис. 3.23. Как видно, отдельные элементы здания распознаны различными алгоритмами не однозначно. Бесспорное преимущество второго метода в его независимости от освещенности объекта. Первый метод косвенно использует информацию об освещенности пикселя и этот признак в данном случае способствует лучшему распознаванию объектов.

Приведем перечень операций, часто используемый при распознавании объектов в нормированном цветовом пространстве:

- вычислим сумму цветовых сигналов в точке;
- определим нормированные цветовые составляющие;
- назначим исключаемую из рассмотрения нормированную цветовую составляющую (обычно не учитывают цветовую составляющую в равной мере присущую всем объектам);
- понизим размерность цветового пространства на единицу;
- назначим новые координатные оси;

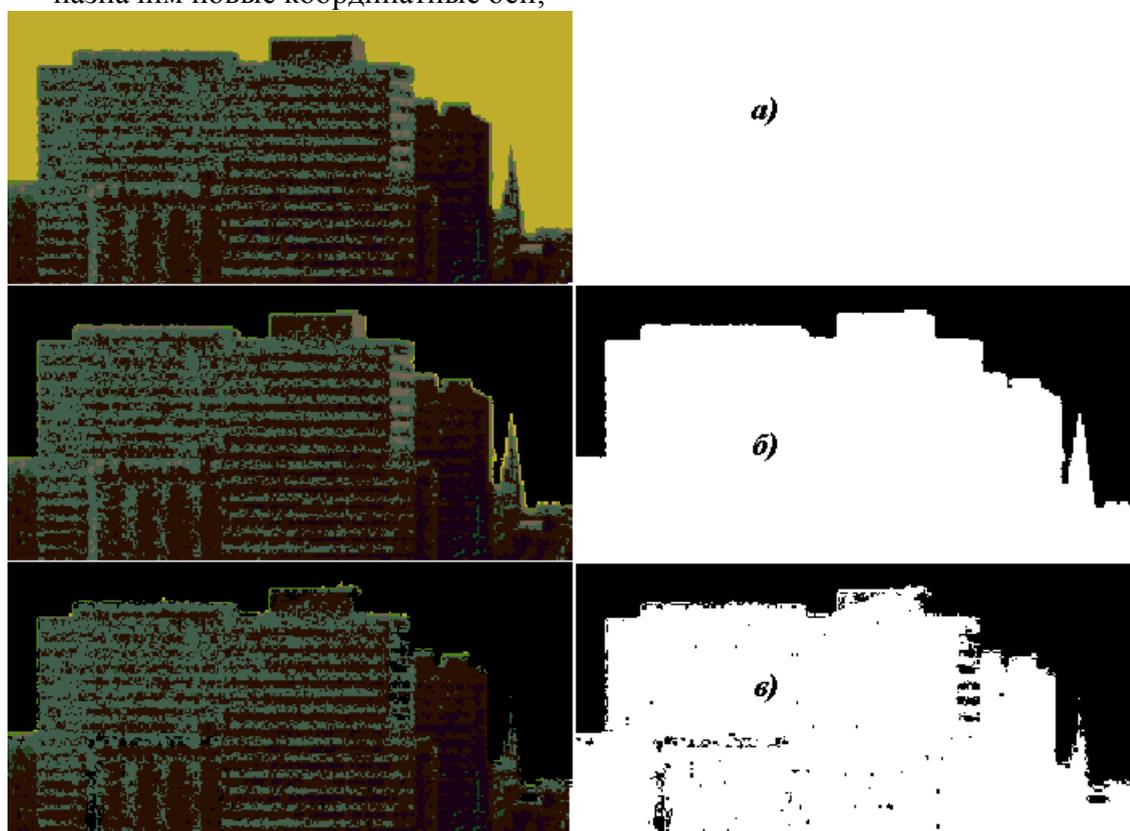


Рис. 3.23. Примеры применения алгоритмов распознавания объекта:

а) – объект - строение на фоне неба; б) – маска $mask1$, вычисленная через проведение решающих плоскостей по уровням сигналов от цветовых составляющих, и результат ее наложения; в) – маска $mask2$, определенная по нормированным цветовым сигналам, и результат ее наложения

- отобразим следы объектов или их гистограммы в новой системе координат;
- проведем решающие границы и классифицируем объекты.

На рис. 3.24 показаны результаты проведения выше перечисленных операций для исходного трехцветного пространства. Нормирование понижает размерность пространства

на единицу, в случае трехмерного пространства это означает переход к плоским отображениям, легко представляемым графически.

Так как нормированные цветовые сигналы используются в качестве индексов в новых массивах, описывающих цветное пространство, и могут принимать только целночисленные значения, умножим r , g на 100 и отбросим дробную часть числа:

$$r_{y,x} = \text{floor}(100 \cdot r_{y,x});$$

$$g_{y,x} = \text{floor}(100 \cdot g_{y,x}).$$

Следы ZS и гистограмму Zhi на плоскости $r0g$ получим по формулам:

$$ZS_{r_{y,x},g_{y,x}} = 1;$$

$$Zhi_{r_{y,x},g_{y,x}} = Zhi_{r_{y,x},g_{y,x}} + 1.$$

Решающая функция из поверхности трансформируется в линию на плоскости, в данном случае параллельную оси $0g$. Сформируем маску $mask3$ выделяющую гору.

$$mask3_{y,x} = \text{if}(r_{y,x} < 30, 0, 255).$$

На рис. 3.25 приведены маскированные распознанные объекты – часть горы и небо.

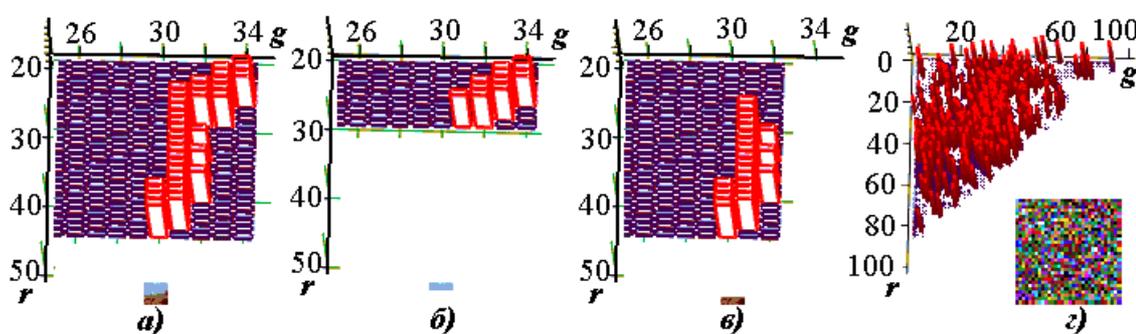


Рис. 3.24. Следы объектов на нормированной цветовой плоскости:
а) – гора и небо; б) – только небо; в) – только гора; г) – цветовой хаос

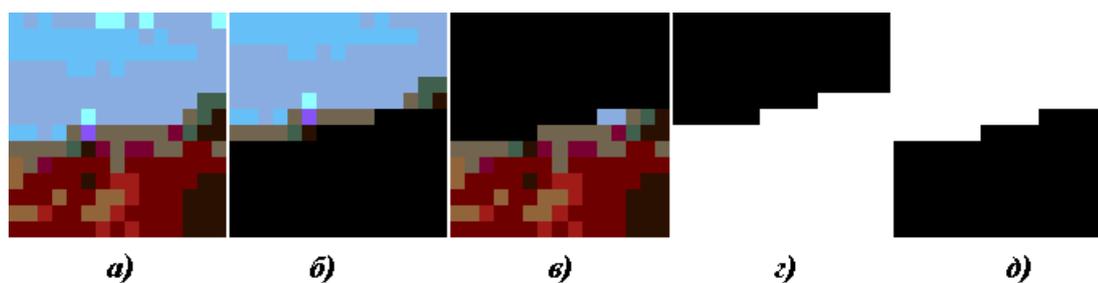


Рис. 3.25. Маски объектов в нормированной цветовой плоскости:
а) – гора и небо; б) – маскирована гора; в) – маскировано небо; г) – маска на небо; д) – маска на гору

Аналогичные рассуждения применимы и к виртуальным объектам – кодовым последовательностям, скрытой информации и т. п. Учет специфики данных объектов и назначения этих объектов и классов объектов, на фоне которых они проявляются, необходим.

Корреляционные алгоритмы распознавания образов

Корреляционные алгоритмы рассмотрим на примере распознавания символов печатного текста. Работы в области распознавания рукописных символов, плохо

пропечатанных документов, надписей на изделиях, восприятия смыслового содержания сообщений с целью корректировки результатов распознавания ведутся сегодня многими коллективами.

Последовательность процедур распознавания печатного текста в ряде алгоритмов включает бинаризацию исходного изображения, определение межстрочечного интервала, сегментацию изображения, формирование эталонных образов символов, формирование признаков и т. п. Все процедуры могут проводиться как с локализацией положения распознаваемого символа, так и при пошаговом движении по всему изображению.

Рекомендуемая последовательность работы над распознаваемым текстом:

- ввести изображение текста низкого качества и сформировать его фрагмент (фрагмент следует выбрать размером не более 100 000 пикселей, это обеспечит приемлемые временные затраты на расчеты);
- выбрать один или несколько различных символов, они представят распознаваемые классы;
- уточнить признаковое описание каждого класса;
- пронормировать эталонный образ;
- сформулировать бегущий сегмент рабочего образа;
- пронормировать данные рабочего сегмента;
- вычислить корреляционное соотношение между признаковым описанием рабочего и эталонного сегмента;
- задав рабочий уровень доверия dr , определить принадлежности рассматриваемого сегмента к одному из классов;
- проанализировать результаты процедур распознавания, определив проценты ошибок и наметить пути совершенствования использованных алгоритмов.

На рис 3.26 приведен выбранный с общего изображения фрагмент текста и выбранный класс символ – с, т. е. распознаются два класса символ – с и не символ – с. Каждый класс представлен 5×6 пикселями т. е. 30 признаками. При работе над эталонным образом можно использовать два варианта:

- эталонный образ формируется в полуавтоматическом режиме с ручной корректировкой описания символа;
- эталонный образ формируется автоматически, путем отбора описаний пикселей в обучающей последовательности, отбор может идти по мажоритарному алгоритму или через параметры гистограмм.

Учитывая временные ограничения, рекомендуется сформировать эталонный образ в полуавтоматическом режиме. Следует помнить, класс искомого символа только в упрощенном варианте представлен одним эталоном. В реальных задачах описание класса задается несколькими эталонными образами.

Нормировка эталонного образа устраняет влияние таких параметров, как средний уровень фона, освещенность символа и т. п. В простейшем случае нормировка выполняется через вычитание среднего и масштабирование по уровню сигнала, обеспечивающее заданное максимальное значение параметра доверия dm . Ниже приведены рекомендуемые формулы расчета bn - нормированного массива признаков:

$$bn_{y,x} = \left(\frac{b_{y,x}}{Nb} \cdot dm \right);$$

где $b_{y,x} = B_{y,x} - \text{mean}(B)$ новый промежуточный массив с устранением среднего, а

$$Nb = \frac{1}{256 \cdot 256} \cdot \sum_y \sum_x b_{y,x} \cdot b_{y,x}$$

значение автокорреляционной функции, вычисленной с

учетом диапазона задания описания пикселя.

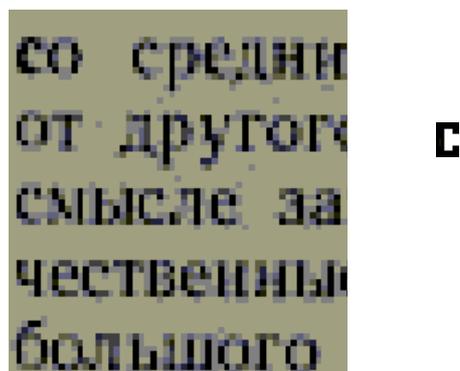


Рис. 3.26. Фрагмент распознаваемого текста и выбранный эталонный символ

При формировании бегущего сегмента рабочего образа шаг смещения по y , x следует выбирать равным единице, а размеры рабочего сегмента равными размерам эталонного. Операции при нормировке также рекомендуется проводить аналогично расчетам по эталонному образу.

$$K_{ky,kx} = \frac{1}{256 \cdot 256} \cdot \sum_y \sum_x b_{ny,x} \cdot r_{y+ky \cdot n, x+kx \cdot m};$$

где r рабочий движущийся сегмент.

Рабочий уровень доверия dr сложный параметр, задающий процент ошибок первого и второго рода, возникающих при распознавании. В практической работе рекомендуется задать этот параметр вручную, как процент от dm . Массив указателей Kd на обнаруженные символы можно вычислить по формуле:

$$Kd_{ky,kx} = \text{if}(K_{ky,kx} \geq dr, 255, 0).$$

На рис. 3.27 приведены результаты поиска символа c на изображении. Малая величина рабочего уровня доверия ($dr < 560$, $dm = 1000$) привела к тому, что ошибочно включены в перечень найденных символов две буквы e и буква o (рис. 5.2а, там же показана маска на текст для уровня dr равного 520). Правильно выбранный уровень доверия ($569 < dr < 598$) указал на все буквы c в тексте (рис. 3.27 б, там же изображение текста). При уровне $dr > 599$ программа не обнаруживает все искомые буквы (рис. 3.27 в).

Автоматическое определение оптимальной величины dr и уточнение описаний признаков эталонных образов, осуществляется в процессе обучения и функционирования системы по информации об ошибках.

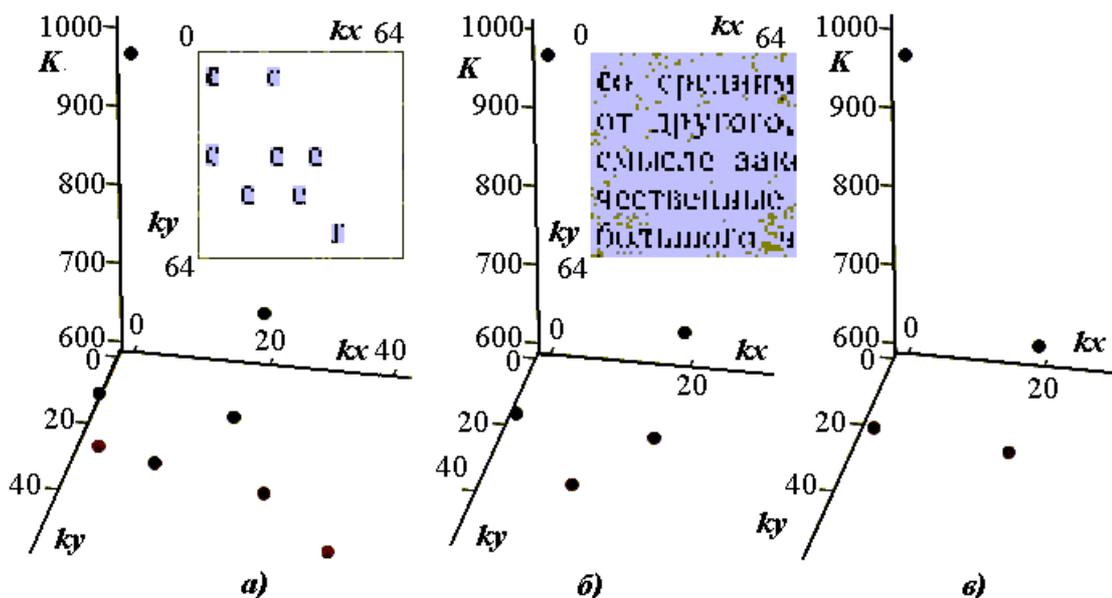


Рис. 3.27. Отобранные символы при различных значениях уровня доверия
 а) – $dr = 520$ б) – $dr = 570-598$ в) – $dr = 600$

Следующим этапом уточнения выводов распознавания является восприятие смыслового содержания сообщений, как это обычно делает человек, но эти задачи выходят за рамки данной работы.

Выбор окончательного решения

Сложившаяся на сегодня методика поддержки принятия решений в большинстве случаев рекомендует последовательное прохождение следующих этапов:

- анализ ситуации с формированием матрицы решений;
- выработку одного или нескольких критериев принятия решений (задание оценочных функций);
- определение номеров решений по выбранным критериям;
- анализ полезности выбранных вариантов решений.

Данные этапы, как правило, повторяются несколько раз с постепенным уменьшением числа возможных решений и перечня анализируемых ситуаций их применения. В системах искусственного интеллекта эти процедуры также программируются с различной степенью адаптации алгоритмов и их параметров к изменению ситуаций в процессе существования системы.

Все компоненты матрицы решений, целевые функции неизбежно имеют статистический характер, поэтому в процессе принятия решений многократно применяются методы анализа случайных процессов и событий.

Формирование матрицы решений

Учитывая опыт составления описания классов распознаваемых объектов, полученный в предыдущих работах, данный этап рекомендуется выполнить упрощенно. В примере, на базе которого поясняются этапы работы, элементы матрицы решения A генерируются, как сумма гармонической функции и функции Бесселя заданных в плоскости, координатами которой являются номер решения y и номер ситуации осуществления решения x .

$$A_{y,x} = 3 \cdot \sin(2x + 0.8y) + 6 \cdot \frac{J_0(0.2y + 0.1x - 2.3)}{(0.2 \cdot y + 1)} + 2.1.$$

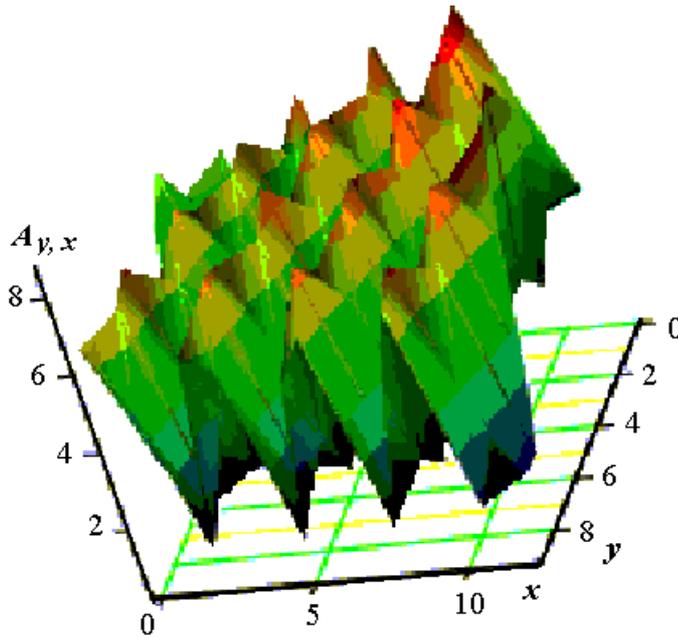


Рис. 3.28. Графическое представление матрицы решений

Дополним матрицу столбцами \min , \max и mean , рассчитанными по строкам. Сформируем вектора X - строки из массива A и определим требуемое содержимое добавляемых столбцов.

$$X_y = \text{submatrix}(A, y, y, 0, xm - 1),$$

где xm – число столбцов матрицы A .

$$\max_X_y = \max(X(y));$$

$$\min_X_y = \min(X(y));$$

$$\text{mean}_X_y = \text{mean}(X(y)).$$

Таблица 1а

Плата за решение при условии												
0	1	2	3	4	5	6	7	8	9	10	11	12
2.4	0.8	5.4	2.6	6.7	2.5	2.8	7.8	3.2	4.3	8.5	6.1	3.7
5.0	4.2	0.5	5.2	5.8	1.4	5.3	7.2	2.5	5.3	8.4	3.7	5.1
6.3	2.2	1.9	6.9	3.7	2.0	7.3	5.3	2.3	7.4	6.7	2.7	7.2
5.6	0.9	4.3	6.7	1.9	4.1	7.6	3.0	3.8	8.2	4.2	3.5	8.4
3.6	1.3	6.5	5.0	1.5	6.4	6.2	1.9	6.0	7.2	2.4	5.5	7.9
1.6	3.2	7.2	2.7	2.9	7.6	3.8	2.5	7.6	4.9	2.3	7.3	5.9
1.0	5.6	6.0	1.4	5.1	6.9	1.9	4.5	7.5	2.6	3.8	7.8	3.3
2.2	7.1	3.7	1.9	7.0	4.8	1.6	6.6	5.7	1.6	5.9	6.4	1.6
4.4	6.7	1.7	3.8	7.2	2.4	3.0	7.4	3.2	2.3	7.2	3.9	1.7
6.4	4.8	1.2	5.9	5.6	1.2	5.2	6.3	1.3	4.3	6.7	1.6	3.3

Нарастим исходную матрицу строками и столбцами \max , \min и mean :

$$A = \text{augment}(A, \min_X, \max_X, \text{mean}_X).$$

Матрица представлена в табл. 1а и табл. 1б. Столбцы 13, 14, 15 – минимальное, максимальное, среднее значение результата решения, номер которого определяется номером строки.

Таблица 1б

Дополнительные столбцы		
13	14	15
0.8	8.5	4.4
0.5	8.4	4.6
1.9	7.4	4.7
0.9	8.4	4.8
1.3	7.9	4.7
1.6	7.6	4.6
1.0	7.8	4.4
1.6	7.1	4.3
1.7	7.4	4.2
1.2	6.7	4.2

Оценочные функции

Ниже приведены некоторые формулировки оценочных функций, как правило, используемых в процедурах принятия решений. Фактически они сводятся к поиску максимума в ранее сформированном столбце табл. 1б. Следующие столбцы табл. 2, 3, 4 содержат номера решений и соответствующий результат решения, на который ориентируется выбранный критерий. Жирным выделены числа, определенные по формулам данной таблицы.

Таблица 2

Оценочная функция пессимистическая $\max \min$ (минимаксный критерий):	Номер решения	Результат решения
$Ar = \text{submatrix}(A, 0, ym - 1, xm, xm)$, где ym – число возможных решений. Промежуточный массив для поиска \max $Arv_y = \text{if}(Ar_y = \max(Ar), y, 0)$. Номер решения, принятого по минимаксному критерию $Yq = \max(Arv)$, $Yq = 2$.	0	0.8
	1	0.5
	2	1.9
	3	0.9
	4	1.3
	5	1.6
	6	1.0
	7	1.6
	8	1.7
	9	1.2

Таблица 3

Оценочная функция предельно оптимистическая $\max \max$ (критерий азартного игрока):	Номер решения	Результат решения
$Ar = \text{submatrix}(A, 0, ym - 1, xm + 1, xm + 1)$. Промежуточный массив для поиска \max - $Arv_y = \text{if}(Ar_y = \max(Ar), y, 0)$. Номер решения, принятого по минимаксному критерию $Yq1 = \max(Arv)$, $Yq1 = 0$.	0	8.5
	1	8.4
	2	7.4
	3	8.4
	4	7.9
	5	7.6
	6	7.8
	7	7.1
	8	7.4
	9	6.7

Таблица 4

Оценочная функция нейтралитета max mean (критерий нейтрального игрока):	Номер решения	Результат решения
$Ar = \text{submatrix}(A, 0, ym - 1, xm + 2, xm + 2)$.	0	4.4
	1	4.6
Промежуточный массив для поиска max -	2	4.7
$Arv_y = \text{if}(Ar_y = \max(Ar), y, 0)$.	3	4.8
	4	4.7
	5	4.6
Номер решения, принятого по минимаксному критерию $Yq2 = \max(Arv)$,	6	4.4
$Yq2 = 3$.	7	4.3
	8	4.2
	9	4.2

Преобразование исходной матрицы решений

Целый ряд критериев требует преобразования исходной матрицы, ее дополнения вероятностным описанием компонентов. Преобразуем матрицу решений с учетом вероятности появления условия x .

Сгенерируем закон распределения, близкий к ожидаемому ps , скорректируем psl , пронормируем p :

В качестве ожидаемого распределения вероятности проявления условия осуществления решения принято ХИ-квадрат распределение с пятью степенями свободы. Корректировка заключается в повышении вероятности появления условий с малыми номерами.

$$d_x = x;$$

$$ps = \text{dchisp}(d, 5);$$

$$psl_x = ps_x + \frac{0.2}{x + 2};$$

$$p_x = \frac{psl_x}{\sum_x psl_x}.$$

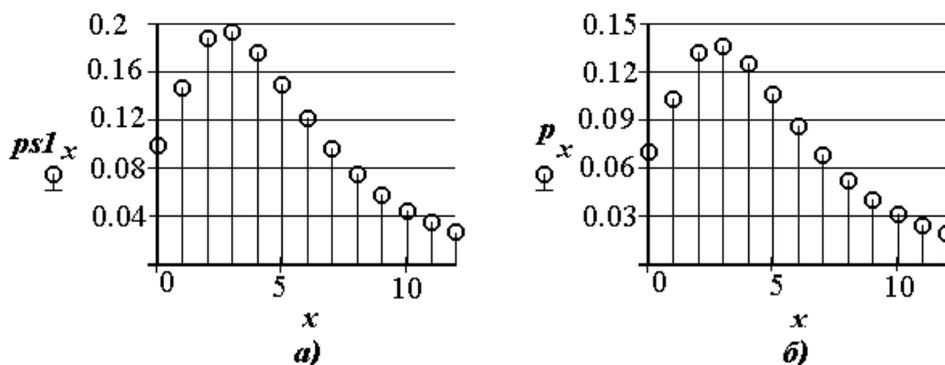


Рис. 3.29. Функция распределения вероятности появления события номер x

Новая матрица индексных оценок принятия решений AI приведена в табл. 5а.

Она дополнена столбцами наименьших, наибольших и средних (табл. 5б.). Оценки рассчитаны по формуле $AI_{y,x} = 10 \cdot A_{y,x} \cdot p_x$. Коэффициент 10 введен искусственно. Он не влияет на процедуры принятия решений, но приближает цифры к оценкам последствий

решений. В качестве этого коэффициента целесообразно брать число рассматриваемых условий реализации решений.

Таблица 5а

Плата за решение при условии номер												
0	1	2	3	4	5	6	7	8	9	10	11	12
1.7	5.7	1.1	3.5	8.4	2.6	2.4	5.3	2.2	1.3	2.7	1.5	0.7
3.5	4.3	0.6	7.2	7.3	1.5	4.6	4.9	1.3	2.1	2.6	0.9	1.0
4.4	2.3	2.5	9.5	4.7	2.1	6.3	3.6	1.2	3.0	2.1	0.6	1.4
3.9	0.9	5.8	9.2	2.4	4.3	6.6	2.0	2.0	3.3	1.3	0.8	1.6
2.5	1.3	8.7	6.8	1.9	6.8	5.3	1.3	3.2	2.9	0.7	1.3	1.5
1.1	3.4	9.6	3.7	3.6	8.0	3.3	1.7	4.0	2.0	0.7	1.8	1.1
0.7	5.8	8.0	2.0	6.4	7.3	1.7	3.1	4.0	1.0	1.2	1.9	0.6
1.5	7.3	5.0	2.6	8.7	5.1	1.4	4.5	3.0	0.6	1.8	1.6	0.3
3.1	7.0	2.3	5.2	9.0	2.6	2.6	5.0	1.7	0.9	2.2	0.9	0.3
4.5	5.0	1.6	8.2	7.1	1.3	4.5	4.3	0.7	1.7	2.1	0.4	0.6

Таблица 5б

13	14	15
0.7	8.4	3.0
0.6	7.3	3.2
0.6	9.5	3.3
0.8	9.2	3.4
0.7	8.6	3.5
0.7	9.6	3.4
0.6	8.0	3.4
0.3	8.7	3.3
0.3	9.0	3.3
0.4	8.2	3.2

В табл. 6 приведены результаты поиска оптимального решения по критерию Бейеса–Лапласа.

Таблица 6

Оценочная функция нейтральная, с вероятностью появления условия (критерий Бейеса–Лапласа):	Номер решения	Индекс решения
$Ar = \text{submatrix}(A1, 0, ym - 1, xm + 2, xm + 2).$ Промежуточный массив для поиска \max $Arv_y = \text{if}(Ar_y = \max(Ar), y, 0).$ Номер решения, принятого по минимаксному критерию $Yq3 = \max(Arv),$ $Yq3 = 4.$	0	3.0
	1	3.2
	2	3.3
	3	3.4
	4	3.5
	5	3.4
	6	3.4
	7	3.3
	8	3.3
9	3.2	

Графическое отображение поля выбора решения

Переработаем матрицу решения оставив два условия осуществления решений. Одно из них определим, как усредненное из последствий условий с номерами 0-4, а условия с номерами 5-10 и 12 опустим. Данная процедура используется практически в каждом этапе

итерации изложенной выше методики принятия решений. Мы проводим ее с предельным упрощением. Это обусловлено возможностями графического отображения функций на листе бумаги. Кроме того, сведение описаний ситуаций к двум вариантам в практике принятия решений человеком встречается довольно часто.

Сформируем два столбца A_2 , A_3 . Первый, – как новое условие осуществления решения, объединяющее условия с номерами 0-5. Во второй перенесем условие с номером 11. Все компоненты умножим на 10, это необходимо для их использования в качестве адресов нового массива Z . Остальные опустим, как неучтенные. В полученных координатах поля выбора решений построим массив номеров решений, поднятый для наглядности на 10.

$$A_{2y} = \text{floor} \left[\left(\frac{1}{5} \cdot \sum_{i=0}^4 A_{y,i} \right) \cdot 10 \right];$$

$$A_{3y} = \text{floor}(A_{y,11} \cdot 10);$$

$$Z_{A_{2y}, A_{3y}} = y + 10.$$

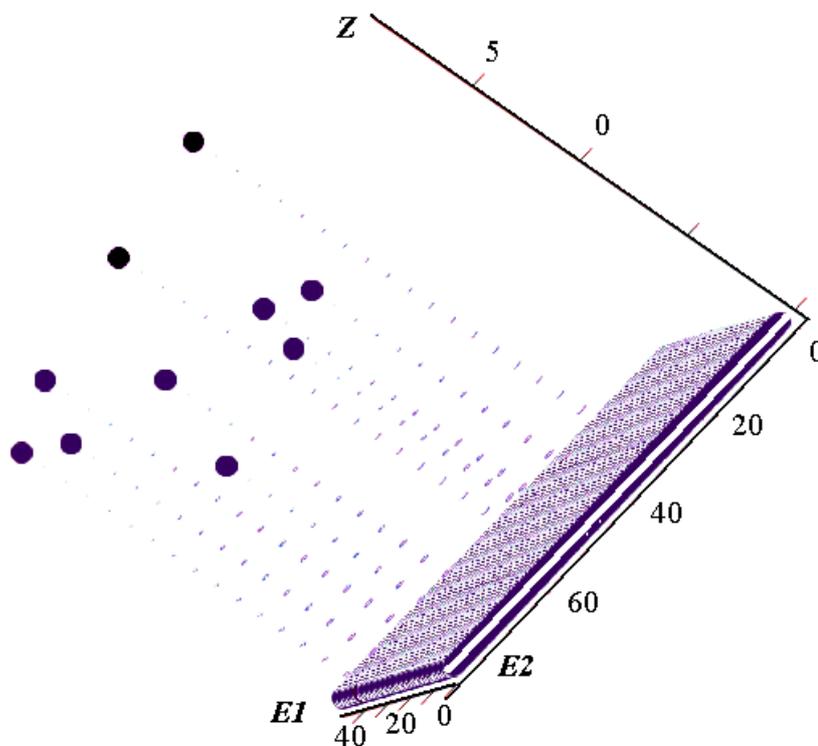


Рис. 3.30. Решения на поле выбора решений

Потеряв номер решения (его можно в дальнейшем восстановить) изобразим следы решений на плоскости. Введем линии, ограничивающие поле выбора решений (вертикальные линии заданы как границы оси ординат).

Помещая рабочую точку в следы решений можно делать выбор, ориентируясь на геометрическую трактовку оценочных функций критериев принятия решений.

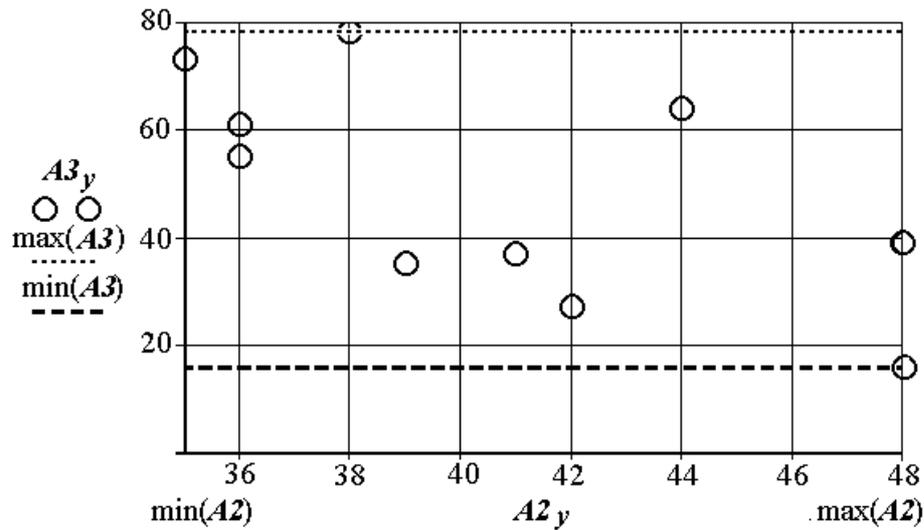


Рис. 3.31. Поле выбора решения

Сформируем новую матрицу, введя столбцы A2, A3 и дополним ее min, max и mean по строке. В табл. 7 первый столбец – номер решения.

Таблица 7

	A2	A3	min	max	mean
0	36	61	36	61	48.5
1	41	37	37	41	39
2	42	27	27	42	34.5
3	39	35	35	39	37
4	36	55	36	55	45.5
5	35	73	35	73	54
6	38	78	38	78	58
7	44	64	44	64	54
8	48	39	39	48	43.5
9	48	16	16	48	32

Выберем критерий нейтрального игрока и найдем для сравнения номер решения по вышеописанной методике. Это решение номер 6.

Далее найдем решение по данному критерию, используя геометрические аналогии. Уравнение, определяющее вид линии предпочтения, в данном случае задает планируемый выигрыш, как величину B , пропорциональную сумме выигрышей по координатам $B=S+i$. Направляющая (N), вдоль которой движется линия предпочтения, является биссектрисой угла, образованного осями координат. Конус предпочтения движется в поле принятия решения до тех пор, пока в нем останется только одна точка.

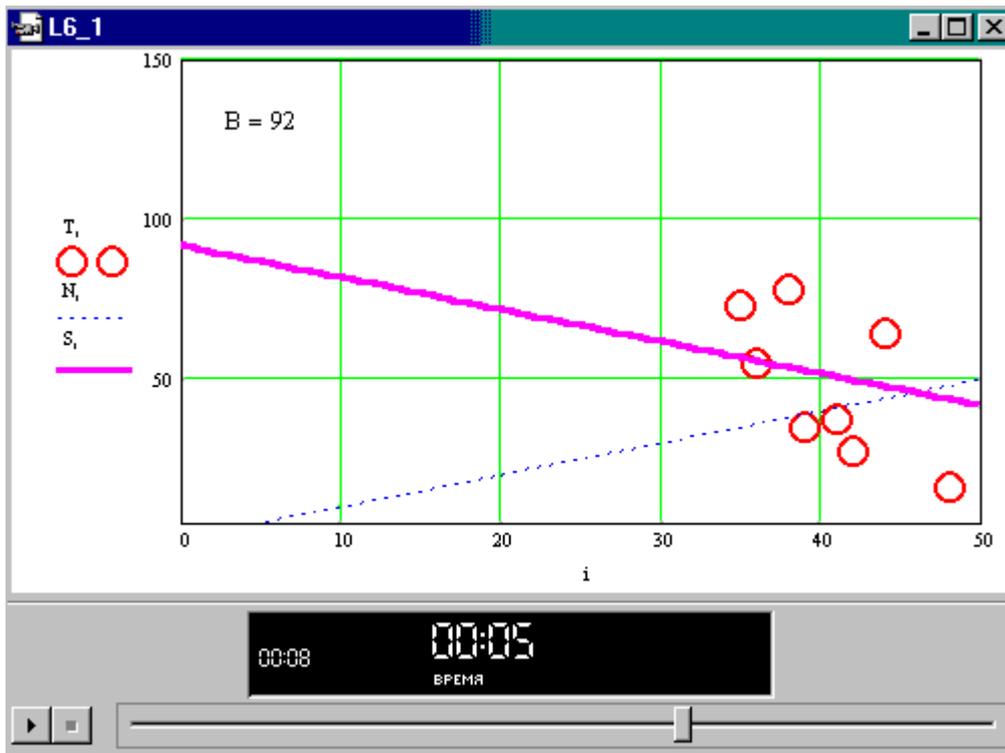


Рис. 3.32. Конус предпочтения содержит три точки, несущие больший выигрыш чем рабочая

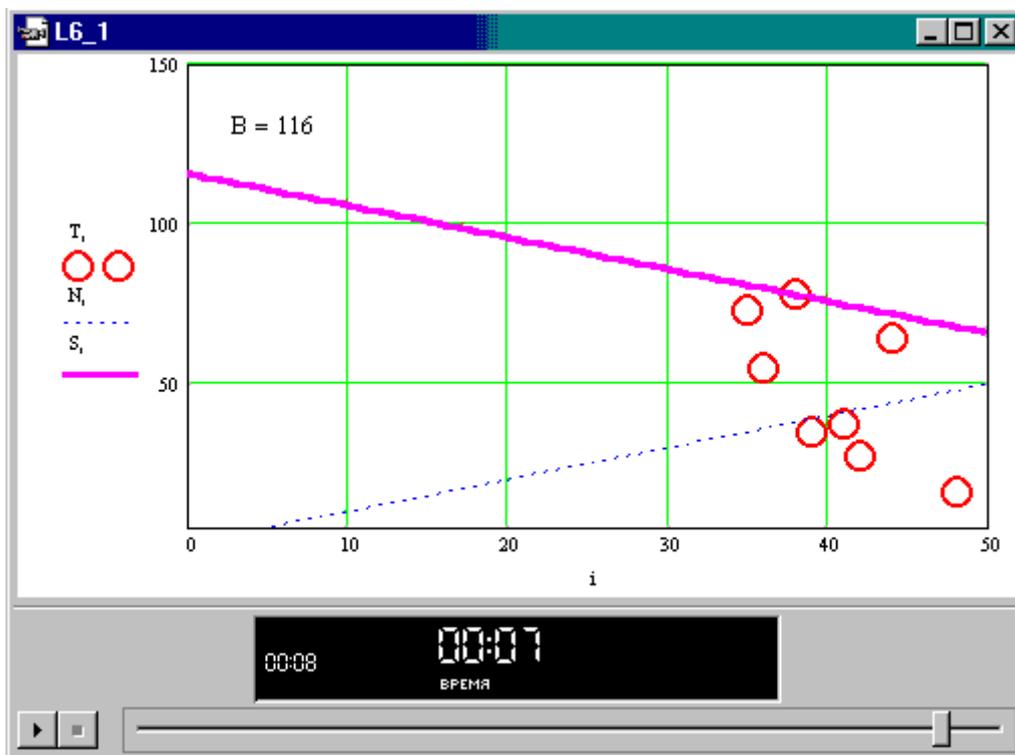


Рис. 3.33. Все точки, кроме рабочей, вне конуса предпочтения

Эта точка соответствует оптимальному решению. На рис. 3.32 и 3.33 показаны кадры видеофильма, в котором смещается конус предпочтения.

Выбрано также решение номер 6 (78, 38).

Задание к практической работе

Сложность выполнения работы в ее объеме. Поэтому определите с преподавателем те задачи, которые Вы будете решать. Наилучшим вариантом выполнения работы является согласование задания с темой Вашей предполагаемой дипломной работы.

1. Выберите объект и задачу распознавания, согласуйте ее с преподавателем.
2. Назначьте классы, составьте алфавит признаков.
3. Оцените параметры согласно алфавита признаков (например: размеры, цвет и др.) подозрительных объектов.
4. Сформулируйте правила выбора решения.
5. Понизить размерность пространства, сформулировать решающие функции и оценить их параметры.
6. Проведите корреляционное распознавание объектов (например: при ориентации на печатные символы, выполните расчеты с 3-4 символами).
7. Оценить эффективность алгоритма по близости ошибочных и верных ответов.
8. Составить матрицу решений.
9. Выбрать один из критериев принятия решений, определить номер решения и полезность его применения.
10. Получить графическое изображение поля выбора решений с номерами решений, определить вид функций предпочтения и направляющую (если она несложно находится) для одного из используемых Вами критериев.

Содержание отчета

1. Характеристики объектов и описание задачи распознавания.
2. Описание классов и назначенных признаков.
3. Полученные числовые значения выбранных компонентов алфавита признаков.
4. Выбранные эталоны классов, их расстояние в пространстве признаков.
5. Описание правил принятия решения.
6. Результаты распознавания при различных порогах решающего правила.
7. Матрица решений и полученные номера решений с оценкой полезности.
8. Графическое изображение поля выбора решений.
9. Выводы по эффективности исследованных решений.

ВОПРОСЫ К ПРАКТИЧЕСКИМ РАБОТАМ

1. Роль теории распознавания образов в формировании математического базиса интеллектуальных систем.
2. Место процедур распознавания образов в задачах искусственного интеллекта.
3. Оптимальный выбор параметра по оценкам двух экспертов.
4. Оптимальная обработка сигналов в линейных системах и её место в базисе теории распознавания образов.
5. Случайные события, процессы, потоки, смеси и технология их анализа.
6. Параметры случайных величин, их оценка.
7. Вероятностные модели компонентов пространства признаков, правила проверки гипотез, примеры критериев согласия.
8. Дискретизация и квантование пространства признаков, снижение объема информационного описания.
9. Риск и его описание в распознавании образов.
10. Оценки вероятности ложной тревоги и пропуска цели при распознавании одного из двух объектов.
11. Составление алфавита признаков: этапы, цели и правила.
12. Математические модели объектов в пространстве признаков.
13. Выбор геометрических размеров пикселей при распознавании объектов по их изображениям.
14. Специфика формирования изображений в акустике, магнитной дефектоскопии.
15. Специфика формирования изображений при использовании заряженных частиц и гамма-излучения.
16. Элементы теории полезности в экономических решениях.
17. Бейсовское решающее правило распознавания образов.
18. Классификация систем распознавания образов.
19. Распознавание объектов по цветовому описанию.
20. Распознавание объектов по геометрическим параметрам.
21. Корреляционные алгоритмы распознавания объектов.
22. Возможности человека в распознавании образов.
23. Решающие правила в распознавании образов, конусы предпочтения.
24. Формальная структура принятия решения.
25. Оценочная функция, преобразование матрицы решений.
26. Графическая интерпретация правил выбора, поле полезности решений, утопическая и антиутопические точки, конусы, опорная линия.
27. Минимаксный критерий принятия решения.
28. Критерий принятия решения Бейса-Лапласа.
29. Критерий принятия решения Сэвиджа.
30. Сравнение классических критериев принятия решений.
31. Основные направления формирования производных критериев принятия решений.
32. Критерий принятия решения Гурвица.
33. Критерий принятия решения Ходжа-Лемана.
34. Критерий принятия решения Гермейера.
35. VL(ММ) критерий принятия решения.
36. Принятие решения по критерию произведений.
37. Расширенный минимаксный критерий принятия решения.
38. Критерий принятия решений азартного игрока.
39. Методика принятия решений по нескольким критериям.
40. Гибкий критерий принятия решения Мушака-Мюллера.
41. Адаптивный критерий принятия решения Кофлера-Менга.

42. Процедура последовательного принятия решения.
43. Обобщенная структура процессов распознавания образов и принятия решения.
44. Принятие решений в условиях маскировки.
45. Принятие решений в условиях противодействия.
46. Многокритериальные задачи в принятии решений.
47. Парето-оптимальные решения в экономике.
48. Оптимизация процессов распознавания образов и принятия решений.

Методические рекомендации по выполнению практических работ

1. Выполнение практикума предполагает хорошее владение методикой обработки изображений, матриц в математических пакетах и умение программировать.

2. До выполнения практической работы по конспекту лекций и рекомендуемым литературным источникам проработайте теоретические вопросы, необходимые для проведения работы.

3. На первой работе подготовьте личную рабочую папку в общей папке практикума. Помните, все ваши документы должны находиться только в вашей личной папке. В конце работы архивируйте ваши документы.

4. В общих материалах Вам предлагаются примеры выполнения работ. Можете использовать их как электронные книги, но при этом копируйте документы, которые Вы модифицируете, в свою папку. По возможности, согласовав предмет исследований с преподавателем, стремитесь наработать материал для вашей курсовой работы, научных исследований.

5. Каждую работу начинайте с четкой формулировки целей исследований.

6. Определитесь с объектами исследований.

7. Обязательно в начале каждой работы создайте документ будущего отчета (рекомендуется редактор Word 97 и выше). Все материалы работы, включая описание цели и объектов, вносите в отчет поэтапно, используя механизмы вставки объектов. Помните, только отчет является достаточным итоговым документом, подтверждающим выполнение Вами работы. В ходе работы систематически сохраняйте все рабочие документы, это предостережет Вас от потери результатов исследований.

8. Не стремитесь использовать исходные файлы объектов размером более 100 Кбайт, это ускорит выполнение работы без практического ухудшения качества и наглядности.

Примечания

1. Некоторые темы занятий по усмотрению преподавателя могут быть заменены на семинарские занятия, на которых заслушиваются и обсуждаются доклады по темам индивидуальных заданий и рефератов, выдаваемым согласно рейтинговой раскладке по лекционному курсу.

2. При самоподготовке следует пользоваться конспектами лекций, и рекомендованной преподавателем литературой.

Приложение А

Рейтинговая раскладка по практическим занятиям по дисциплине «Методы искусственного интеллекта в мехатронике и робототехнике»

Семестр 9

№ п/п	Тема	Количество баллов
1	Практической работе №1, отчет по практической работе №1	25
2	Практической работе №2, отчет по практической работе №2	25
3	Практической работе №3, отчет по практической работе №3	25
7	Всего баллов	75