

Тема 9. Базовые алгоритмы обработки изображений

План занятия

1. Общие сведения
2. Форматы хранения изображения в СТЗ
 - 2.1. Структура графического файла
 - 2.2. Сжатие изображения
3. Типовые алгоритмы обработки изображения
 - 3.1. Предварительная обработка
 - 3.2. Сегментация
 - 3.3. Кодирование
 - 3.4. Описание
4. Распознавание
5. Способы получения 3D изображения

1. Общие сведения

Сущность обработки изображения заключается в приведении исходного изображения сцены к виду, позволяющему решить задачу распознавания ее объектов.

Процесс преобразования информации в СТЗ можно представить в виде шести основных этапов.

Этапы обработки изображения



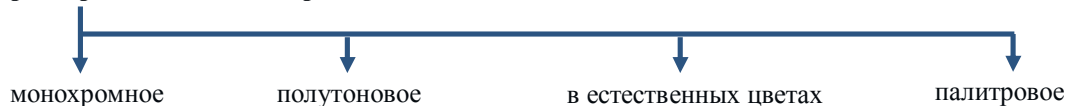
Этапы **распознавания** (идентификации), т. е. установление принадлежности объекта к некоторому **классу**, а также **интерпретация**, т. е. формализация нового класса объектов – относятся к задачам верхнего уровня СТЗ.

2. Форматы хранения изображения в СТЗ

Изображения сохраняют в максимально компактной и стандартной форме — в виде графического **файла**. Различают два типа графических файлов:

- векторные;
- растровые.

Форма представления изображения

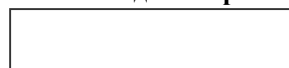


Векторное изображение представляет собой совокупность независимых математических объектов — **контуров**, каждый из которых можно перемещать и масштабировать. Примерами примитивных объектов являются: ломаные линии, многоугольники, окружности и эллипсы, кривые Безье, текст.

Кривая Безье представляет собой семейство кривых (первой, второй и третьей степени). Например, для случая кривой третьего порядка, справедлива зависимость



Растровое изображение представляет собой совокупность отдельных пикселей, записанную в ячейки памяти в виде таблицы или битовой карты. **Физический** размер пикселя связан с **разрешением** по полю устройства ввода изображения. Разрешение по полю определяют числом пикселей на дюйм **dpi**. Эта единица измерения является альтернативой числу телевизионных линий (**твл**), причем



Существуют два способа преобразования файлов из растрового формата в векторный:

- преобразование растрового файла в растровый объект векторного изображения;
- трассировка растрового изображения для создания векторного объекта.

Графический файл состоит из заголовка и собственно данных. **Заголовок** содержит спецификацию и общие сведения о файле. Данные разделяют на поля, теги и потоки.

- **Поле** называется структура данных, имеющая фиксированный размер и позицию в файле.
- **Тег** представляет собой структуру, размер и позиция которой изменяются от файла к файлу.
- В **потоке** организовано последовательное чтение блока данных различной длины.

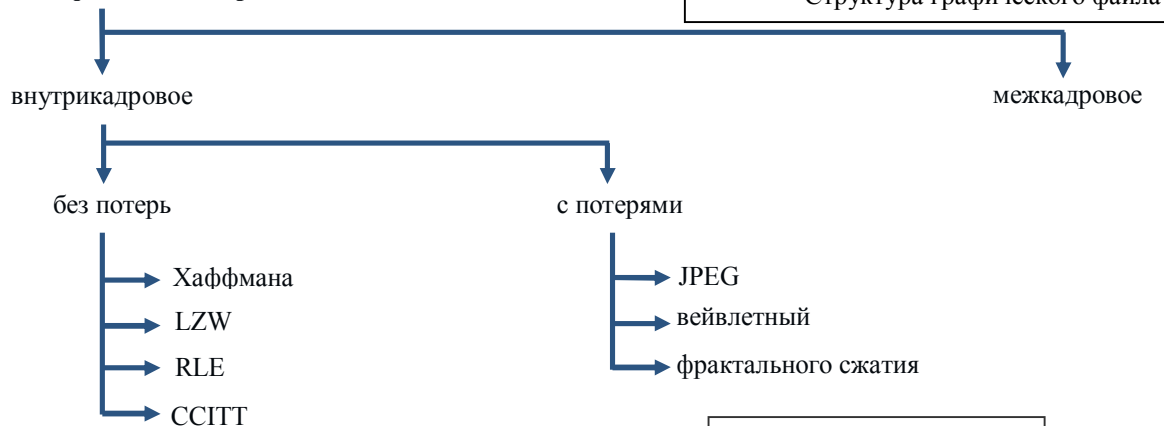
Простейшим способом организации пиксельных данных в растровом файле является использование **строк развертки**.

Сжатие (кодирование) изображения осуществляется аппаратно-программными устройствами — **кодеками**. Процедуры сжатия могут входить в спецификацию графического формата или выполняться отдельно.

Методы сжатия изображений подразделяют на:

- симметричные
- асимметричные.

Классификация алгоритмов сжатия



Эффективность сжатия оценивают **коэффициентом сжатия**:

Алгоритмы сжатия

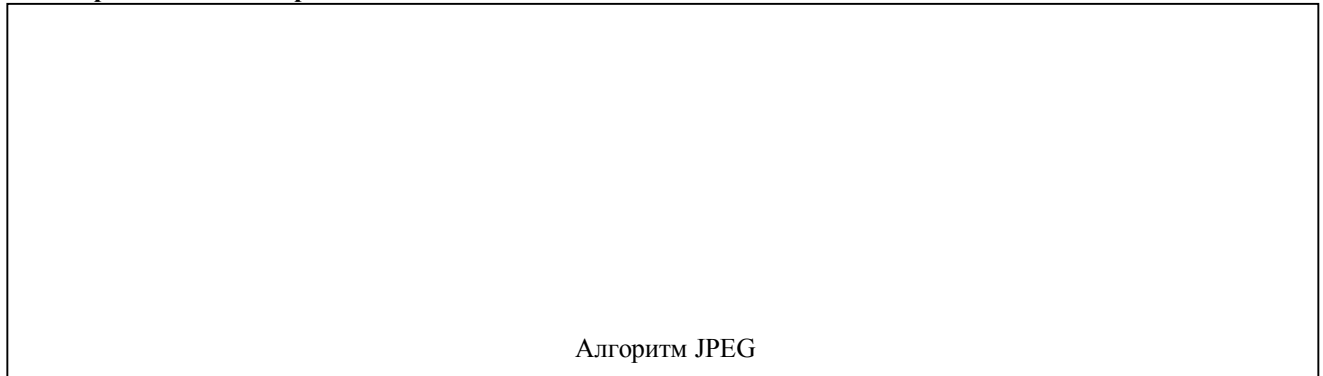
Различают две группы алгоритмов сжатия:

- сжатие без потерь;
- сжатие с потерями.

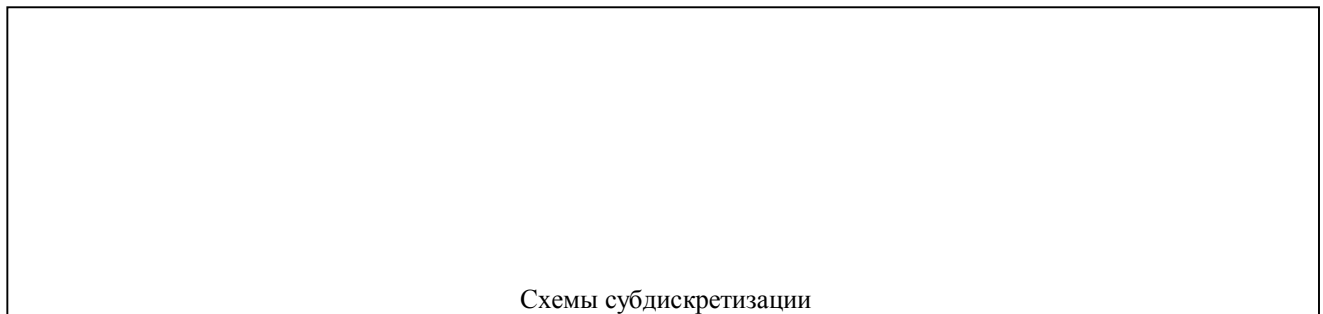
В алгоритмах сжатия **без потерь** либо оптимизируется кодирование минимального элемента информации — байта (алгоритмы Хаффмана и ССИТТ), либо удаляется избыточная информация (алгоритмы LZW и RLE).

К алгоритмам сжатия **с потерями** относят алгоритмы JPEG, вейвлет-преобразования и фрактального кодирования.

Этапы реализации алгоритма JPEG



Сущность **субдискретизации** заключается в том, что каждому блоку из 4-х пикселей (2×2) яркостного канала Y ставятся в соответствие усреднённые значения U (Cb) и V (Cr)



Дискретное косинусное преобразование (ДКП, DCT) — применяется в алгоритмах сжатия информации с потерями MPEG и JPEG. Это преобразование тесно связано с дискретным преобразованием **Фурье**.

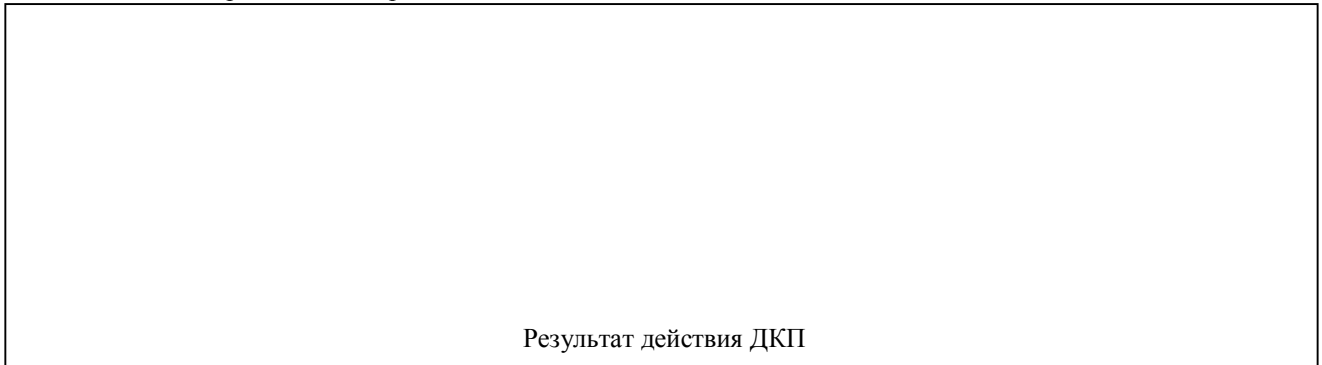
Действие ДКП, а также обратного ему преобразования можно описать с помощью матрицы преобразования **D**. Для прямого и обратного ДКП справедливы выражения:

$$\boxed{\phantom{Y_{D_{ij}} = C_{ij} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} Y_{xy} \cos \frac{(2y+1)i\pi}{2N} \cos \frac{(2x+1)j\pi}{2N}}} \quad \boxed{\phantom{Y_{xy} = C_{ij} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} Y_{D_{ij}} \cos \frac{(2y+1)i\pi}{2N} \cos \frac{(2x+1)j\pi}{2N}}}$$

При обработке изображений используют **двумерное ДКП**. При этом прямое и обратное ДКП производятся по следующим формулам:

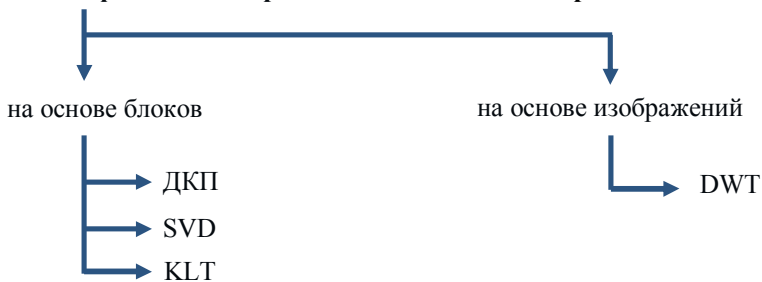
$$Y_{D_{ij}} = C_{ij} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} Y_{xy} \cos \frac{(2y+1)i\pi}{2N} \cos \frac{(2x+1)j\pi}{2N} \quad Y_{xy} = C_{ij} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} Y_{D_{ij}} \cos \frac{(2y+1)i\pi}{2N} \cos \frac{(2x+1)j\pi}{2N}$$

После ДКП получается **матрица**, в которой коэффициенты в левом верхнем углу соответствуют **низкочастотной** составляющей изображения, а в правом нижнем – **высокочастотной**



Большинство методов кодирования динамического изображения – **видеоизображения** основано на устранении временной и пространственной избыточности данных. В первом случае, имеет место значительная корреляция между ближайшими кадрами видеоизображения, особенно при большой частоте кадров. Во втором, наблюдается сходство значений соседних пикселей в кадре.

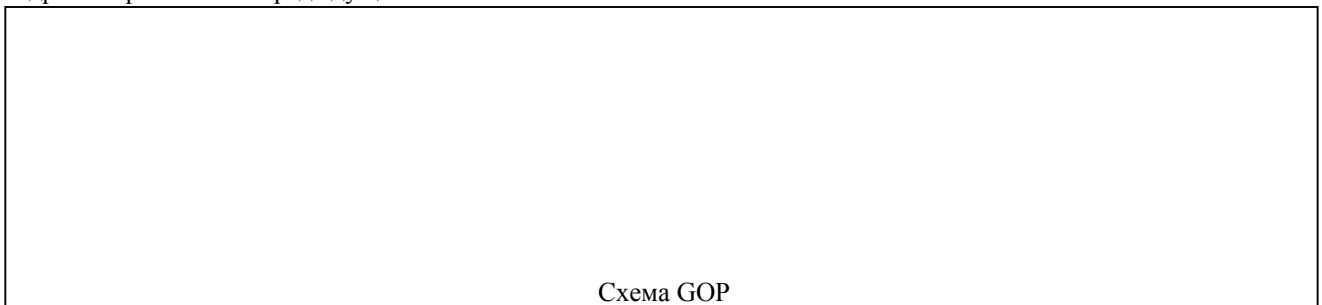
Классификация алгоритмов сжатия видеоизображений



Примерами преобразований на **основе блоков** являются **ДКП**, разложение по **сингулярным** числам (**SVD**) и преобразование Кархунена – Лозэвэ (**KLT**). Каждое из этих преобразований работает с $N \times N$ блоками пикселей (сэмплами) исходного или текущего изображения.

Преобразования на **основе изображений** работают с целым кадром или с его большим фрагментом. Наиболее популярным преобразованием этого типа является дискретное **вейвлетное** преобразование (ДВП, DWT).

Межфреймовое или дельта-сжатие, представляет собой группу алгоритмов, в которых кодируется отличие одного кадра изображения от предыдущего.



3. Типовые алгоритмы обработки

Выделяют три группы типовых алгоритмов обработки изображения, характерных для всех систем зрения:

1. предварительная обработка изображения;
2. сегментация;
3. описание.

Целью **предварительной обработки** изображения является формирование и последующее улучшение изображения, его бинаризация и кодирование (в частности, получение контурного представления).

Назначение и реализация



Формированием изображения называется процедура непосредственного получения изображения в виде расположенного в памяти видеопроцессора массива дискретных элементов — пикселей, образующих матрицу или контур

Гистограммой изображения называется график распределения полутонов изображения, в котором по горизонтальной оси представлена яркость, а по вертикали — относительное число пикселей с данным значением яркости.



Гистограмма позволяет оценить количество и разнообразие **оттенков** изображения, а также общий уровень **яркости** изображения.

Основные операции с гистограммами:

- выравнивание;
- эквализация;
- линейное контрастирование;
- степенное преобразование;
- гамма-коррекция;
- бинаризация.



Контрастирование относится к методам гистограммного выравнивания и используется при недостаточном диапазоне воспроизводимых яркостей. Линейное контрастирование реализуют линейным поэлементным преобразованием вида



К **степенным** преобразованиям относят преобразование вида:



Бинаризацию изображения проводят непосредственно после улучшения изображения также с помощью гистограммы полутонов. Выходное **бинаризованное** изображение $Y_b(x, y)$ содержит лишь две градации яркости: 0 или 1, причем



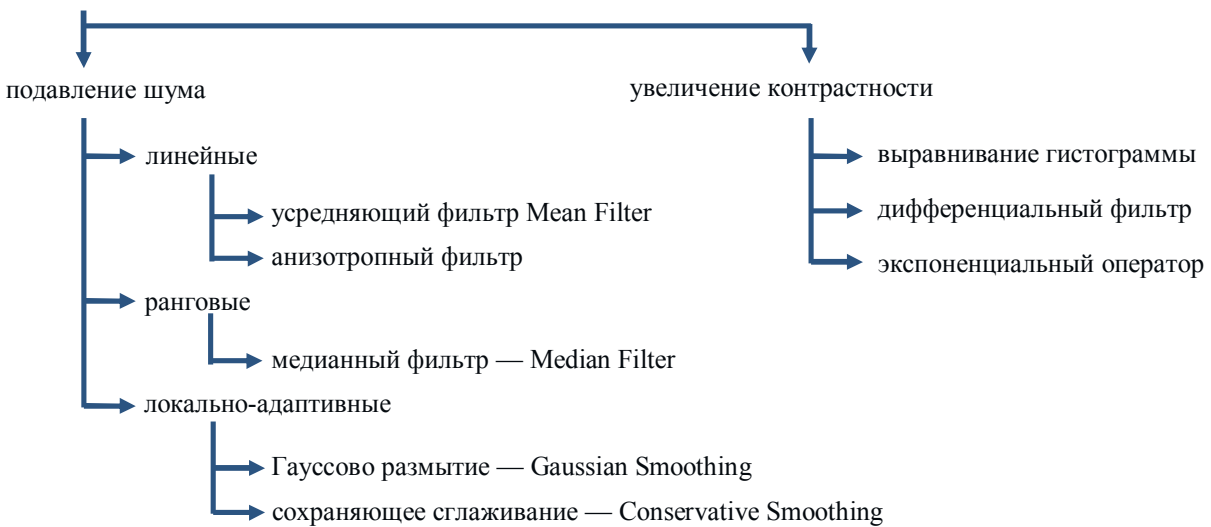
Под **фильтрацией** изображений понимают операцию, имеющую своим результатом изображение того же размера, полученное из исходного по некоторым правилам. В общем случае фильтрация решает следующие основные задачи:

- сглаживание (подавление высокочастотной помехи типа «снег»);

- повышение контрастности;
- выделение контура.

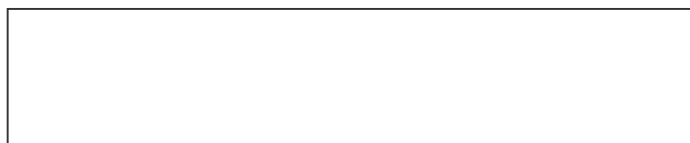
Функции преобразования Гауссова низкочастотного и высокочастотного фильтров в пространственной и частотной областях

Классификация фильтров



Возможность **сглаживания** обусловлена тем, что спектр визуальных помех обычно содержит более высокие пространственные частоты, чем спектр изображения, т. е. размер помехи существенно меньше размера фрагмента объекта.

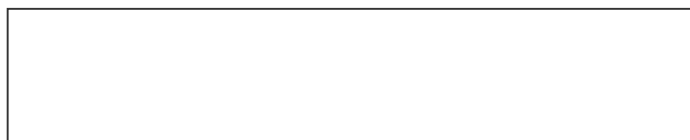
Метод **порогового сглаживания** основан на сканировании цифрового изображения программным окном размером $N \times N$ (где N равно 3 ... 7) и вычислении на каждом шаге средней яркости Y_{cp} группы элементов:



Сущность метод **анизотропной фильтрации** заключается в свертке исходного дискретного массива изображения Y и сглаживающего массива (фильтра) W размером $N \times N$. Результатом является новый (сглаженный) массив Y^* , элементы которого вычисляются согласно выражению:



В основу метода **рекуррентной фильтрации** положен тот же принцип свертки, что и при анизотропной фильтрации, однако здесь используются не только элементы исходного массива Y , но и элементы уже сглаженного массива Y^* . Следовательно, поэлементные операции проводятся согласно выражению



Медианная фильтрация для некоторой окрестности пикселя с координатами i, j описывается выражением

Для **выделения контура** наиболее распространены методы, основанные на использовании операторов **пространственного дифференцирования**, которые находят в каждом фрагменте изображения элементы, расположенные на границе различных по яркости областей.

Градиент яркости Γ_{ij} в некоторой точке x изображения с координатами i, j определяется как двухмерный вектор:

Норма градиента определяется выражениями

или приближенно

Основные алгоритмы выделения контуров



Оператор Робертса использует программное окно 2×2 .

Оператор Собеля работает в программном окне 3×3 пикселя.

В вычислительных процедурах контур изображения Y_{ij}^* получают путем **свертки**, т.е. умножения исходного изображения Y_{ij} на соответствующую маску \mathbf{M} . Для градиентных фильтров $\mathbf{M} = \mathbf{\Gamma}$.

Оператор Робертса

Оператор Собеля

Оператор Превитта

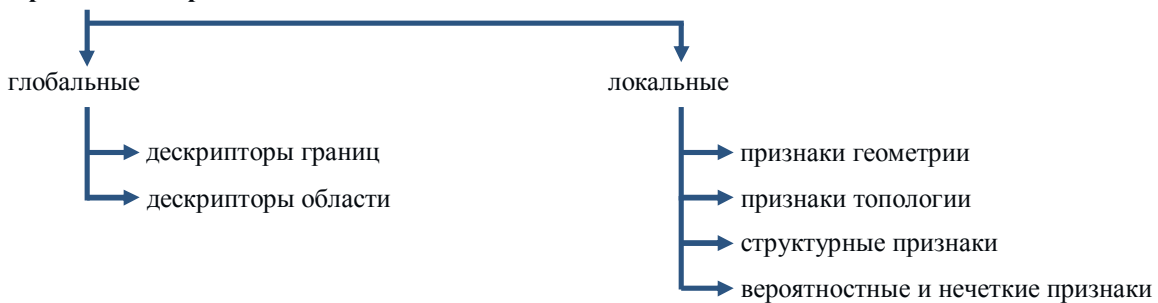
Процедура разделения отдельных контуров изображения и соотнесение их с определенными объектами называется **сегментацией**. Алгоритмы сегментации, как правило, основываются на двух процедурах:

- поиске **разрывности** в контуре;
- определении **подобия** областей.

Под **кодированием** изображения понимается обратимое преобразование информации, позволяющее получить компактный массив чисел, однозначно описывающий это изображение в удобной для данной вычислительной структуры форме.

Под **описанием** понимается определение характерных параметров объекта — **признаков** (дескрипторов), необходимых для его выделения из числа всех, образующих сцену.

Признаки изображения



Момент инерции порядка $p+q$ (где p, q — целые положительные числа) для

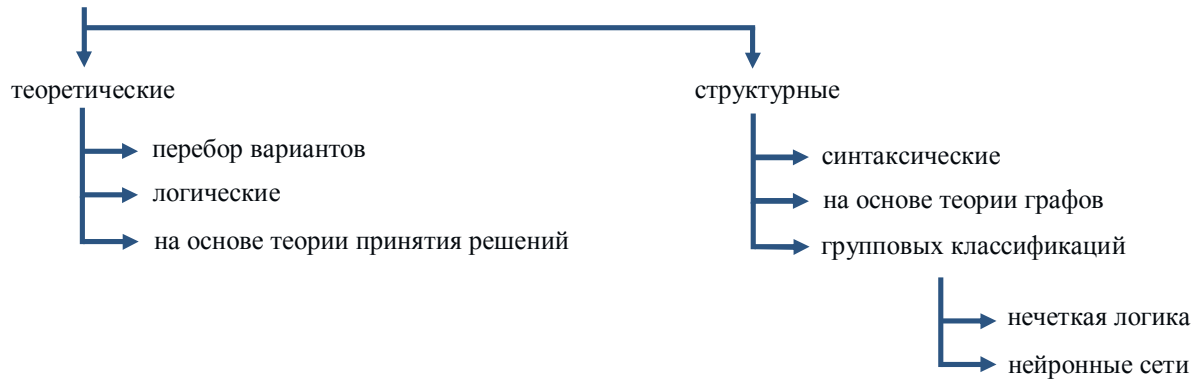
можно определить по формуле:

Центральный момент инерции порядка $p+q$ равен

4. Распознавание

Распознаванием называется процесс, при котором на основании набора признаков некоторого изображения объекта определяется его принадлежность к определенному **классу**.

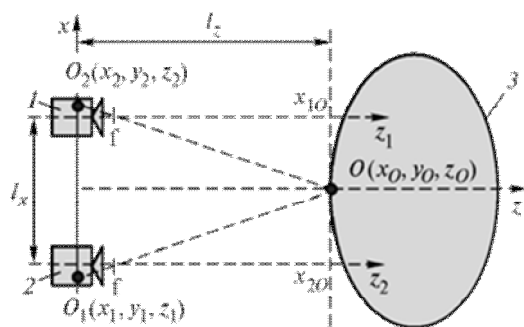
Методы распознавания



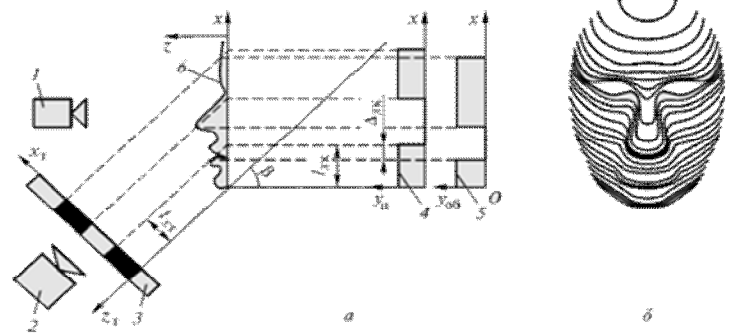
Упрощенная **математическая постановка** задачи распознавания сводится к классификации объектов, т. е. отношению априорно известных объектов к априорно известным классам.

5. Способы получения 3D изображения

В СТЗ под **трехмерным (3D)** понимают изображение, содержащее информацию о трех геометрических измерениях объекта. Трехмерное изображение может быть получено с помощью двух телекамер или с помощью специальных приемов.



Определение третьей координаты объекта с помощью двух телекамер: 1, 2 — телекамеры, 3 — объект



Определение третьей координаты по плоскому изображению: а — получение псевдотрехмерного изображения, б — трехмерная модель; 1 — телекамера; 2 — проектор; 3 — транспарант; 4, 5 — распределение яркости при плоской сцене при отсутствии и при наличии объекта; б — объект

Используя две телекамеры, координату z_0 точки объекта можно определить в системе координат каждой из телекамер.

На изображении для любой точки с координатами x_p, y_p , принадлежащей i -й линии транспаранта, можно восстановить **третью** координату z_p :